

Oracle Utilities Customer Care and Billing

Administration Guide

Volume 2

Release 2.3.1

E18368-01

September 2010

Copyright © 2000, 2010, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Defining Meter, Item & Equipment Options

Located at a premise's service points are the various meters, items and equipment that regulate and measure consumption. Before you can define meters, items and equipment you must set up the control tables defined in this section.

Refer to [An Illustration Of A Premise](#) for an illustration of how meters, items and equipment are related to the service points.

Contents

- [The Big Picture of Meters, Items and Equipment](#)
- [Setting Up Meter Options](#)
- [Setting Up Metered Service Point Options](#)
- [Setting Up Metered Premise Options](#)
- [Setting Up Consumption Estimation Parameters](#)
- [Setting Up Meter Read Options](#)
- [Setting Up Items](#)
- [The Big Picture Of Device Testing](#)
- [Setting Up Device Test Options](#)
- [The Big Picture Of Usage Requests](#)

The Big Picture of Meters, Items and Equipment

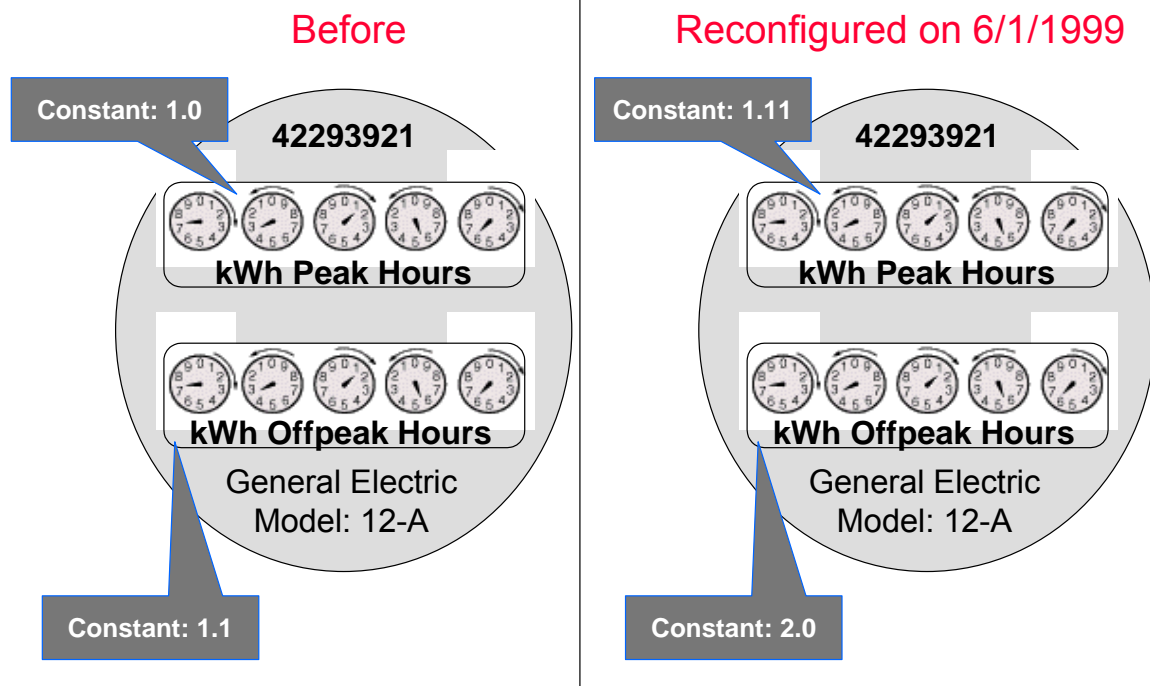
The topics in this section provide background information about meters, items and equipment.

Contents

- [The Structure Of A Meter](#)
- [Items Are Used For Other Devices Associated With A Customer's Service](#)

The Structure Of A Meter

A customer's consumption is measured using a meter. What a meter measures and how it measures it can change over time. The following example illustrates a simple meter before and after a reconfiguration.



The system maintains how a meter looks over time so that it can reproduce bills using historical consumption. Information about this meter is recorded in the following tables:

- Every meter has a single **meter** record that contains information about the meter that doesn't change over time. For example, its meter number and manufacturer.
- A **meter configuration** record is required whenever something changes about what the meter measures. The meter shown above has two configurations – the original and the one effective on 1-June-1999.
- Every meter configuration contains one or more **registers**. Each register references the unit being measured and how the measured quantity is manipulated before it is billed.

Information about how to set up meters is discussed in [Maintaining Meters](#).

Items Are Used For Other Devices Associated With A Customer's Service

In addition to meters, there are many other devices that can be involved with a customer's service. We refer to these other devices as "items".

We purposefully use the ambiguous term *item* because items are used for many different devices including lamps, poles, current transformers, backflow devices, pulse initiators, etc. Refer to the diagram in [An Illustration Of A Premise](#) for an example of the various items that could be associated with a customer's service.

The topics in this section provide more information about items.

Contents

[Equipment versus Badged Items](#)

Meters May Have Equipment
Items May Have Equipment
Service Points May Have Equipment
Chargeable Items Must Be Associated With Service Points
Start / Stop and Items
Billing and Items
Generic Equipment

Equipment versus Badged Items

This is a rather confusing subject, but there are two notional types of items:

- **Equipment.** An item that is considered to be “equipment” is a physical device that regulates consumption; it does NOT measure consumption. You would only define equipment if it is of interest to your organization. For example, if your organization periodically tests the pulse initiators associated with your meters, you will need to set up items for each pulse initiator and link them to their respective meters. Equipment can be linked to either a service point (e.g., a current transformer, a backflow device), a meter (e.g., a pulse initiator), or an item (e.g., the components of an installation).
- **Non Equipment.** An item that is not considered to be “equipment” is a physical device that does NOT measure consumption, but impacts billing in some way (i.e., there are charges in your rates based on the number and type of items installed at a service point). Examples include street lights, light poles, and security cameras. These types of items are related to service points. Refer to [Chargeable Items Must Be Associated With Service Points](#) for information about the two types of items that may be linked to service points.

The topics that appear below provide more information about both types of items.

Meters May Have Equipment

If you have physical devices that are intrinsically associated with meters (but aren't meters), you can set up an item for each such device. For example, if you have pulse initiators linked to your meters, you would set up an item for each pulse initiator. After creating these items, you would update each meter's collection of equipment to include the respective pulse initiator.

Items May Have Equipment

If you have physical devices that are intrinsically associated with other non-metered devices, you can set up an item for each such device. For example, you could set up an item for an “installation” (an installation is a group of devices that work together to regulate electric consumption) and link to it the individual items that do the work. To do this, you would set up items for each individual piece of equipment in the installation. After creating these items, you would update the collection of equipment associated with the installation's item to include each piece of equipment.

Service Points May Have Equipment

If you have physical devices that don't have line item charges but are intrinsically associated with a service point, you can set up an item for each such device. For example, if you have voltage regulators linked to your service points, you would set up an item for each voltage regulator. After creating these items, you would update each service point's collection of equipment to include the respective voltage regulator.

Chargeable Items Must Be Associated With Service Points

If there are charges in your rates based on the number and type of items installed at a service point, you can use either of the following techniques:

- If the item is badged (i.e., uniquely identified), you would set up an item and link it to an **item-based** service point. An item-based service point may have zero or one item installed at any instant in time. Over time, an item-based service point may have many badged items installed and removed. Refer to [Installing / Removing An Item](#) for more information.
- If the item does not have a unique identifier, you do NOT have to set up phony items for it. Rather, you can use the system's multi-item functionality and simply define the number of each TYPE of item that is installed at a service point. Refer to [Service Points With Multiple Items](#) for an illustration of such a service point.

Start / Stop and Items

When a customer start or stops service at an **item-based** service point, it takes into account whether an item is currently installed at the SP and whether the item is on or off. Start / Stop does not concern itself with equipment.

Billing and Items

It's important to note that billing ignores equipment relationships when it constructs the snapshot of number and types of items associated with a service agreement's service points. In other words, billing constructs the item snapshot as follows:

- It inserts an entry into the item snapshot for every item type referenced in the service agreement's service points' multi-item collection.
- It inserts an entry into the item snapshot for every badged item linked to service points with an SP type of **Item**.

Refer to [Item Snapshot](#) for more information.

Equipment and billing. Be aware that the only way equipment can impact billing would be if you developed a SQ or register rule that analyzed the equipment associated with a service point (directly or indirectly via the meters and items) and manipulated billed consumption accordingly. Refer to [Defining Measured Quantity Manipulation Rules](#) for more information.

Generic Equipment

Some companies do not badge their equipment, but they still need to keep track of the type of equipment that is linked to service points, meters and badged-items. For example,

- You may want to keep track of the type of modem linked to a meter, but you don't care about the individual modem or pulse-initiator
- You may want to keep track of the type of backflow device linked to a water meter

Why is generic equipment recorded in the system? Generic equipment gets linked to SP's / meters / items when the type of equipment linked to an SP / meter / item affects bill calculations. For example, some water companies charge extra if a backflow device is present at a service point.

If you encounter the need for generic equipment, do the following:

- Create an Item Type for each type of unbadged equipment. On this item type, indicate that items of this type may be linked to multiple SP's / meters / items. Note: we recommend suffixing the description of the item type with the word "generic".
- Update the relevant meter type(s) / SP type(s) / item type(s) to indicate that they may use equipment associated with the generic item type.
- Create one (1) item for each type of unbadged equipment. Note: we recommend suffixing the description of the item type with the word "generic".
- Link the generic item to the relevant SP's / meters / items.

Setting Up Meter Options

This section describes tables that must be set up before you can define meters.

Contents

- [Setting Up Meter Configuration Types](#)
- [Setting Up Meter Types](#)
- [Setting Up Manufacturers & Their Models](#)
- [Setting Up Meter ID Types](#)
- [Setting Up Read Out Types](#)
- [Setting Up Protocol Codes](#)
- [Setting Up Unit Of Measure Codes](#)
- [Setting Up Time-Of-Use Codes](#)
- [Setting Up TOU Groups](#)
- [Setting Up Retirement Reasons](#)

Setting Up Meter Configuration Types

Every meter configuration must reference a meter configuration type. The meter configuration type indicates the valid (required or optional) unit of measure and time of use registers for the configuration.

Navigate to **Admin Menu, Meter Configuration Type** to set up valid meter configuration types.

Description of Page

Enter an easily recognizable **Meter Configuration Type** and **Description** for the meter configuration type.

Choose the **Service Type** associated with the meter configuration type. Refer to [Setting Up Service Types](#) for more information.

Use **Prepaid Meter** to indicate whether or not the meter configuration is used to record prepaid usage.

If you have set up a **TOU Group** to indicate your collection of time of use codes for the meter configuration, enter it here. Use the **Default TOU Registers** button to have the system build the collection of registers for you based on the TOU codes linked to the TOU group. Refer to [Setting Up TOU Groups](#) for more information

In the grid, specify the attributes of the collection of valid registers. If the register is **Interval**, put a check in the checkbox. For any register, indicate the valid **UOM**. For interval registers, indicate the **Interval Register Type**. If a **TOU** is applicable for the register, enter the TOU code. Finally, specify a **Validation** for each register. The register pair can be **Optional** or **Required**.

Note. In certain markets, registers with the same **UOM** and **TOU** combination or, for **Interval** meters, the same **UOM** and **Interval Register Type** combination may need to be defined under a single meter configuration. If your implementation is in one of these markets, add the **Allow Duplicate UOM/TOU Combination Option Type** on the **Meter Management Options Feature Configuration** and set its value to **Y**.

Attributes related to interval registers may not appear. These attributes are suppressed if **Meter Data Management** module is [turned off](#).

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MTR_CONFIG_TY](#).

Setting Up Meter Types

Every meter references a meter type. The meter type defines the type of service and common characteristics shared by its meters. The meter type also controls the characteristics that may be specified on meters of a given type.

When a new meter type is added. After adding a new meter type, you must define the SP types at which meters of this type can be installed.

The topics in this section describe how to maintain your meter types.

Contents

- [Meter Type - Main](#)
- [Meter Type - Meter Characteristics](#)
- [Meter Type - Equipment Types](#)
- [Meter Type - Test Types](#)
- [Meter Type - Meter Configuration Type](#)

Meter Type - Main

To define a meter type, open **Admin Menu, Meter Type**.

Description of Page

Enter an easily recognizable **Meter Type** and **Description** for the meter type.

Choose the **Service Type** associated with all meters of this type. Refer to [Setting Up Service Types](#) for more information.

Turn on **Allow Duplicate Meter Badges** if more than one meter of this type may have the same badge number.

Turn on **Prepaid-Capable** if meters of this type are used to record prepaid usage.

Turn on **Allow Interval Registers** if meters of this type may contain interval registers. Refer to [The Big Picture of Raw Data Collection and Aggregation](#) for more information.

Indicate in **Track Location** whether you **Track** or **Do Not Track** the location of meters of this type. Refer to [The Big Picture of Asset Inventory](#) for more information.

Use the **Characteristic Values** collection to define **Characteristic Types** and their respective **Characteristic Values** and **Adhoc Values** to describe characteristics common to all meters of this type.

Note. You can only choose characteristic types defined as permissible on the meter type record.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MTR_TYPE](#).

Meter Type - Meter Characteristics

To define characteristics that can be defined for meters of a given type, open **Admin Menu**, **Meter Type** and navigate to the **Meter Characteristics** tab.

Description of Page

Use the **Meter Characteristics** collection to define characteristics that can be defined for meters of a given type. Turn on the **Required** switch if the **Characteristic Type** must be defined on meters of a given type. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on. Use **Sequence** to control the order in which characteristics are defaulted.

Meter Type - Equipment Types

Open **Admin Menu**, **Meter Type** and navigate to the **Equipment Types** tab to define the types of equipment that can be linked to meters of a given type. Refer to [Equipment versus Badged Items](#) for more information about equipment.

Description of Page

Use the collection to define the item types of **Equipment** that can be linked to meters of this type.

Item types are being specified. There is no equipment type control table. Rather, items are used to define equipment and therefore you are actually defining item types rather than equipment types. Refer to [Equipment versus Badged Items](#) for more information about equipment.

Meter Type - Test Types

Open **Admin Menu, Meter Type** and navigate to the **Test Types** tab to define the types of device tests that can be performed on meters of a given type. Refer to [The Big Picture Of Device Testing](#) for more information about device tests.

Description of Page

Use the **Device Test Type** collection to define the types of [device tests](#) that can be performed on meters of a given type.

Meter Type - Meter Configuration Type

Open **Admin Menu, Meter Type** and navigate to the **Meter Configuration Type** tab to define the types of meter configurations that are allowed for this type of meter.

Description of Page

Use the **Meter Configuration Type** collection to define the types of meter configurations that can be used for this type of meter.

Setting Up Manufacturers & Their Models

When you set up a meter (or an item) you must define the manufacturer and model number of the meter / item. To define a manufacturer and its models, open **Admin Menu, Manufacturer**.

Description of Page

Enter an easily recognizable **Manufacturer** and **Description** for the manufacturer.

Enter a **Model** and **Description** for every model supplied by the manufacturer. Enter the **Service Type** with which the model is associated. Refer to [Setting Up Service Types](#) for more information.

If the model is associated with a meter, turn on **Use On Meter**. If the model is associated with an item, turn on **Use On Item**.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MFG](#).

Setting Up Meter ID Types

A meter may have alternate ID's other than its badge number. If so, it will reference one or more meter ID types (one for each alternate form of ID). To define meter ID types, open **Admin Menu, Meter ID Type**.

Description of Page

Enter a unique **Meter ID Type** and **Description** for every meter ID type.

Where Used

A meter may have alternate ID's other than its badge number. If so, it will reference one or more meter ID types (one for each alternate form of ID). Refer to [Meter – Meter ID Information](#) for more information.

Setting Up Read Out Types

Every register has a read out type that defines how the register's measurements are physically displayed (e.g., circular dials, digital, n/a-electronic transmission, and so on). To define read out types, open **Admin Menu, Read Out Type**.

Description of Page

Enter a unique **Read Out Type** code and a **Description** for every read out type.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_READ_OUT_TYP](#).

Setting Up Protocol Codes

Every register has a protocol code that defines the method used to record the register's measurements (e.g., visual, probe, AMR, modem). To define protocol codes, open **Admin Menu, Protocol**.

Description of Page

Enter a unique **Protocol** and **Description** for every protocol.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_PROTOCOL](#).

Setting Up Unit Of Measure Codes

You must create a unit of measure (UOM) for:

- Every UOM that is measured by your meters' registers.
- Every UOM in which your rates' prices are expressed.
- Every UOM in which your items' estimated consumption is expressed.

Note. Typically, the measured UOM is the same as the UOM in which the prices are expressed. However, there are instances where the UOM of the price is not measured by a meter's registers. Natural gas is a good example – the prices are expressed in an amount per therm, but the gas meters measure some volume (e.g., cubic feet, cubic meters). Refer to [Defining Measured Quantity Manipulation Rules](#) for more information.

For more information about UOM's, refer to [UOM versus TOU versus SQL](#).

To define unit of measure codes, open **Admin Menu, Unit of Measure**.

Description of Page

The following fields display for each unit of measure:

UOM	The unique identifier of the unit of measure.
Description	The full description of the UOM.
Service Type	The type of service (e.g., electric, gas, water, and so on.) associated with this UOM. This value controls the UOMs that may be referenced on meters belonging to a given service type. Refer to Setting Up Service Types for more information.
Decimal Positions	The number of decimal positions that appear on bill lines that show consumption.

Suppression of trailing zeroes. If you wish to suppress trailing zeroes in the consumption that appears on bill lines, you must set up the ***Allow Zero Suppression Of Consumption In Bill Description*** option type on the ***Financial Transaction Options*** [Feature Configuration](#).

Allowed on Register	Turn on this switch if the UOM can be referenced on a meter's register. A unit of measure may not be allowed on a register when it exists purely because there is a price in a rate expressed in this UOM. (For example, prices of natural gas are frequently express in therms, but it is rare to find a meter that measures gas in therms.)
Measures Peak Quantity	Turn on this switch for UOMs that exist to record the peak amount of consumption, e.g., KW and KVAR. Peak UOMs are treated differently by billing when determining the amount of consumption. They are also treated differently if rating has to prorate consumption.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI UOM](#).

Setting Up Time-Of-Use Codes

You will only create time of use (TOU) codes if your meters measure consumption in respect of broad time bands. For example, some electric meters measure consumption in respect of WHEN the power was used – peak period, off peak period, or partial peak period.

If you have “time of use” meters, then you must create TOU codes for:

- every TOU that is measured by your meters, and
- every TOU in which your rates' prices are expressed.

Note. Typically the measured TOU is the same as that in which the prices are expressed. However, there are instances where the TOU of the price is not measured directly by a meter's registers. A simple example is where the meter measures peak hours consumption and total consumption but there are prices in the rate for off peak consumption. In this situation, the peak hours consumption would have to be subtracted from the total consumption to derive off peak consumption. In this situation, you would need TOU codes for both peak (measured) and off-peak (derived) periods.

For more information about TOU's, refer to [UOM versus TOU versus SQL](#).

Time-of-use codes will also be needed if your company offers interval billing with TOU mapping. Refer to [Designing Your Time of Use Options](#) for more information.

To define time-of-use codes, open **Admin Menu, Time of Use**.

Description of Page

Enter a unique **TOU** code and **Description** for every time of use code.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI TOU](#).

Setting Up TOU Groups

TOU Groups may be used to group together time of use codes, which are included in a meter configuration. It is also used in TOU mapping and pricing functionality.

Refer to [Grouping of TOU Codes](#) for more information about the TOU group's role in TOU mapping and pricing.

Open **Admin Menu, TOU Group** to define your TOU Groups.

Description of Page

Enter a unique **TOU Group** and **Description** for the TOU group.

Enter the Collection **Time of Use** codes. This is a list of time of use codes that define the time of use periods to be used for certain TOU maps. For each time of use code, you may indicate a **TOU Sequence** to indicate the relative order or relative priority of each TOU for the TOU group. Refer to [Grouping of TOU Codes](#) for more information.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI TOU_GRP](#).

Setting Up Retirement Reasons

When you change a meter or item's status to **Retired**, you must supply a retirement reason. To define retirement reasons, open **Admin Menu, Retire Reason**.

Description of Page

Enter a **Retire Reason Code** and **Description** for every retirement reason.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_RETIRE_RSN](#).

Setting Up Metered Service Point Options

This section describes tables that must be set up before you can define metered service points.

Contents

[Defining Meter Location Codes](#)

[Setting Up Service Cycles And Routes](#)

Defining Meter Location Codes

When you set up a metered service point you must define where the meter's service point is located on the property.

Note. The meter location code is provided to meter readers and field workers to help locate the meter.

To define meter location codes, open **Admin Menu, Meter Location**.

Description of Page

Enter a unique **Meter Location** and **Description** for every meter location.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MTR_LOC](#).

Setting Up Service Cycles And Routes

Refer to [Designing Service Cycles, Routes, And Schedules](#) for more information.

Setting Up Metered Premise Options

This section describes tables that must be set up before you can define premises.

Contents

[Setting Up Meter Read Instructions](#)

[Setting Up Meter Read Warnings](#)

Setting Up Meter Read Instructions

When you set up a premise you may define instructions to be supplied to the individuals who read the meters located at the premise. To define meter read instruction codes, open **Admin Menu, Meter Read Instruction**.

Description of Page

Enter a unique **Meter Read Instruction Code** and **Description** for every meter read instruction.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MR_INSTR](#).

Setting Up Meter Read Warnings

When you set up a premise you may define warnings to be supplied to the individuals who read the meters located at the premise. To define meter read warning codes, open **Admin Menu, Meter Read Warning**.

Description of Page

Enter a unique **Meter Read Warning** and a **Description** for every meter read warning.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MR_WARN](#).

Setting Up Consumption Estimation Parameters

The system estimates consumption under the following situations:

- At billing time, when “real” consumption cannot be computed due to the lack of a meter read, the system estimates how much the customer used assuming the following conditions are true:
 - The service agreement allows estimation.
 - The rate schedule allows estimation.
 - The bill cycle schedule (if one is being used) allows estimation.
 - The register is not a peak register (this is an artificial constraint, but most organizations do not estimate peak consumption).
 - The register is subtractive. This is because only subtractive registers are self-correcting. For example, assume we estimate a reading of 100 and the next real read is 102, the customer will only be charged for 2 units. But consider what would happen if we estimated a consumptive register, the consumption associated with the next real read will be billed in its entirety.

- When a meter read is added, the system verifies that the resultant consumption is congruous with historical consumption trends (i.e., high / low checks are performed).

This section describes the tables that must be set up to estimate consumption.

Contents

- [Estimating Consumption](#)
- [Setting Up Trend Areas](#)
- [Setting Up Trend Classes](#)
- [Setting Up High / Low Factors](#)
- [Setting Up Trends](#)

Estimating Consumption

How the system generates estimated consumption is up to you because a plug-in algorithm is called when it's time to estimate consumption. The identity of the plug-in algorithm is defined on the [trend area](#) of the premise at which the meter is installed.

The base package is supplied with two algorithm types that you may use to estimate consumption:

- The [TSMRE-LA](#) algorithm type first tries to estimate consumption using historical data for the account and service point. If this is unsuccessful, it uses trend data to estimate consumption.
- The [MR EST TREND](#) algorithm type uses trend data to estimate consumption.

If neither of these algorithms works for your organization, you will need to write a new algorithm and plug it in on your trend areas.

The following discussion explains how the sample estimation plug-ins work.

Contents

- [Estimating Using Historical Consumption](#)
- [Estimating Using Trends](#)
- [Tips for Consumption Estimation Using Trends](#)
- [Estimation and Negative Consumption](#)

Estimating Using Historical Consumption

One of the sample estimation algorithms provided with the base package attempts to estimate consumption using historical data for the account and service point.

When you estimate consumption using historical data, you assume that the customer will have current consumption that is similar to their historical usage. For example, if you need to estimate consumption for the month of February of the current year, you could assume that the customer has similar consumption during the month of February of the previous year or that they have similar consumption to the previous month.

The base package algorithm type that uses a customer's historical data to estimate consumption is called [TSMRE-LA](#) (three-step meter read estimation). It makes three attempts to estimate consumption stopping at the first successful attempt.

First, it tries to estimate consumption using a bill segment for the service point from the previous year that covers a similar period to the one being estimated. The year-old bill segment can be used for estimation if it is for the same account AND the bill segment's end read is not a system estimate AND there are enough days in the bill segment for estimation (as defined by the algorithm).

If the year-old bill segment cannot be used for estimation, the system attempts to use the bill segment for the account and service point that immediately precedes the start date of the estimation period. Again, this bill segment must not have an estimated end read and it must have a sufficient number of days for estimation.

Once a historical bill segment that can be used for estimation is located (using one of the above methods), the estimation occurs as follows:

- If the register is non-peak, the system calculates the average amount of consumption per day on the historical bill segment and multiplies this by the number of days in the current estimation period. For example, if the customer averaged 30.350877 kWh per day in the historical bill period and there are 30 days in the current period, the estimated consumption is: 911 kWh.
- If the register is peak, the estimation amount is the same as the peak amount on the historical bill segment.

If the system is unsuccessful in estimating consumption using the historical data for the previous year or the previous bill period, the system estimates consumption [using trends](#).

Estimating Using Trends

Warning! The topics that appear below explain how the [MR EST TREND](#) sample estimation algorithm works. Another algorithm is also available in the base package. It attempts to estimate consumption using the customer's historical consumption; if historical consumption does not exist, it estimates using the logic described below (refer to [Estimating Using Historical Consumption](#) for the information about this algorithm).

Contents

[The Theory Behind Consumption Estimation Using Trends](#)
[Subtractive Register Where UOM Does Not Measure Peak Quantity](#)
[Consumptive Register Where UOM Does Not Measure Peak Quantity](#)
[Subtractive Register Where UOM Measures A Peak Quantity](#)
[Consumptive Register Where UOM Measures A Peak Quantity](#)

The Theory Behind Consumption Estimation Using Trends

The following discussion explains how the sample estimation plug-ins use trend data. If you decide to create your own version of one of these plug-ins, you still have to set up trend areas and trend classes as these are required fields on premises and SP types.

The standard estimation plug-ins assume that if a customer historically used, say, twice as much as customers of a similar profile, then the customer should use twice as much in the current period. Therefore, to estimate consumption for any period of time the system needs to know:

- A. How much the specific customer used in the previous consumption period.
- B. How much the average customer in the customer's trend profile used in the previous consumption period.
- C. How much the average customer in the customer's trend profile used in the period being estimated.

Once the above are known, estimated consumption for the specific customer equals $(A/B) * C$.

Knowing how much the specific customer used in the previous consumption period simply involves looking at the customer's previous readings (or reading, in the case of consumptive meters).

Knowing how much the average customer used requires consumption trend data. Consumption trend information is continuously updated behind-the-scenes using meter reads. A consumption trend is identified by a unique combination of:

- **Trend area.** Trend areas are used to differentiate consumption trends in different geographic areas. If your service territory doesn't have appreciable differences based on geography, then you will have just one trend area.
- **Trend class.** Trend classes are used to differentiate consumption trends based on the type of service and the type of property. At a minimum, you would probably have a trend class to differentiate between residential, commercial, and industrial usage.
- **Unit of Measure and Time of Use.** All consumption in the system is identified using a unit of measure code and, optionally, a time-of-use code.

For more information about unit of measure codes, see [Setting Up Unit Of Measure Codes](#). For more information about time-of-use codes, see [Setting Up Time-Of-Use Codes](#). Trend areas, trend classes, and consumption trends are described later in this section.

Bottom line. A separate consumption trend is maintained behind-the-scenes for every combination of trend area, trend class, unit of measure and time of use.

Subtractive Register Where UOM Does Not Measure Peak Quantity

Assume the following read history exists for a subtractive register (i.e., one where you have to subtract the previous read from the current read in order to derive consumption) where the register's unit of measure doesn't measure a peak quantity.

Read Date	Reading	Derived Consumption
15-Jan-1999	1000	N/A (first read)
15-Feb-1999	3000	2000
15-Mar-1999	4500	1500

Next, assume the meter's trend profile looks as follows:

Read Date	Total Qty (Assume kWh)	No of Units (Total Days)	Number of Reads	Average Consumption (Per Day)
13-Mar-1999	6,000,000	135,000	4,500	44.444444
14-Mar-1999	900,000	15,000	500	60
15-Mar-1999	5,000,000	137,750	4,750	36.297641
...				
13-Apr-1999	4,000,000	135,000	4,500	29.629630
14-Apr-1999	4,650,000	155,000	5,000	30

Note. The system keeps a separate trend for every combination of trend area, trend class, unit of measure and time of use code. Trend area comes from the premise at which a meter is installed. Trend class comes from the SP type of the service point at which a meter is installed. Unit of measure and time of use come from the register read's register.

Next, assume a new register read is recorded on 15-Apr-1999 with a value of 5000.

In order to calculate estimated consumption for this register read, we have to calculate average daily consumption using the following formula:

$$\frac{\text{This customer's usage in previous period}}{\text{Avg customer's usage in previous period}} \times \text{Avg customer's usage in current period}$$

The following points describe exactly how the system calculates each variable in this formula:

- Determine **Average Customer's Usage In Current Period**:
 - Extract the number of reads from the meter's service point's SP type's trend class. We'll assume this is set to 7,500 reads.
 - Read trend records in reverse chronological order from the read date (assuming there is no trend record on 15-Apr-1999, we'll start amassing trends from 14-Apr-1999).
 - Read enough trend records until the number of reads is greater than or equal to the trend class' number of reads. We will have to read 2 trends records to amass this value (the ones on 14-Apr-1999 and 13-Apr-1999). We remember how many reads we extracted from the trend table for use a little later. In this case, we used 9,500 reads.
 - Divide **Total Qty** by **No of Units**. Using our example, we'd divide 8,650,000 kWh by 290,000 days to get 29.827586 kWh per day.
- Determine **This Customer's Usage In Previous, Non-Estimated Period**:
 - Find the consumption associated with the previous, non-estimated read for the register's unit of measure / time of use codes at the service point. In this case, we'd find the read on 15-Mar-1999. Because this register is subtractive, we'd also need to find the previous, previous non-estimated read in order to calculate consumption. In this case, we'd find the read on 15-Feb-1999.

Minimum Days Between Readings. The algorithm parameter Minimum Days Between Readings controls the minimum number of days needed between the previous non-estimated read and the "previous, previous" non-estimated read.

- Divide the total consumption by the number of days. Using our example, we'd divide 1,500 kWh by 28 days to get 53.571429 kWh per day.
- Note: if the customer doesn't have consumption in the previous, non-estimated period, e.g., if it's a meter at a new premise, the system assumes the customer uses the same as the average customer's usage in the previous period (see next point).
- Determine **Average Customer's Usage In Previous Period**:
 - Read trend records in reverse chronological order from the read date of the previous reading (15-Mar-1999).
 - Read enough trend records until the number of reads is greater than or equal to the number of reads amassed when determining the Average Customer's Usage In The Current Period (the first point). Using our example, we'd have to read 3 trends records (the ones on 15-Mar-1999, 14-Mar-1999 and 13-Mar-1999).
 - Divide **Total Qty** by **No of Units**. Using our example, we'd divide 11,900,000 kWh by 287,750 days to get 41.355343 kWh per day.
- Next, determine the number of days of estimated consumption. This will be equal the number of days between the estimation date and the prior reading (note, the prior reading could have been estimated). Using our example, we'd have 31 days (the number of days between 15-Apr-1999 and 15-Mar-1999).
- At this point, we have everything we need to estimate consumption. This will equal $((53.571429 / 41.355343) * 29.827586) * 31$ days. This is equal to 1,198 kWh.
- If we need to calculate high and low boundaries, we multiply 1,198 kWh by the high and low values defined for the register's unit of measure, the service point's service type and the read's read type.

Consumptive Register Where UOM Does Not Measure Peak Quantity

The algorithm used to estimate consumption for a consumptive, non peak unit of measure is identical to [Subtractive Register Where UOM Doesn't Measure Peak Quantity](#); the only difference is that we don't have to find the previous, previous read in order to determine the customer's usage in the previous period.

Subtractive Register Where UOM Measures A Peak Quantity

The algorithm used to estimate consumption for a subtractive, peak unit of measure is identical to [Subtractive Register Where UOM Doesn't Measure Peak Quantity](#) the previous example except:

- The No of Units on the trend table is not the number of days. Rather, it's the number of reads that contributed to the trend.
- Because we don't care about number of days, we don't have to multiply final estimated consumption by the number of days in the estimation period.

Consumptive Register Where UOM Measures A Peak Quantity

The algorithm used to estimate consumption for a consumptive, peak unit of measure is identical to [Subtractive Register Where UOM Measures A Peak Quantity](#) with the exception that we don't have to find the previous, previous read in order to determine the customer's usage in the previous period.

Tips for Consumption Estimation Using Trends

The quality of the trend information and setup impacts your estimation results (depending on your estimation algorithm) and this in turn impacts any estimated bills you send your customers. When you are setting up consumption estimation parameters keep the following guidelines in mind.

Contents

- [Make Sure the Number of Reads on Trend Class Is Large Enough](#)
- [Trend Classes and Trend Areas Should be Defined Appropriately](#)
- [Customers Should Be Classified Into Appropriate Trend Classes and Areas](#)

Make Sure the Number of Reads on Trend Class Is Large Enough

You should make the Number of Reads on the [trend class](#) is large enough that individual reads do not cause statistical anomalies. For example, assume that a "normal" read in a trend class is 500 kWh and, that within the trend class, there is one "abnormal" customer using 500 percent of normal:

- If the number of reads in the class is 10, the average read is calculated to be 700 kWh. The abnormal read causes a 40 percent increase in the average.
- If the number of reads in the class is 100, the average read is calculated to be 520 kWh. The abnormal read causes a 4 percent increase in the average.
- If the number of reads in the class is 1000, the average read is calculated to be 502 kWh. The abnormal read causes a 0.4 percent increase in the average.

By sizing the number of reads in a trend class appropriately, you can reduce the impact of statistical anomalies.

Example Values. The example above is meant to illustrate the importance of using an appropriate number of reads in a trend class. The numbers used in the example should not be viewed as recommended values when setting up your trend classes.

Trend Classes and Trend Areas Should be Defined Appropriately

In addition to defining a large enough number of reads on the trend class, you want to make sure that the system can collect the number of reads in a relatively short period of time or the benefit of trends is diluted. For example, if the system collects the number or reads over a period of two days as opposed to two months, the seasonal trend information is better represented in your estimates.

Consider the following example. Assume the number of reads required in the trend sample is 1000 and your system collects the following trend information for the area/class:

Month	Number of Reads	Average
-------	-----------------	---------

March	400	200 kWh
April	400	300 kWh
May	400	500 kWh

The trended average in May is 333.33 kWh $((80000 + 120000 + 200000) / 1200)$.

However, if the number of reads required in the trend sample is 1000 and your system collects the following trend information for the area/class:

Month	Number of Reads	Average
March	1100	200 kWh
April	1100	300 kWh
May	1100	500 kWh

The trended average in May is 500 kWh $(550000/1100)$, which is a more accurate representation of the trend in May.

Example Values. The examples above are meant to illustrate the importance of ensuring that the system collects enough reads in a short enough time span to accurately capture trends. The values are summarized by month and do not represent actual records in a trend profile.

The collection of an appropriate number of reads is a function of the number of reads defined for a trend class and the number of customers who are in each trend class/trend area combination. Make sure that the number of reads is not too large for the number of customers who are in each trend class/trend area combination and that the trend class/trend area combinations do not create groups of customers that are too small to calculate accurate trends.

Customers Should Be Classified Into Appropriate Trend Classes and Areas

Make sure that your customers are classified into appropriate trend area/trend class categories. If a customer's actual usage does not follow the trend (within a certain percentage) for the area/class in which the customer is classified, the consumption estimates for that customer may be inaccurate. For example, a residential customer should not be in a trend class with industrial customers.

A customer's premise references the trend area in which the customer is classified. A customer's service point type references the trend class in which the customer is classified.

Estimation and Negative Consumption

In some cases, a service point may be fitted with its own renewable energy source such as wind turbines or solar panels. The energy generated at this service point, as well as any energy used, could possibly be measured by a single meter, with the energy generated being netted against the energy used (a situation also referred to as net-metering). When the energy generated exceeds the energy used, negative consumption can result.

The possibility of negative consumption occurring at a particular SP/meter combination is indicated by the presence of a characteristic on the SP that identifies it as one that has generation capacity, as well as a characteristic on the meter type that identifies the meter as one capable of rolling backwards. Both characteristics must be present before negative consumption is allowed for that SP/meter combination. These characteristic types and values are defined on the [Meter Management Options Feature Configuration](#). The following points describe the various **Option Types** that must be defined if negative consumption is allowed by your implementation:

- **Negative Consumption SP Characteristic Type.** This option indicates the characteristic type used on service points to denote generation capacity.
- **Negative Consumption SP Characteristic Value.** This option indicates the characteristic value used on service points to denote generation capacity.
- **Negative Consumption Meter Type Characteristic Type.** This option indicates the characteristic type used on meter types to identify meters capable of rolling backwards.
- **Negative Consumption Meter Type Characteristic Value.** This option indicates the characteristic value used on meter types to identify meters capable of rolling backwards.

Because the factors that cause a meter to roll forward are very different from the factors that cause it to roll backward, it is impracticable for the estimation algorithms to try to derive an estimate in situations where negative consumption is allowed. The estimation rules simply cannot combine the effects of all these disparate factors into one meaningful figure. By extension, the same issue applies to the determination of high/low factors. Besides the fact that high/low validation is impossible without first obtaining an estimate, a meaningful set of high/low factors cannot be determined when it is not known how energy generated at the service point may have affected the consumption in the first place. Therefore, no estimation is done by the system for SP/meter combinations that allow negative consumption. Reads that belong to these SP/meter combinations are also excluded from the trending process so as not to skew the rest of the estimates for a certain trend area and trend class.

Setting Up Trend Areas

When you set up a premise, you must define the consumption trend area in which it is located. This categorization matters when consumption trends differ across your service territory AND you want the system to estimate consumption in different areas differently.

This categorization does not have to be done in respect of classic geographic boundaries like cities and counties. Rather, trend areas may be based on economic factors, climatic conditions, or anything else related to geography that affects consumption.

Note. A premise's trend area will default based on its postal code. See [Setting Up Premise & Service Point Postal Defaults](#) for more information.

To define trend areas, open **Admin Menu, Trend Area**.

Description of Page

Enter a unique **Trend Area** and **Description** for every trend area.

Define the **Meter Read Estimate Algorithm** that is used to estimate consumption for meters installed at premises associated with this trend area. If you haven't done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that performs meter read estimation. Click [here](#) to see the algorithm types available for this plug-in spot.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_TREND_AREA](#).

Setting Up Trend Classes

When you set up an SP type, you must define the consumption estimation trend class in which its consumption will be categorized. This categorization matters when consumption differs based on the type of property AND you want the system to estimate consumption in different classes differently.

These categories could be the classic divisions of residential versus commercial versus industrial consumers. Alternatively, they could be finer-grained divisions: single family residence, versus duplex, versus triplex, versus medical office, versus grocery store.

To define trend classes, open **Admin Menu, Trend Class**.

Description of Page

Enter a unique **Trend Class** and a **Description** for every trend class.

Enter the **Number of Reads** that must be amalgamated to create a statistically significant sample when amassing the average customer's consumption. The system uses this number to determine the number of consumption trend records to amalgamate.

Don't worry. The Number of Reads is not the number of rows that will be read when the system estimates consumption. This is because the total consumption from many reads is stored on a single consumption trend record.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_TREND_CL](#).

Setting Up High / Low Factors

When consumption estimation is done for the purpose of meter read validity, the system checks if the meter read's consumption is reasonable. The High / Low Meter Read algorithm type ([HILO-FACTBL](#)) supplied with the base package uses the factors defined in this table to calculate the high / low values.

High and low factors are defined for every measurable unit of measure (UOM) and read type combination. The word measurable was underlined because some UOMs exist purely for pricing purposes and are never measured by a meter (e.g., a therm of gas).

For more information, refer to [High / Low Checks](#).

To define high / low factors, open **Admin Menu, High Low Factor**.

Description of Page

Enter the **Unit of Measure** (UOM) for which the high / low factors are used.

Enter the **Read Type** for which the high / low factors are used. Valid values are: **Billing Force**, **Customer Read**, **Office Estimate**, **No Read**, **Service Provider Estimate**, **System Estimate**, **Regular**, and **Verified**.

Enter the **Low Factor** for the UOM and Read Type. This value will be multiplied by estimated consumption to derive the acceptable low value of a meter read with this UOM and read type.

Enter the **High Factor** for the UOM and Read Type. This value will be multiplied by estimated consumption to derive the acceptable high value of a meter read with this UOM and read type.

Where Used

This information is used by the High / Low Meter Read algorithm type ([HILO-FACTBL](#)) when the system calculates the low and high consumption amounts against which a meter read's consumption is compared when the read is added to the system.

Detecting Theft Of Service. When the status of a meter is **Off**, the system sets the high and low read values equal to the previous register read. This catches any consumption activity at the service point after a meter has been turned off. However, for some implementations, a small amount of consumption trickle at the service point is acceptable. In this case, you would want to relax the high/low thresholds so that the reads are not constantly flagged to have failed high/low validation. To do this, you can add the **Always Estimate And Apply High/Low Factors Option Type** under **Meter Management Options Feature Configuration** and set its value to **Y**.

Setting Up Trends

Refer to [Setting Up Trends](#) for more information.

Setting Up Meter Read Options

This section describes tables that must be set up before you can enter a meter read.

Contents

- [Setting Up Meter Reader Remarks](#)
- [Setting Up Meter Read Sources](#)

Setting Up Meter Reader Remarks

When you enter a meter read, you may define remarks using remark codes. The topics in this section describe how to set up meter reader remark codes.

Contents

[Meter Reader Remark - Main](#)

[Meter Reader Remark - Bill Messages](#)

Meter Reader Remark - Main

To define meter read remark codes, open **Admin Menu, Meter Reader Remark**.

Description of Page

Enter a unique **Meter Reader Remark** and a **Description** for every meter read remark.

Turn on **Eligible for Processing** if meter reads marked with a given remark code should cause one or more **Action Algorithms** to execute. For example, if you need a “reread” field activity to be created when a “high bill” remark is uploaded, you’d create an algorithm called “reread” and associate it with the “high bill” remark code. Then, whenever such a meter read with a “high bill” remark is recorded in the database, the system will execute the algorithm (and generate the field activity). If you haven’t done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that is associated with a meter read remark special activity. Click [here](#) to see the algorithm types available for this plug-in spot.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_READER_REM](#).

Meter Reader Remark - Bill Messages

To define bill messages to appear on bills that use a meter read that references a given meter reader remark code, open **Admin Menu, Meter Reader Remark** and navigate to the **Bill Messages** tab.

Description of Page

Use the **Bill Messages** collection to define **Bill Message** codes that should appear on bills that use a meter read that references a given meter reader remark code. For each message, also specify the **Start Date** and **End Date** when such a message should appear on the bill (leave **End Date** blank if the message should appear indefinitely).

Where Used

The system snaps bill messages on a bill during bill completion. Refer to [The Source Of Bill Messages](#) for more information.

Setting Up Meter Read Sources

When you add or upload a meter read, you may define the source of the meter read. The source could reference a specific reader, a meter reading agency, or any other possible source. To define meter read sources, open **Admin Menu, Meter Read Source**.

Description of Page

Enter a **Meter Read Source** code and **Description** for every meter read source.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MR_SOURCE](#).

Setting Up Items

This section describes tables that must be set up before you can define items.

Contents

- [Setting Up Item Types](#)
- [Setting Up Manufacturers and Models](#)

Setting Up Item Types

Every item has an item type that defines characteristics common to all items with this type. The topics in this section describe how to set up your item types.

Contents

- [Item Type - Main](#)
- [Item Type - Item Characteristics](#)
- [Item Type - Equipment Types](#)
- [Item Type - Test Types](#)

Item Type - Main

To define item types, open **Admin Menu, Item Type**.

When a new item type is added. After adding a new item type, you must define the SP types at which items of this type can be installed.

Refer to [Service Points \(SPs\)](#) for information about the difference between badged and non-badged items. Refer to [Items Are Used For Other Devices Associated With A Customer's Service](#) for more information about items in general.

Description of Page

Enter an easily recognizable **Item Type** and **Description** for the item type.

Choose the **Service Type** associated with all items of this type. Refer to [Setting Up Service Types](#) for more information.

Turn on **Use Estimates** if estimated consumption is recorded for this type of item (e.g., lamps have estimated consumption that is used to calculate billable charges for these types of items). When this switch is turned on, also define the **Unit Of Measure** in which the estimated consumption is expressed.

For more information about estimated consumption, refer to [Setting Up Estimated Consumption For Lamps](#).

Turn on **Summarize For Billing** if billing is supposed to summarize all items of this type on a customer's bill. If billing is supposed to show a separate bill line for every individual item of this type, turn this switch off.

Warning! The Summarize For Billing switch is only pertinent if badged items are linked to the service points being billed. If unbadged items are linked to the service points, the bill will contain a summary of items by item type regardless of the value of this switch. Refer to [Metered versus Item-Based versus Non-Badged Service Points](#) for more information.

Turn on **Multiple Equipment Assignment** if equipment of this type can be linked with more than one service point / meter / item at any instant in time.

Turn on **Billable** if billing should amalgamate items of this type when it calculates a bill segment. This switch would typically only be turned-off for items used to describe [equipment](#) and other non-billable items linked to a service point.

Warning! If this switch is turned on, rates linked to service agreements used to bill for items of this type must include an [item type rate component](#) that references this item type. Why? Because we assume that a bill line should be produced for "billable" items (and bill lines are produced for items using "item type" rate components). If you neglect to have such a rate component, a bill segment error will be produced.

If your organization bills for items of this type using [estimated consumption](#) rather than on a "per item" basis, you'll still need an "item type" rate component in these rates (otherwise billing will generate an error). In order to prevent the system from calculating a charge for such an "item type", make sure to indicate the rate component is "for calculation purposes only" (FCPO). Refer to [Rate Component - Main](#) for more information about FCPO's. Also note that you might also want to indicate that the rate component is not printable if you want to suppress the item type's description on the printed bill. Refer to [Rate Version - Bill Print Info](#) for more information about printable rate components.

Indicate in **Track Location** whether you **Track** or **Do Not Track** the location of items of this type. Refer to [The Big Picture of Asset Inventory](#) for more information.

Use the **Characteristic Values** collection to define **Characteristic Types** and their respective **Characteristic Values** to describe characteristics common to all items of this type.

Note. You can only choose characteristic types defined as permissible on the item type record.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_ITEM_TYPE](#).

Item Type - Item Characteristics

To define characteristics that may be defined for items of a given type, open **Admin Menu, Item Type** and navigate to the **Item Characteristics** tab.

Description of Page

Use the **Item Characteristics** collection to define characteristics that can be defined for items of a given type. Turn on the **Required** switch if the **Characteristic Type** must be defined on items of a given type. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on. Use **Sequence** to control the order in which characteristics are defaulted.

Item Type - Equipment Types

Open **Admin Menu, Item Type** and navigate to the **Equipment Types** tab to define the types of equipment that can be linked to items of a given type. Refer to [Equipment versus Badged Items](#) for more information about equipment.

Description of Page

Use the collection to define the item types of **Equipment** that can be linked to items of this type.

Item types are being specified. There is no equipment type control table. Rather, items are used to define equipment and therefore you are actually defining item types rather than equipment types. Refer to [Equipment versus Badged Items](#) for more information about equipment.

Item Type - Test Types

Open **Admin Menu, Item Type** and navigate to the **Test Types** tab to define the types of device tests that can be performed on items of a given type. Refer to [The Big Picture Of Device Testing](#) for more information about device tests.

Description of Page

Use the collection to define the types of **Device Tests** that can be performed on items of a given type.

Setting Up Manufacturers and Models

When you set up an item (or a meter) you must define the manufacturer and model number of the meter / item.

Refer to [Setting Up Manufacturers & Their Models](#) for more information.

The Big Picture Of Device Testing

The topics in this section describe meter and item (i.e., device) testing at a high level.

We strongly recommend examining the demo data to see how the concepts explained in this section could be used in real life.

Warning! Setting up the device testing control tables is as challenging as your organization's business rules. If you don't test your devices, you don't have to setup anything. If you have sophisticated testing requirements, your setup process will be more challenging.

Contents

[The Level of Complexity Depends On What You Test and Your Record Keeping Requirements](#)
[A Device Test Records Test Results](#)
[Field Activities And Device Testing](#)
[Device Test Validation](#)

The Level of Complexity Depends On What You Test and Your Record Keeping Requirements

Every organization's device testing requirements are different. Consider the following:

- Some organizations simply use the CIS system to create and dispatch field activities when it's time to test devices. After the field activities are dispatched, the test results are maintained in a separate system. You can set up the system to work this way for you.
- Some organizations maintain very detailed test result records, others don't. For example, some organizations record both "as found" and "as left" test results, i.e., they test the meter in its "as found" state, then they recalibrate it and test it again and record the "as left" state. You can set up the system to work this way for you.
- Some organizations maintain separate test results for every individual register on a meter. For example, they might record 3 separate test results for an electric meter with kWh, KW, and Kvar registers (and, if they recorded both "as found" and "as left" results, they would have 6 separate test results, at least, for this meter). You can set up the system to work this way for you.
- The type of information that can be recorded in respect of a test varies widely. Some organizations simply record if the test passed or failed. Other organizations keep track of a great deal of information about the test results. Consider the following examples:
 - Most organizations record who performed the test, the state of the seals on the meter, and whether or not a repair is necessary.
 - A company with gas meters might record the ambient air temperature, the temperature of the oil used in the test, the test spin time, the actual meter reading, the "chart" (should be) reading and the percent accuracy.
 - A company with electric meters might record the following information: meter reading at full capacity, meter reading at light capacity, meter reading with a power factor of 50%, and the percent accuracy.

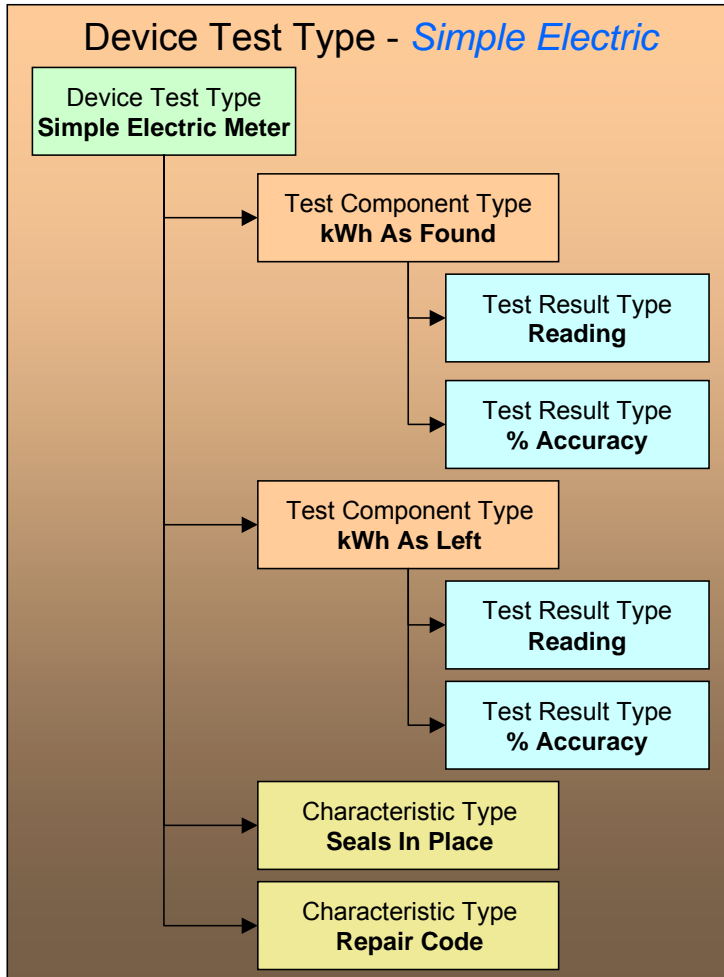
You can set up the system to work this way for you.

Given that the system supports the above disparate requirements, you can understand why the set up process is either straightforward or challenging. The remaining topics in this section provide some guidelines to help you through this setup process.

A Device Test Records Test Results

You create a device test each time you test a meter or item. The device test keeps track of when the test was conducted, who conducted it, and the results of the test. Every device test references a *device test type*.

When you design your device test types, you are actually defining the type of information that can / must be recorded when such a device test is saved in the system. The following picture illustrates a device test type used for tests of simple electric meters.



Notice that the above device test type example uses two different mechanisms to record test results:

- **Component tests.** If your organization maintains the test results from individual registers, you will use component tests. In the above example, two different types of component tests can be recorded for this type of device test: one is used to record a register's accuracy before calibration (the "as found" component test), another is used to record the register's accuracy after calibration (the "as left" component test).

The component test type controls the type of information that is recorded for a component test. In the above example, each component test type requires the same result types – a register reading and a percent accuracy. Note, the fact that these two test types require the same results is coincidental.

You have to set up a component test types and test result types to satisfy your organization record keeping requirements.

- **Characteristics.** If your organization doesn't keep register-specific test results, you don't have to use component tests. Rather, you can simply use characteristics to record test results. In the above device test type, characteristics are used to record whether the seals were in placed and a repair code (if the test results in a repair). Refer to [Setting Up Characteristic Types & Their Values](#) for more information about characteristics.

Field Activities And Device Testing

Most device tests occur as a result of a field activity. The field activity is associated with the service point at which the device is currently installed. Refer to [Examples of Device Testing Activity Types and their Steps](#) for an overview of how these types of field activities look.

You can create device testing field activities manually OR you can take advantage of the [Device Test Selection](#) page. This page will generate field activities to test meters and items based on a user-defined Test Selection Algorithm. The system comes with a sample Test Selection Algorithm type that selects meters of a given manufacturer / model that haven't been test for a given number of months. In all likelihood, you or your implementer will have to develop other algorithms to meet the test selection requirements of your organization. For example, if you test all meters that generate more than \$100,000 of revenue per annum every 6 months, you will have to write a new Test Selection Algorithm Type.

After defining which test selection algorithm types you need, you must set up Test Selection Algorithms that make use of them (these algorithms are specified by the operator on the [Device Test Selection](#) page). To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that is associated with a device test selection. Click [here](#) to see the algorithm types available for this plug-in spot.

Device Test Validation

After entering the test results for a device test, some organizations want the system to analyze the test results and determine if the meter / item passed or failed the test. If your organization does this, you will have to determine the characteristic values and test result values associated with "passing the test". You will then have to design an algorithm that contains logic that determines if a device test contains the appropriate characteristic and test result values.

After developing your device test algorithms, you associate them with the appropriate device test types. The system invokes the respective algorithm when an operator pushes the Validate Device Test button on the device test page.

Setting Up Device Test Options

The topics in this section describe how to set up the control tables that must exist before a device test can be recorded.

Contents[Setting Up Component Test Types](#)[Setting Up Device Test Types](#)

Setting Up Component Test Types

If your organization records component test results, you must set up a component test type for each type of component test. Refer to [A Device Test Records Test Results](#) for more information about component test types.

Open **Admin Menu, Device Test Component Type** to define the type of information that must be recorded on a component test of a given type.

When a new component test type is added. After adding a new component test type, you must define the device test types that make use of it.

Description of Page

Enter an easily recognizable **Test Component Type** and **Description**.

Turn on **Meter Test** if this type of component test is associated with a meter. If this type of component test is associated with an item, this switch should be off.

If **Meter Test** is on, turn on **Register Required** if a register must be referenced on component tests of this type. You would require a register if a) you keep component tests for individual registers, or b) if your component tests require the entry of register readings (as defined in the following grid).

The grid that follows defines the type of test results that are recorded.

Result Seq	Result sequence controls the order in which the test results are captured.
Description	Enter the prompt that is displayed for this type of test result.
Result Required	Turn this switch on if a result must be specified on a test component of this type.
Validation Algorithm	The validation algorithm controls how the test result is validated. Leave this field blank if no validation should be imposed.
Prompt For Read	Turn on this switch if a user can enter a register read for this type of test result.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_TST_COMP_TYP](#).

Setting Up Device Test Types

Every device test references a device test type. The device test type controls business rules associated its device tests. The topics in this section describe how to set up your device test types.

Contents

- [Device Test Type - Main](#)
- [Device Test Type - Component Types](#)
- [Device Test Type - Characteristics](#)

Device Test Type - Main

Open **Admin Menu, Device Test Type** to define general business rules shared by device tests of a given type.

When a new device test type is added. After defining your device test types, you must update your meter and item types to define their valid test types. This is what prevents a gas test from being performed on an electric meter.

Description of Page

Enter an easily recognizable **Test Type** and **Description**.

Turn on **Meter Test** if this type of test is associated with a meter. If this type of test is associated with an item, this switch should be off.

Turn on **Validation Required** if tests of this type use a **Validation Algorithm** to determine if the test results (and therefore the device test) are of a passing grade. If you haven't done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that is associated with a device test validation algorithm. Click [here](#) to see the algorithm types available for this plug-in spot.

Define **Characteristic Types** and their respective **Characteristic Values** to describe characteristics common to all tests of this type. Note that you can also define characteristic types for which values are required to be entered when the device test is created (see [Device Test Type - Characteristics](#)).

Note. You can only choose characteristic types defined as permissible on a device test type record.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_DV_TEST_TYPE](#).

Device Test Type - Component Types

Open **Admin Menu, Device Test Type – Component Types** to define the types of component tests (if any) that can be performed for device tests of a given type.

Description of Page

Use **Sequence** to define the relative order of each **Test Component Type** that can be performed during device tests of a given type.

Device Test Type - Characteristics

Open **Admin Menu, Device Test Type – Characteristics** to define required and optional characteristics for device tests of a given type.

Description of Page

Define **Characteristic Types** to describe characteristics that might be entered for all tests of this type. Turn on **Required** if this type of characteristics must be specified on tests of this type, otherwise, the characteristic will be optional. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on. Use **Sequence** to control the order in which characteristics are defaulted.

Note. You can only choose characteristic types defined as permissible on a device test type record.

The Big Picture Of Usage Requests

Some organizations use a meter data management (MDM) system to record meter reading information. Since meter reads are not available in Oracle Utilities Customer Care and Billing to calculate consumption, during the billing process usage (or bill determinants) must be requested from the MDM system.

The following sections describe how the base-package usage request process works.

Contents

- [Requesting Bill Determinants](#)
- [Usage Request Lifecycle](#)
- [Corrected Read Notifications](#)
- [Configuring The System For Usage Request Integration](#)
- [Designing Your SA Types For Usage Requests](#)
- [Start And End Times For Billing](#)
- [Navigating To MDM](#)

Requesting Bill Determinants

The term bill determinant request is another way to refer to a usage request. These usage requests are created during the batch billing process, and also when a user generates an online bill segment or performs cancel / rebill. The get consumption algorithm specified on the SA Type's Bill Segment Type is responsible for creating the usage request.

Contents

- [Batch Billing Usage Requests](#)
- [Online Billing Usage Requests](#)

Batch Billing Usage Requests

Batch billing usage requests are created and held in the initial pending state. A separate batch process is responsible for transitioning these requests and sending them to the MDM system. Here's how this works:

- The get consumption algorithm specified on the SA Type's Bill Segment Type is responsible for creating the usage request. These batch billing usage requests are held in the **Pending** state until the batch **Usage Scheduled Monitor Process** is executed
- Similarly, once a response is received from MDM, the usage request will not transition to the **Bill Determinants Processed** state until the batch **Usage Scheduled Monitor Process** is executed
- In addition to the standard MDM usage request elements, the system captures the bill cycle and window start date.

Online Billing Usage Requests

- Usage requests created from an online billing request are not held in the initial state. These usage requests are transitioned through their lifecycle resulting in the request being sent to MDM and, once a response is received, the bill segment is regenerated.
- The bill segment remains in the **Freezable** state until the user freezes the bill segment and completes the bill. If however, the freezable bill segment is regenerated, the system cancels the usage request and creates another usage request for the bill segment.
- In addition to the standard MDM usage request elements, the system captures the responsible user id.

External Reference Id. When creating a usage request, the external reference id is populated based on the version of MDM that your implementation integrates with. This is defined as the **MDM Version** MO option on the **Usage** maintenance object. If integrating with MDM 1, the system populates the external reference id using the batch run thread scheduler id to facilitate processing in the external system. Otherwise, the external reference id is left blank and may be updated on the response from the external system.

Usage Request Lifecycle

The usage request business object (BO) contains the rules that govern the processing of a usage request. The base product provides the BO **C1-UsageRequest** which serves as a parent BO and contains the following lifecycle:

- All usage requests are created in the initial **Pending** state. If the usage request was created from the batch billing process, the batch **Usage Scheduled Monitor Process** must be executed to transition the usage request.
- If there are pending sync requests for the usage request's service agreement, the usage request is held in the **Awaiting Data Sync** state until the sync is processed.
- Once all pending sync requests related to the usage request have been processed, the usage request transitions to the **Send Request** state. An enter algorithm on this transitory state is responsible for sending the usage request to MDM. The base algorithm creates a JMS Queue outbound message. This outbound message is configured to not persist on the database.

- Once the outbound message has been successfully sent, the usage request sits in the **Awaiting Bill Determinants** state until a response is received from MDM, or a time out is encountered.
- When a response is received from the external system, the usage request is transitioned to either the **Bill Determinants Received** state, or the **Error** state. In the case of an error, the following occurs:
 - If the usage request was created from the batch billing process and the billing window is still open, a new usage request will be created the next time billing runs and the process is repeated. If however, the billing window is no longer open, a to do entry is created for manual follow up.
 - If the usage request was created from an online billing process, the user will be notified. The bill segment remains in the error state and the user can either delete the bill segment, or attempt to generate it again. If the bill segment is regenerated, a new usage request is created and the process is repeated.
- The system transitions usage requests in the **Bill Determinants Received** state to the **Bill Determinants Processed** state. An enter algorithm on this final state is responsible for generating the bill segment. If the usage request was created from the batch billing process, the bill segment is frozen and the bill is completed. For online billing usage requests, the user is responsible for freezing the bill segment and completing the bill.

Corrected Read Notifications

If a read that was used for bill determinant calculations is modified in the MDM system, notification is sent to Oracle Utilities Customer Care and Billing. This results in the creation of an off cycle bill generator. The system uses the business object defined as the **OCBG Corrected Read BO** MO option on the **Off Cycle Bill Generator** maintenance object to create the OCBG. If any errors are encountered while attempting to create the OCBG, the system sends a message to the external system using the Outbound Message Type and External System defined as BO options on the **OCBG Corrected Read BO**

What happens next depends on the lifecycle that your implementation has configured for the OCBG Corrected Read BO. Here are examples of what might occur:

- Create a to do entry for manual follow up.
- Find any frozen bill segments that might be affected by the corrected read and perform cancel / rebill.

Configuring The System For Usage Request Integration

Oracle Utilities Customer Care and Billing sends usage requests to MDM in the form of an xml message. These messages are transformed by the integration layer and then sent to MDM. Similarly, MDM sends responses to the integration layer so that the data can be transformed and sent to Oracle Utilities Customer Care and Billing.

The following sections describe at a high level the data setup required to send usage requests to an MDM system.

Contents

[Define the Outbound Message Type](#)

Define the XAI Sender

Define the External System and Configure the Messages

Define the Outbound Message Type

An outbound message type is required for the batch billing usage request outbound message. This outbound message type must reference the base **C1-CyclicalUsgReqOutMsg** business object. The outbound message type must also be specified as a BO option on the base **C1-CyclicalUsgReqOutMsg** business object so the system knows which outbound message type to use when sending usage requests to MDM.

An outbound message type is required for the online billing usage request outbound message. This outbound message type must reference the base **C1-NonCyclicalUsgReqOutMsg** business object. The outbound message type must also be specified as a BO option on the base **C1-NonCyclicalUsgReqOutMsg** business object so the system knows which outbound message type to use when sending usage requests to MDM.

Defining the Usage Business Objects. The business objects used by the get consumption algorithm when creating a batch or online billing usage request are defined as MO options on the **Usage** maintenance object.

Define the XAI Sender

An XAI Sender is required to define how to send usage requests to MDM. Use the context of the XAI Sender to define the web service interface.

Define the External System and Configure the Messages

Define an external system and configure the valid outbound message types and the method of communication for each. You will also need to select the appropriate XSLs to format the request for usage. The external system must also be specified as a BO option on the base **C1-CyclicalUsgReqOutMsg** and **C1-NonCyclicalUsgReqOutMsg** business objects so the system knows which external system to use when sending usage requests to MDM.

Refer to the *Oracle Utilities Customer Care and Billing - Meter Data Management Integration Implementation Guide* for more information.

Designing Your SA Types For Usage Requests

SA Types used to bill service agreements that require bill determinants from a meter data management system must have the following characteristics:

- The SA Type must have a special role flag of **Bill Determinants Required**
- Bill segment type:
 - Reference the bill segment creation algorithm that creates a bill segment from a usage request.
 - Reference the bill segment get consumption algorithm that gets bill segment consumption using a usage request.

Start And End Times For Billing

As you know, there is logic in billing to determine the start date and end date for a bill segment. Refer to [Ways to Control The End Date Of A Bill](#) for more information. When billing for a customer with interval data, the system also needs to know the time.

The time used by billing, referred to as the cutoff time, is stored on the service agreement. There is also a control on the service agreement called Start Day Option that determines which day to use for the start time. Billing uses the billing date, the cutoff time, and the start day option to determine the correct interval data to process.

When integrating with a meter data management system, Oracle Utilities Customer Care and Billing lacks knowledge of the type of meter installed at a service point. Thus, both interval processing period as well as scalar processing information is captured on a usage request. MDM then uses the appropriate period to calculate bill determinants based on the type of meter installed.

MDM returns the true usage period used to calculate bill determinants on the usage response. This usage period is captured on the bill segment.

Refer to [Start and End Times for Billing](#) for more information.

Navigating To MDM

It might be necessary for Oracle Utilities Customer Care and Billing users to navigate to MDM to view detailed read information for a service point. For example, a user may want to see the unbilled consumption that's available in MDM. The following is required to implement this:

- Add a menu entry to the service point context menu. This menu item must reference the base ***Go To MDM From SP*** navigation option
- Add a menu entry to the service agreement context menu. This menu item must reference the base ***Go To MDM From SA*** navigation option
- Set up the ***MDM URL*** option type on the ***General System Configuration Feature Configuration***. The ***MDM URL*** option must contain the URL for the MDM application.

Once configured, users should be able to launch the MDM system from these context menus with either a service point or service agreement id in context.

Defining Premise & Service Point Options

A premise is where a customer consumes the services supplied by your company. Located at a premise are the various devices that consume energy and measure consumption. Before you can define premises and devices, you must set up the control tables defined in this section.

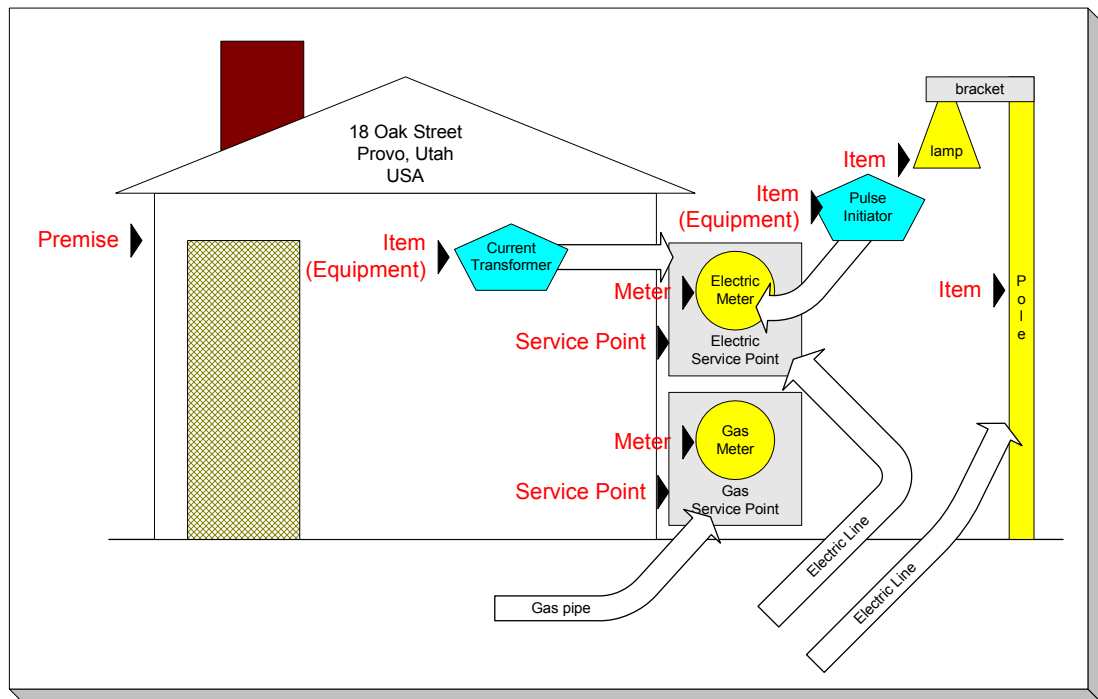
Warning! The topics in this section do not describe every table that must exist in order to set up premises and service points. Many premise and service point control tables are described in [Defining Meter & Item Options](#), and [Defining Field Order Options](#). The tables described in this section are those that must be set up regardless of the type of service.

Contents

- [An Illustration Of A Premise](#)
- [Setting Up Premise Options](#)
- [Setting Up Generic Service Point Options](#)
- [Setting Up Premise & Service Point Postal Defaults](#)
- [Designing SP Types](#)

An Illustration Of A Premise

The following picture illustrates a premise with 2 service points, 2 meters, and 2 badged items:



The following concepts are illustrated above:

Premise

A premise describes a location at which your company supplies some type of service. In addition to the obvious address information, a premise also contains geographic coordinates, meter read instructions, and taxation jurisdiction information.

For more information about the control tables that must be set up before you can define a premise see [Setting Up Premise Options](#).

Service Point

A service point (SP) is a geographic location at which service(s) are delivered to a premise. The SP record maintains information about the type of service, the service cycle (if the service is metered), the field office responsible for maintaining the service, the distribution company that supplies the service, etc.

There are three major categories of service points:

- Those where the rate of consumption and the total amount of consumption is measured (e.g., electricity, gas, water) by a meter. You can think of this type of service point as a “socket” into which a meter can be plugged. Over time, many meters may be plugged into the socket. We refer to these types of service points as metered.
- Those that hold badged items. A badged item is a physical device with a unique identity (e.g., a specific street light, a specific hydrant). You can think of this type of service point as a “socket” into which a badged item can be plugged. Over time, many items may be plugged into the socket. We refer to these types of service points as item-based.
- Those used to hold one or more non-badged items. For example, if your organization doesn’t badge street lamps, you can use a single service point to hold an infinite number of lamps. We refer to these types of service points as non-badged.

Refer to [Service Points \(SPs\)](#) for more information about non-badged items.

An unlimited number of SP’s may exist at a premise. However in reality, the number of SP’s is related to the number of services supplied by your company. For example, an electric and gas company will typically have 2 SP’s per premise.

For more information about the control tables that must be set up before you can define service points refer to [Setting Up Generic Service Point Options](#), [Defining Meter & Item Options](#).

Field activities may be dispatched to all types of service points.

Meter

A meter is a physical device used to measure the amount of gas, water, or electricity used by a customer. While most meters measure consumption in a single unit of measure (e.g., gallons, cubic feet, kilowatt-hours), some electric meters are extremely sophisticated and measure several different factors. For example, some electric meters measure how much was used, when it was used, the efficiency of consumption, the maximum amount used, and a few other unusual things.

For more information about the control tables that must be set up before you can define a meter, refer to [Setting Up Meter Options](#), and [Setting Up Consumption Estimation Parameters](#).

Item (Equipment)

An item that is considered to be “equipment” is a physical device that regulates consumption; it does NOT measure consumption. You would only define equipment if it is of interest to your organization. For example, if your organization periodically tests the pulse initiators associated with your meters, you will need to set up items for each pulse initiator and link them to their respective meters. Equipment can be linked to either a service point (e.g., a current transformer, a backflow device), a meter (e.g., a pulse initiator), or an item (e.g., the components of an installation).

Equipment and billing. Be aware that the only way equipment can impact billing would be if you developed a service quantity or register rule that analyzed the equipment associated with a service point (directly or indirectly via the meters and items) and manipulated billed consumption accordingly. Refer to [Defining Measured Quantity Manipulation Rules](#) for more information.

Item (NOT Equipment)

An item that is not consider to be “equipment” is a physical device that does NOT measure consumption, but impacts billing in some way (i.e., there are charges in your rates based on the number and type of items installed at a service point). Examples include street lights, light poles, and security cameras. Items are related to service points and a service point can have one or more items linked to it.

For more information about the control tables that must be set up before you can define an item, refer to [Setting Up Items](#).

For more information about premises and service points, refer to [Understanding The “V”](#).

Setting Up Premise Options

This section describes tables that must be set up before you can define premises.

Defining Premise Types

Open **Admin Menu, Premise Type** to define the premise types used to categorize your premises.

Description of Page

Enter a unique **Premise Type** and a **Description** for every premise type.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_PREM_TYPE](#).

Setting Up Generic Service Point Options

This section describes tables that must be set up before you can define service points.

Contents

[Facility Levels](#)

[Setting Up Service Point Types](#)

Facility Levels

Every type of service tends to use a different mapping philosophy to designate the facility hierarchy that supplies service to the service point. For example, electric service typically uses a substation / feeder / node facility hierarchy to define how electricity is supplied to a service point (the substation is the highest level in the hierarchy, the feeder comes next, and finally the node). Whereas gas service uses a city gate / main / feeder hierarchy.

If your organization maintains this type of information on service points, you will set up your facilities and their interrelationships using 3 windows. On the first you set up the number and type of facility levels used for every service and you define the valid values for each facility level (you define these when you define your Service Types). On the second and third you define the values that may coexist in each level. After these set up tasks are complete, you're ready to enter facility levels on your service points.

Note. A service point's facility levels are used to help pinpoint problems and dispatch service crews during outages.

Contents

[Setting Up Facility Levels 1 & 2 Combinations](#)

[Setting Up Facility Levels 2 & 3 Combinations](#)

Setting Up Facility Levels 1 & 2 Combinations

To define which values of facility level 2 may be used with a given value of facility level 1, open **Admin Menu, Facility Level 1 to 2**.

Description of Page

Choose a **Service Type** and **Facility Level**, then use the **Facility Level 2** collection to define the level 2 facility levels that may coexist with the selected level 1 facility level.

Setting Up Facility Levels 2 & 3 Combinations

To define which values of facility level 3 may be used with a given value of facility level 2, open **Admin Menu, Facility Level 2 to 3**.

Description of Page

Choose a **Service Type** and **Facility Level 2**, then use the **Facility Level 3** collection to define the level 3 facility levels that may coexist with the selected level 2 facility level.

Setting Up Service Point Types

Every service point must reference a service point (SP) type. The SP type controls almost all aspects of the service point's behavior (e.g., the type of field activity that may be dispatched to it, the type of service agreement that may be linked to it, the type of meter that may be installed at it).

The topics in this section describe the windows used to set up your SP types.

Warning! Setting up SP types requires careful analysis of your company's SA types, field activities, and its consumption estimation philosophy. Refer to [Designing SP Types](#) for more information about this design process.

Contents

- [SP Type - Main](#)
- [SP Type - Meter Type](#)
- [SP Type - SA Type](#)
- [SP Type - Item Type](#)
- [SP Type - SP Characteristic](#)
- [SP Type - Equipment Type](#)
- [SP Type - FA Type Review](#)

SP Type - Main

You begin to define a service point type using the **Admin Menu, SP Type** window.

Warning! You may find that your desire to use some of the more sophisticated control functions in the system will necessitate many SP types. For example, notice that one of the other windows on this group allows you to define the meter types that can be installed in service points of a given type. If you have many different types of meters and many restrictions as to the types of service points in which they can be installed, you will end up with many SP types.

Description of Page

Enter an easily recognizable **SP Type** for the service point type. This value will appear on many windows throughout your system as a “shorthand” for how the SP is used, so think carefully about the format. We recommend using a consistent format for all of your SP types. You should include the service type, an indication if it’s metered or item-based, and the trend classification. For example, you might enter “E-M-RES”, for an electric, metered, residential SP type.

Select the **Service Type**. Refer to [Setting Up Service Types](#) for more information.

Enter a **Description**.

Select the **Sub Type** to indicate the type of device that may be installed at service points of this type: **Meter**, **Item**, **Unbadged**.

Turn on the **Allow Service Route** switch if service points of this type should have a route. This switch would typically be turned on for metered service points.

Multiple Route Usage controls whether a service point can reference multiple service routes. Select **Allowed** if multiple service routes are allowed on service points of this type. Select **Not Allowed** if multiple service routes are not allowed on service points of this type. This switch would typically be turned on for garbage service points.

If this SP type is for metered service, select the **Trend Class** to define how to categorize this SP type’s consumption for estimation and high / low validation purposes. This field will be gray for item-based services because items don’t have meter reads that can be estimated or that are subject to high / low validation. Refer to [Setting Up Trend Classes](#) for more information.

If field activities are created for this SP type’s service point, select the **Field Activity Type Profile** that controls which type of field activities may be linked to the service points. This Profile will also control which field activities are automatically generated by the system under various circumstances. Refer to [Setting Up Field Activity Type Profiles](#) for more information.

If a geographic type is used to identify individual service points of this SP type, select the **Identifying Geographic Type** used. You typically have an identifying geographic type if you refer to a service point using an identifier that is assigned by a third party. Refer to [Defining Geographic Types](#) for information on setting up geographic types.

Use the **SP Type Characteristic** collection and their respective **Characteristic Values** to describe characteristics common to all service points of this type.

Note. You can only choose characteristic types defined as permissible on an SP Type record. Refer to [Setting Up Characteristic Types & Their Values](#) for more information.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_SP_TYPE](#).

SP Type - Meter Type

Open **Admin Menu**, **SP Type** and navigate to the **Meter Types** tab to define the types of meters that may be installed at service points belonging to this SP type.

Description of Page

The **Meter Types** collection contains the types of meters that may be installed at service points belonging to this SP type. This collection is not relevant if the SP sub type is **Item** or **Unbadged**.

Note that you can connect meters with different service types to your SP Type. For example, a metering device such as a recorder can be used on both gas and electric service points.

SP Type - SA Type

Open **Admin Menu, SP Type** and navigate to the **SA Types** tab to define the SA types that may be linked to (and therefore pay for) service points belonging to this SP type.

Description of Page

The following fields display:

CIS Division / SA Type	Indicate the type of service agreement that may be linked to service points of this type.
Initial	Turn on this switch if the Start Service process should default this SA type when service is initially started at a service point of this type. Multiple SA types may be marked as Initial if you want multiple service agreements created when service is initially started. For example, if you have both wastewater and water service agreements linked to the water service point.

SP Type - Item Type

Open **Admin Menu, SP Type** and navigate to the **Item Types** tab to define the types of items that may be installed at service points belonging to this SP type.

Description of Page

The **Item Types** collection contains the types of items that may be installed at service points belonging to this SP type. This collection is not relevant if the SP sub type is **Meter** or **Unbadged**.

Note that you can connect items with different service types to your SP Type. For example, an item like a cell phone can be used on both gas and electric service points.

SP Type - SP Characteristic

To define characteristics that may be defined for service points of a given type, open **Admin Menu, SP Type** and navigate to the **SP Characteristic** tab.

Description of Page

Use the **SP Characteristics** collection to define characteristics that can be defined for service points of a given type. Turn on the **Required** switch if the **Characteristic Type** must be defined on service points of a given type. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on. Use **Sequence** to control the order in which characteristics are defaulted.

SP Type - Equipment Type

Open **Admin Menu**, **SP Type** and navigate to the **Equipment Types** tab to define the types of equipment that can be linked to service points of a given type.

Description of Page

Use the collection to define the item types of **Equipment** that can be linked to service points of this type.

Item types are being specified. There is no equipment type control table. Rather, items are used to define equipment and therefore you are actually defining item types rather than equipment types. Refer to [Equipment versus Badged Items](#) for more information about equipment.

SP Type - FA Type Review

Open **Admin Menu**, **SP Type** and navigate to the **FA Type Review** window to review the field activities that are allowed for the SP type.

Four dimensions. For every **field activity type** that is eligible for dispatch, you define the **dispatch group** that performs the activity at every **SP type** located in every **operations area**. This information is maintained on the field service control page. This is a rather complex relationship because it involves the four dimensions highlighted in bold. Due to this complexity, we have provided review trees on the SP Type, Dispatch Group, and Field Activity Type windows to help you understand what you've set up.

Description of Page

This window is dedicated to a tree that shows the field activities allowed for this SP type. For each field activity type that is eligible for dispatch, you can view the dispatch group that will perform the activity at every operations area.

Setting Up Premise & Service Point Postal Defaults

You set up postal defaults if your company is able to default field values onto new premises and service points based on the premise's postal code. The topics in this section describe how to maintain postal defaults.

For more information about where these default values are used, refer to [Maintaining Premises](#) and [Maintaining Service Points](#).

Contents

- [Postal Defaults - Main](#)
- [Postal Defaults - Service Default](#)

Postal Defaults - Main

To define premise postal defaults, open **Admin Menu, Postal Code Default** and navigate to the **Main** tab.

Description of Page

Enter the **Country Code** and range of postal codes to which the default values apply using the **From Postal Code** and **To Postal Code**.

Note. You may not have postal defaults whose from / to postal codes overlap.

Select the **Trend Area** to be defaulted onto new premises located in this postal code range. The trend area is used to categorize premises into geographic areas when consumption estimation is controlled, in part, by where the customer lives. Refer to [Setting Up Trend Areas](#) for more information.

Enter the **County** to be defaulted onto new premises located in this postal code range.

Enter the **City** to be defaulted onto new premises located in this postal code range.

Enter the **CIS Division** to be defaulted onto new premises located in this postal code range.

Enter the **State** to be defaulted onto new premises located in this postal code range.

Enter the **Time Zone** to be defaulted onto new premises located in this postal code range.

Use the **Characteristic Types and Values** collection to define the **Characteristic Types** and their respective **Characteristic Values** to be defaulted on premises located in this postal code range. In addition to providing interesting information, these characteristics may also determine the prices and tax rates on the bills generated for the services consumed at a premise.

For more information about characteristics, see [Setting Up Characteristic Types & Their Values](#) and [An Illustration Of A Bill Factor And Its Characteristics](#).

Use the **Geographic Types and Values** collection to define the **Geographic Types** and their respective **Values** to be defaulted on premises located in this postal code range. In addition to providing interesting information, these values may be used to sort field activities in geographic value order.

Where Used

This information is defaulted when a new Premise is added. Characteristics and geographic values are also defaulted when the postal code for a Premise is changed. Refer to [Maintaining Premises](#) for more information.

Postal Defaults - Service Default

To define values to be defaulted for a service point located in a postal range, open **Admin Menu, Postal Code Default** and navigate to the **Service Defaults** tab.

For more information about where these default values are used, refer to [Maintaining Service Points](#).

Description of Page

Use the **Service Defaults** collection to define values to be defaulted on service points located in a given postal code range belonging to a given **Service Type** (note, a service point's Service Type comes from its SP Type).

The following values may be defined per **Service Type**:

- When a meter or item is removed from a service point located in this postal code range, the system assigns it by default to **Stock Location**. Note that this is only used if the meter's meter type or item's item type indicates that stock locations are tracked.

For more information about location history, refer to [Stock Location History Is Created Behind The Scenes](#).

- Use the **Field Services** collection to define the **Field Service Classifications** and their respective **Operation Area** to be defaulted on new service points located in this postal range. Refer to [Designing Field Service Classifications](#) for more information.
- Use the **Characteristic Types and Values** collection to define the **Characteristic Types** and their respective **Characteristic Values** to be defaulted on new service points located in this postal code range. In addition to providing interesting information, these characteristics may also determine the prices and tax rates on the bills generated for the services consumed at a service point.
- Use the **Geographic Types and Values** collection to define the **Geographic Types** and their respective **Values** to be defaulted on new service points located in this postal code range. In addition to providing interesting information, these values may be used to sort field activities in geographic value order.

Where Used

Information is defaulted when a new Service Point is added to a Premise. Refer to [Service Point – Main Information](#) for more information.

Stock Location is defaulted when a Meter or an Item is removed from a Service Point. [Refer to SP/Meter Installation](#) and [SP/Item Installation](#).

Designing SP Types

Every service point must reference an SP type. When you set up an SP type, you define how the system manages many aspects of its service points' behavior.

Perfect foresight. In a perfect world, the other control tables would have been set up with perfect foresight of setting up your SP types. In reality, setting up your SP types may invalidate some of your earlier decisions. Don't feel bad if this happens, some amount of iteration is natural.

Designing your SP types is an iterative process. To minimize the number of iterations, we recommend using the steps outlined in this section to complete the following table. When the table is complete, you're ready to set up your SP types.

SP Type	Service Type

Contents

[Service Segmentation](#)
[Device Segmentation](#)
[Meter Read Estimation Trend Class Segmentation](#)
[Field Activity Type Profile Segmentation](#)
[SA Type Segmentation](#)
[Meter Type Segmentation](#)
[Item Type Segmentation](#)

Service Segmentation

At a minimum, you will have one SP type for every different type of service that exists at your premises. If we assume that your organization sells gas, water, waste water and electricity services, you will need four SP types.

SP Type	Service Type
GAS	Gas service
WATER	Water service
WASTE WATER	Waste water service
ELECTRIC	Electric service

Device Segmentation

For each service, determine if there exist meters, badged items, or non-badged items at the service points. For example, if we assume:

- Electric service has service points with meters, badged lamps, and parking lots.
- Gas and water services just use meters.
- Waste water service doesn't actually have a meter installed at it (it uses the water consumption measured by the water service point's meter)

Then your SP types will be:

SP Type	Service Type	SP Sub Type
GAS – METERED	Gas service	Meter
WATER – METERED	Water service	Meter
WASTE WATER	Waste water service	Unbadged
ELECTRIC – METERED	Electric service	Meter
ELECTRIC – BADGED LAMP	Electric service	Item
ELECTRIC – PARKING LOT	Electric service	Unbadged

Meter Read Estimation Trend Class Segmentation

When you set up a metered SP type, you must define the consumption estimation trend class in which its consumption will be categorized. This categorization matters when consumption differs based on the type of property AND you want the system to estimate consumption in different classes differently.

These categories could be the classic divisions of residential versus commercial versus industrial consumers. Alternatively, they could be finer-grained divisions: single family residence, versus duplex, versus triplex, versus medical office, versus grocery store.

Refer to [The Theory Behind Consumption Estimation](#) for more information about how trend class is used to estimate consumption.

If we assume you differentiate between residential and commercial service for all of your metered services, then your SP types will be:

SP Type	Service Type	SP Sub Type	Trend Class
GAS – METERED – RESID	Gas service	Meter	Residential
GAS – METERED – COMM	Gas service	Meter	Commercial
WATER – METERED – RESID	Water service	Meter	Residential
WATER – METERED – COMM	Water service	Meter	Commercial
WASTE WATER – RESID	Waste water service	Unbadged	N/A
WASTE WATER – COMM	Waste water service	Unbadged	N/A
ELECTRIC – METERED – RESID	Electric service	Meter	Residential
ELECTRIC – METERED – COMM	Electric service	Meter	Commercial
ELECTRIC – BADGED LAMP	Electric service	Item	N/A
ELECTRIC – PARKING LOT	Electric service	Unbadged	N/A

Notice that the non-metered service points don't use a trend class. This is because they don't have meters (and only meters have estimated consumption).

Field Activity Type Profile Segmentation

When you set up any type of service point that can have field activities, you must define the field activity type profile. This profile defines:

- The type of field activities that may be dispatched to the service points.
- The type of field activity to be defaulted by the start / stop process given the condition of the service point.

Field activity type profiles should not impact your SP type design as these profiles should be designed after the SP types are designed.

Refer to [Designing Your Field Activity Profiles & Types](#) for more information about how field activity profiles are used.

SA Type Segmentation

Every SP Type whose service points can be linked to a service agreement has one or more SA types. These define which type of service agreements can pay for the service point's service. If different service points have different valid SA types, you will need to split the SP types accordingly.

The SA Type segmentation of SP Types is the most complicated design decision you'll have to make. Unfortunately, the decision making process is subjective and iterative. The iterations are caused by the fact that the number of SA types is dependent on the number of SP types (and vice versa). We recommend the following to help work your way through this conundrum:

- Design your SA types using the information in [Defining Service Agreement Types](#).
- Return to your SP types and determine if, given the proposed SA types, you can define a list of valid SA types for each SP type. If you find the population of SA types (and their valid rates) could result in invalid rates paying for service at a service point, divide your SA types further.

Meter Type Segmentation

Every metered SP Type has one or more meter types. These define which type of meter can be installed at a service point of a given type. If different service points have different valid meter types, you will need to split the SP types accordingly.

For example, if you have 3-phase electric service points and you want to make sure that only 3-phase meters are installed in these service points, you will need to split the electric SP types accordingly.

Refer to [Setting Up Meter Types](#) for more information.

Item Type Segmentation

All SP Types may have item types. These define which type of item can be installed at a service point of a given type. If different service points have different valid item types, you will need to split the SP types accordingly.

Refer to [Setting Up Item Types](#) for more information.

Defining Cycles and Schedules

This chapter is dedicated to issues related to defining cycles, routes and schedules in the system.

Contents

- [Defining Bill & Service Cycles](#)
- [Defining Statement Cycles](#)

Defining Bill & Service Cycles

Every account references a bill cycle. An account's bill cycle controls when it is billed.

Every metered service point references a service cycle. A service point's service cycle controls when a service point's meter is read.

The design of your meter read and bill cycles is inextricably linked because you probably want to bill your customers shortly after their meters are read.

In this section, we describe how to design and set up these cycles. In addition, we discuss how to set up bill period schedules. These are used to define the bill segment end date for special types of non-metered service agreements.

Recommendation. We recommend reading [Bill Frequency – Bill Cycle vs Bill Segment Duration](#) before setting up this information.

Contents

- [The Big Picture Of Bill Cycles, Service Cycles and Bill Periods](#)
- [Designing Cycles for Waste Collection Services](#)
- [Setting Up Bill Cycles](#)
- [Setting Up Service Route Types](#)
- [Setting Up Service Cycles And Routes](#)
- [Setting Up Service Cycle Schedules](#)
- [Setting Up Bill Periods](#)

The Big Picture Of Bill Cycles, Service Cycles and Bill Periods

The topics in this section provide background information about a variety of bill cycle, service cycle, and bill period issues.

Designing Cycles for Metered Services

The topics in this section provide background information about a variety of service cycle issues.

Contents

- [A Description Of The Cyclical Meter Read Process](#)
- [Designing Service Cycles, Routes, And Schedules](#)

A Description Of The Cyclical Meter Read Process

Meter readers using handheld devices record most meter reads in the field. These meter reads are uploaded into the system for use by billing.

A service point's meter is read due to the following data relationships:

- Every metered service point references a service route (henceforth called "route").
- A route references a service cycle.
- A service cycle has service schedules that define when the service points in the cycle are read.
- Every service cycle schedule contains two dates:
 - **Scheduled selection date.** This is the date the system selects the service points in the cycle for download to your handheld software.
 - **Scheduled work date.** This is the date the meter is expected to be read.

Overriding a route within a specific schedule. Rather than downloading all routes within a cycle, you can set up the system so that only specific routes are downloaded on any given date. This is a very powerful feature. You can use it, for example, to estimate specific cycles every other month or indicate the customer reads the meter every third month. Refer to [Designing Service Cycles, Routes, And Schedules](#) for more information.

- On the scheduled selection date, the system creates a download file containing information to be sent to your handheld software. This download file contains information about every register on every meter in the routes being downloaded.
- Your handheld software distributes this information to the handheld devices and then your meter readers do their job. When they finish, the resultant meter reads are uploaded into the system for subsequent use by billing.

Warning! It is very important to create a service cycle schedule for every expected read date regardless of whether the cycle's routes are downloaded. Why do you have to do this? Because billing uses the scheduled work date on the service cycle to know when to look for a reading. If it can't find a reading on or near this date, billing estimates consumption (given estimation is allowed on the service agreement). Without a service cycle schedule, billing wouldn't know when to look for readings. So, for example, if you estimate a given cycle's consumption EVERY OTHER MONTH, you must create a service cycle schedule for EVERY month. On each month's schedule, you must define if the routes should be downloaded.

For more information about the how to control when a cycle is read, refer to [Setting Up Service Cycle Schedules](#).

Designing Service Cycles, Routes, And Schedules

The topics in this section provide information describing how to design your service cycles, routes, and schedules.

Contents

[Designing Service Cycles For Meter Reading](#)

Designing Service Routes For Meter Reading

Designing Metered Service Cycle Schedules

Designing Service Cycles For Meter Reading

The criterion that affects the number of service cycles has nothing to do with when meter readers physically read your meters. Rather, the frequency that you bill the meter's consumption (real or estimated) is what controls the number of service cycles.

So, for example, if you bill every month, but read every OTHER month, you'll have 20 service cycles – one for each bill day during the month. If you bill bimonthly, you'll have 40 service cycles. If you bill quarterly, you'll have 60 service cycles. Etc.

Different billing frequencies are possible for different service points. If you have different billing frequencies for your different types of metered service, you'll need a different set of cycles for each billing frequency. For example, if you bill water quarterly and electricity monthly, you'll have 20 bill cycles (one for each bill day during a month), but you'd need 60 quarterly service cycles for your water service points, and 20 monthly service cycles for your electric service points. This would result in a customer getting billed every month. However, four times a year, their bill would contain a charge for both electricity and water.

Designing Service Routes For Meter Reading

The following points describe the relationship between a meter read, a route and a service cycle:

- A service cycle contains one or more routes.
- A route has one or more service points.
- A service point holds a meter.
- And a meter is what is read.
- Therefore, the number of meters a person can read in a day limits the number of service points in a route.

Warning! If your company supplies electric service and uses MV90's, you will need to take advantage of "notional" routes. A "notional" route's service points are never actually read by a human. Rather, the service points' consumption is fed to the system by a sophisticated device (e.g., an MV90). These service points still need to be linked to a route because billing is dependent on the route's cycle to determine the expected meter read date.

Designing Metered Service Cycle Schedules

The process of designing your service schedules is either easy or complicated. It will be easy if all routes within a cycle are downloaded when the service cycle is scheduled for download. It will be complicated if you download a subset of routes within a cycle on any given download date. We'll provide a few examples to help explain why.

If you download all cycles in a route whenever the cycle is downloaded, your service cycle schedule will look as follows.

Bill cycles. We've included each service cycle's related bill cycle to help you understand when the service cycle's consumption will be billed. Bill cycles are discussed in [Bill Cycles](#).

Service Cycle	Download Date	Sched MR Date	Which Routes To Download	Bill Cycle	Bill Window	Estimation Date
1	28-May-99	31-May-99	All	1	31-May-99 to 2-Jun-99	2-Jun-99
2	31-May-99	1-Jun-99	All	2	1-Jun-99 to 3-Jun-99	3-Jun-99
3	1-Jun-99	2-Jun-99	All	3	2-Jun-99 to 4-Jun-99	4-Jun-99
4	2-Jun-99	3-Jun-99	All	4	3-Jun-99 to 7-Jun-99	7-Jun-99
Etc. to 20						

Now let's complicate life. If we assume you physically read your routes every other month (but bill monthly using estimated consumption), then you'll need the following service schedule.

Service Cycle	Download Date	Sched MR Date	Which Routes	Bill Cycle	Bill Window	Estimation Date
1	30-May-99	31-May-99	1, 2, 3 – Download 4, 5, 6 - Estimate	1	31-May-99 to 2-Jun-99	2-Jun-99
2	31-May-99	1-Jun-99	1, 2, 3 – Download 4, 5, 6 - Estimate	2	1-Jun-99 to 3-Jun-99	3-Jun-99
3	1-Jun-99	2-Jun-99	1, 2, 3 – Download 4, 5, 6 - Estimate	3	2-Jun-99 to 4-Jun-99	4-Jun-99
4	2-Jun-99	3-Jun-99	1, 2, 3 – Download 4, 5, 6 - Estimate	4	3-Jun-99 to 7-Jun-99	7-Jun-99
Etc. to 20						
1	29-Jun-99	30-Jun-99	1, 2, 3 – Estimate 4, 5, 6 – Download	1	30-Jun-99 to 2-Jul-99	2-July-99
2	30-Jun-99	1-July-99	1, 2, 3 – Estimate 4, 5, 6 – Download	2	1-July-99 to 3-Jul-99	3-July-99
3	1-July-99	2-July-99	1, 2, 3 – Estimate 4, 5, 6 – Download	3	2-July-99 to 4-July-99	4-July-99
4	2-July-99	3-July-99	1, 2, 3 – Estimate 4, 5, 6 – Download	4	3-July-99 to 7-July-99	7-July-99
Etc. to 20						

Notice the following:

- You still have 20 service cycles even though you only read the meters every other month. Why? Because billing uses the scheduled read date on the service cycle to know when to look for a reading. If it can't find a reading on this date, billing estimates consumption (given estimation is allowed on the service agreement). Without a service cycle schedule, billing wouldn't know when to look for readings.
- Every other month you download half the routes in each cycle and estimate consumption for the other half.
- If you don't download all routes when a service cycle is scheduled, you have to indicate how to handle every route in the cycle.

The above design is infinitely flexible. You can use it to handle any number of requirements:

- Estimate consumption every 7th month.
- Bill every month, but only read once a quarter.
- Etc.

Designing Cycles for Waste Collection Services

The topics in this section describe how to design service cycles to support your waste collection requirements.

Contents

[A Description Of The Cyclical Waste Collection Process](#)
[Bill Cycles](#)
[Designing Bill Periods](#)

A Description Of The Cyclical Waste Collection Process

Waste collection equipment (e.g., garbage trucks) travel from service point to service point using a predefined schedule. This schedule defines which groups of service points are collected by a given truck on a given day (we use the term “route” to define a group of service points that are collected by a given truck on a given day).

Waste service points tend to belong to multiple routes because they can be collected in a different order depending on a wide variety of factors:

- A service point may have one truck responsible for collecting recycling and a different truck responsible for collecting non-recyclable waste.
- A service point that is collected on Mondays, Wednesdays and Fridays may be on a different route on each day of the week.
- Etc.

Contrary to metered service points, the service schedules linked to your waste collection service cycles / route groups do not affect billing. Why? Because billing doesn't need to look for a read for these types of service points and therefore it doesn't need a schedule to know when to look for a read.

Bill Cycles

The topics in this section provide background information about a variety of bill cycle features.

Contents

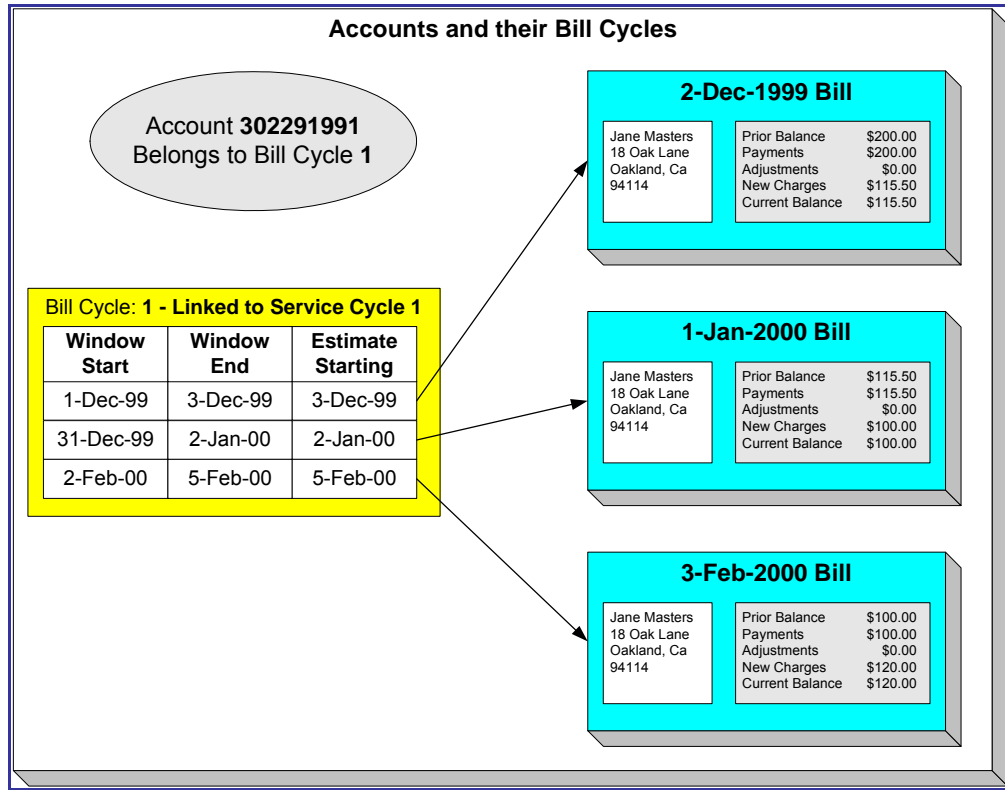
[The Cyclical Billing Process & Window Billing](#)
[Designing Your Bill Cycle](#)
[The Relationship Between Metered Service Cycles and Bill Cycles](#)
[How Does An Account Get Its Bill Cycle?](#)
[Protecting An Account's Bill Cycle](#)
[What Happens If An Account Has Service Agreements Spanning Metered Service Cycles?](#)

The Cyclical Billing Process & Window Billing

The cyclical bill creation process creates most bills. This process works as follows:

- Every account references a bill cycle. The bill cycle's schedule controls WHEN the system attempts to create bills for the account.
- Every bill cycle has a bill cycle schedule that defines the dates when a cycle's accounts are to be billed. Rather than attempt to create bills on one evening, most bill cycles use a concept of "window billing" where the system attempts to produce bills for accounts over a few nights. The first night (i.e., the day the window opens) should be the earliest day on which meter reads for the account can enter the system. The last night (i.e., the day the window closes) should be the last possible day in which a meter read can enter the system. This concept of window billing allows you to start billing accounts on the earliest possible day and then bill the stragglers over successive evenings. This results in much better cash flow.
- When the bill cycle creation process runs, it looks for bill cycles with open bill windows. It then attempts to create bills for the accounts in each such cycle. If a bill is created, it will send it out that evening. If a bill cannot be created, the system will create a bill in the "error" state with a message that can be analyzed by your billing staff. The next day, your staff can either correct the problem or not. When the bill cycle creation process next runs, it deletes all "error" bills and attempts to recreate them. It continues this throughout the bill window. If bills are in error at the end of the window, they will remain in this state until a user fixes them. If the bills are still in error when the cycle's next window opens, a billing error will be generated.

The following diagram should help clarify the above.



For more information about the how to control when bills are produced for a cycle, refer to [Bill Cycle - Bill Cycle Schedule](#).

Designing Your Bill Cycle

The number of bill cycles is determined by the frequency that you bill your customers. So, for example, if you bill every month, you'll have 20 bill cycles – one for each bill day during the month. If you bill bimonthly, you'll have 40 bill cycles. If you bill quarterly, you'll have 60 bill cycles. Etc.

Keep in mind the following:

- You may need additional bill cycles if you allow customers to be billed off-cycle. For example, you could create a bill cycle called "Seniors" and link this to every senior citizen. You would set up this bill cycle's schedule to create bills shortly after social security checks are issued.
- You may need other bill cycles for customers with non-metered services (e.g., land leases, one time invoices).

Important! An account's bill cycle should attempt to create bill segments at least as often as the shortest service agreement duration. For example, if an account has both monthly and quarterly service agreements, the account should be placed on a monthly bill cycle. Refer to [Bill Frequency – Bill Cycle vs Bill Segment Duration](#) for more information.

The Relationship Between Metered Service Cycles and Bill Cycles

As you can deduce, a service point's service cycle is related to an account's bill cycle.

The following table is an example of how you would set up the various dates on the various schedules that control meter reading and billing.

Service Cycle	Download Date	Sched MR Date	Bill Cycle	Bill Window	Estimation Date
1	28-May-99	31-May-99	1	31-May-99 to 2-Jun-99	2-Jun-99
2	31-May-99	1-Jun-99	2	1-Jun-99 to 3-Jun-99	3-Jun-99
3	1-Jun-99	2-Jun-99	3	2-Jun-99 to 4-Jun-99	4-Jun-99
4	2-Jun-99	3-Jun-99	4	3-Jun-99 to 7-Jun-99	7-Jun-99
Etc.					

Notice the following about this example:

- The bill cycle code is the same as the meter read (MR) cycle. This is not necessary, it's just good practice.
- The bill window starts on the first date on which a meter read could be uploaded.
- The bill window ends on the day after the last possible day a read could be uploaded for the bill cycle.
- Billing is only allowed to estimate consumption on the last day of the bill window.

For more information about the bill cycle schedule, refer to [Bill Cycle - Bill Cycle Schedule](#). For more information about the service cycle schedule, refer to [Setting Up Service Cycle Schedules](#).

How Does An Account Get Its Bill Cycle?

Most accounts are created behind-the-scenes when a user uses the “add account” option on [Person – Main Information](#). An account created like this doesn't have a bill cycle. Rather, it sits in limbo awaiting the activation of its first service agreement. When a service agreement is activated, the system populates the account's bill cycle using the following algorithm:

- If an account's bill cycle is protected, the activation of a service agreement will not affect an account's bill cycle. Refer to [Protecting An Account's Bill Cycle](#) for more information.
- If the service agreement being activated is for metered service, the account is given a bill cycle that will generate bills shortly after the service point is read. The route the system follows to get this bill cycle is a bit indirect:
 - A metered service agreement references one or more metered service points.
 - Every metered service point references a service cycle (the service cycle controls when the meter at the service point is read).
 - Every service cycle references a default bill cycle. It is this bill cycle that is populated on the account paying for service at the service point.

- If the service agreement being activated is not metered, the system cannot populate the account's bill cycle because there is no service cycle from which to default the bill cycle. This means the account's bill cycle will be blank until a user specifies a bill cycle for the account (using the Account page).

A To Do entry highlights accounts without a bill cycle. A To Do entry highlights those accounts that require a user to specify a bill cycle. This entry is generated for accounts without a bill cycle that have active service agreements.

When the last service agreement linked to an account is **stopped**, the account's bill cycle will be changed so that the account will be final billed when billing next executes. Refer to [What Happens At Finalization Time?](#) for more information.

When a service point's service cycle is updated, and the account's bill cycle is not protected, the system automatically updates the account's bill cycle to be in sync with the service cycle. Note that this will only take place if the **Allow Bill Cycle Synchronization Option Type** on the **General System Options Feature Configuration** is set to **Y**.

Protecting An Account's Bill Cycle

Over time, as a customer moves from premise to premise, their bill cycle is changed behind-the-scenes to be in sync with the latest service point's service cycle (as described in the previous section). If you do not want the system to change an account's bill cycle when a metered service agreement is activated, you need to turn on the account's **protect bill cycle** flag. You would do this if a customer liked to be billed at the start of the month regardless of when their meter was read.

When the last service agreement for an account is stopped, the protect bill cycle flag is reset. This is to ensure that if the customer returns to start new service again, the bill cycle can be set based on [How Does An Account Get Its Bill Cycle](#).

What Happens If An Account Has Service Agreements Spanning Metered Service Cycles?

A single account can have service agreements that are in several service cycles. The bill cycle on such an account will default based on the last activated metered service agreement.

It's important to be aware that an account will only have bills created when its bill cycle schedule so indicates. This means that the consumption for some service points could remain unbilled for a few weeks.

Designing Bill Periods

Bill periods are used by non-metered service agreements whose bill end dates need to fall on strict dates. You need only set up this information if you have this type of service agreement.

Refer to [Ways To Control The End Date Of A Bill Segment](#) for more information.

Every bill period has a calendar that defines when bill segments are created for service agreements that use the bill period. Examples of bill periods include:

- Quarterly Bill - Last Day Of Quarter
- Quarterly Future Bill - Last Day Of Quarter

- Monthly - 15th Day Of Month.
- Monthly Future Bill - Last Day Of Month.

The Quarterly Bill – Last Day Of Quarter bill period would have a schedule that looked as follows:

Earliest Date On Which To Create A Bill Segment	Bill End Date
1-Oct-1998	30-Sep-1998
1-Jan-1999	31-Dec-1998
1-Apr-1999	31-Mar-1999
1-Jul-1999	30-Jun-1999
...	

The Quarterly Future Bill – Last Day Of Quarter bill period would have a schedule that looked as follows:

Earliest Date On Which To Create A Bill Segment	Bill End Date
15-Dec-1998	31-Mar-1999
15-Mar-1999	30-Jun-1999
...	

The remainder of this section provides examples using the above calendars.

The following example assumes the following:

- The service agreement starts on 18-Dec-1998.
- The service agreement's SA type references the Quarterly Future Bill - Last Day Of Quarter bill period.

The following table shows when bill segments will be produced (assuming the account's bill cycle attempt to create segments as soon as possible) for several bill periods:

Earliest Date Segment Will Be Produced	Bill Period
18-Dec-1998	18-Dec-1998 thru 31-Mar-1999
15-Mar-1999	1-Apr-1999 thru 30-Jun-1999
...	

The following example assumes the following:

- The service agreement starts on 18-Dec-1998.
- The service agreement's SA type references the Quarterly Bill - Last Day Of Quarter bill period.

The following table shows when bill segments will be produced (assuming the account's bill cycle attempts to create segments as soon as possible) for several bill periods:

Earliest Date Segment	Bill Period
-----------------------	-------------

Will Be Produced	
1-Jan-1999	18-Dec-1998 thru 31-Dec-1998
1-Apr-1999	1-Jan-1999 thru 31-Mar-1999
...	

Refer to [Setting Up Bill Periods](#) for information about how to define this information.

Setting Up Bill Cycles

An account references a bill cycle. The bill cycle defines when the account is billed and when the account's debt is checked to determine if it's overdue. To define a bill cycle and its bill cycle schedule, open **Admin Menu, Bill Cycle**.

Description of Page

Enter a unique **Bill Cycle** and **Description** for every bill cycle.

Use the **Bill Cycle Schedule** collection to define when bills are produced for the accounts in a given bill cycle. The following fields are required for each instance:

Window Start Date	Specify the date on which the system should start trying to create bills for accounts in the cycle.
Window End Date	Specify the last day on which the system will create bills for accounts in the cycle. This should be the last possible date on which a meter read could be loaded into the system for the account.
Accounting Date	Specify the financial date associated with the bills' financial transaction. The accounting date defines the financial period(s) to which the bills will be booked in your general ledger.
Estimate Date	<p>The date on which the system is allowed to start estimating consumption if a real read cannot be found. Leave this date blank to inhibit consumption estimation. When specified, this date is typically on or shortly before the window end date.</p> <p>The system will only estimate a service agreement's consumption if: 1) the bill cycle schedule allows estimation, and 2) the service agreement allows estimation.</p>

Freeze and Complete

Turn on this switch if the system should freeze and complete any bill that is created without errors. If this switch is turned off, all bills created by the billing process will be left in the unfinished state. You would only turn this switch off if you want to verify an entire bill run prior to freezing it (e.g., if you are introducing a new version of a rate). If you turn this off, you will need to return to this page after verifying a bill run and turn it back on for the customers to receive bills. When the system next runs, it deletes all unfrozen bills and recreates them as per the instructions on the bill cycle schedule.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI BILL CYC](#).

The batch bill creation process uses this schedule to define the bill cycles for which it should create bills.

Don't forget. After you set up the bill cycles that correspond with service cycles, go to [Setting Up Service Cycle Schedules](#) and update the service cycles accordingly.

Setting Up Service Route Types

Every route within a service cycle references a route type. The route type controls:

- The batch process used to download and upload a route's meter reads.
- Whether a route is downloaded as part of the service cycle schedule download process.

Open **Admin Menu, Route Type** to define your route types.

Description of Page

The following fields display:

Route Type

The unique identifier of the route type.

Description

The description of the route type.

Batch Control

This defines the batch process used to download the meter reads for routes of this type. If you have multiple meter reading device formats you will have multiple batch processes – one for each format.

Process Routes In Cycle

Turn this switch on if routes of this type have their meter reads downloaded when their service cycle is downloaded. This switch will be on for most types of routes. A classic example of a type of route where this switch is off would be a route containing translators (a translator sends in reads when it is polled).

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MR RTE TYPE](#).

Setting Up Service Cycles And Routes

When you set up a metered service point you must define its service cycle, route and sequence within the route. To define a service cycle and its routes, open **Admin Menu, Service Cycle**.

Note. Every service cycle has a calendar that defines when the service points in the cycle are read. For more information about this calendar, see [Setting Up Service Cycle Schedules](#).

Description of Page

Enter a unique **Read Cycle** and **Description** for the service cycle.

Enter the **Bill Cycle** to populate on accounts with service points in this service cycle. The system updates an account's bill cycle when a service agreement is activated (assuming the account's bill cycle is not protected). Refer to [How Does An Account Get Its Bill Cycle?](#) for more information.

If the service cycle is associated with a **Service Provider** (rather than your own company's), define the owner of the cycle. Refer to [MDMAs And Service Cycles](#) for more information about service providers and their service cycles.

Use the **Service Routes** collection to define the routes within a given service cycle. The following fields display for each service route:

Service Route	The unique identifier of the route within the cycle (you can use the same route ID in many cycles).
Description	The route's description.
Route Type	The type of route. This controls if and how the route's meter reads are downloaded. Refer to Setting Up Service Route Types for more information.

Characteristics

The **Characteristics** collection contains information that describes miscellaneous information about the service route.

The following fields display:

Effective Date	Indicate the effective date for this characteristic.
Characteristic Type	Controls the order in which characteristics of the same type are displayed.
Characteristic Value	Indicate the value of the characteristic.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_MR_CYC](#) and [CI_MR RTE](#).

Setting Up Service Cycle Schedules

The service cycle schedule defines when the service points in a given cycle are scheduled to be read. Open **Admin Menu, Service Schedule** to maintain this information.

Refer to [Designing Cycles for Metered Services](#) for more information about service cycle. Refer to [The Relationship Between Metered Service Cycles and Bill Cycles](#) for more information about how service cycles are linked to bill cycles.

Description of Page

When you want to add a new service cycle schedule, you must specify the following information:

Service Cycle	Enter the service cycle ID of the cycle whose routes will be downloaded.
Scheduled Selection Date	Specify the date on which the system is meant to download information about the cycle's meters. This date should be a day or two before the scheduled work date.

Define the date the meters in the route are scheduled to be read using **Scheduled Work Date**. This date is extremely important as billing uses it when it looks for meter reads for service points in this cycle. If billing can't find a reading, consumption will be estimated as of this date (assuming the service agreement allows estimation).

Click **Pre-Generate Routes** if you need to finesse the cycle's routes on this Scheduled Selection Date (e.g., because some of the routes shouldn't be downloaded on a given date because the system is meant to estimate consumption). If you don't click this button, the system will create the routes on the scheduled selection date. It does this by creating a route extract for each downloadable route within the cycle (as defined by the route type). If you click this button, the system shows all routes within the cycle in the grid below. You must then define how the system is supposed to download each route on the scheduled selection date.

Click **Delete Routes** if you have pre-generated the routes and you want to remove them and allow the system to create the routes on the scheduled selection date.

Use the **Service Schedule Routes** to define how individual routes within the cycle should be handled during the download. The following fields are required for each schedule read date:

Service Route	The unique ID of the service route. The route's route type is displayed adjacent.
Schedule Type	This defines if and how the system is supposed to download the route's meter reads. Valid values are: Cust Read , Download , Estimate . Only those routes defined as Download will have meter reads downloaded. The other values are used to document why the route won't be downloaded.
Schedule Status	This defines the download status of the route. Valid values are Pending and Complete . This value is only displayed for routes with a schedule type of Download .

Where Used

This information is used by the meter read download process to determine which meter reads to download.

This information is used by the billing process to determine the date on which it expects to find a read. If it cannot find a read on or around this date, and it's OK to estimate consumption, consumption will be estimated as of the scheduled work date. The system uses the service agreement's rate schedule's frequency to determine the period of time around the scheduled work date in which it looks for a read. Refer to [Defining Frequency Codes](#) for more information.

Setting Up Bill Periods

Some SA types reference a bill period. The bill period defines when the service agreement's bill segments are produced and the respective end date of each bill segment.

Refer to [Designing Bill Periods](#) for more information.

To define a bill period and the bill period schedule, open **Admin Menu, Bill Period**.

Description of Page

Enter a unique **Bill Period** and a **Description** for every bill period.

Use the **Bill Period Schedules** collection when the system should create bill segments for service agreements that use a given bill period. It also defines the end date of each respective bill segment. The following fields are required:

Bill Date	Specify the earliest date on which the system is allowed to create a bill segment for service agreements using this bill period.
Bill Seg End Date	Specify the end date of the bill segment. For future bills, this will be after the bill date. For retro bills, this will be before the bill date.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI BILL PERIOD](#).

This information is used by the bill segment creation process to determine the end date of service agreements that use a bill period.

Defining Statement Cycles

If you have persons set up in the system to receive statements with financial information, you will need to assign them to a statement cycle and define a schedule for the statement cycle.

Refer to [The Big Picture of Complex Statements](#) for more information about statements.

Contents

[The Big Picture Of Statement Cycles](#)
[Setting Up Statement Cycles](#)

The Big Picture Of Statement Cycles

A statement cycle has a similar purpose to that of a bill cycle. It controls when statements will be produced.

Contents

[The Cyclical Statement Process](#)
[Designing Your Statement Cycles](#)
[Lifecycle of a Statement Cycle Schedule](#)

The Cyclical Statement Process

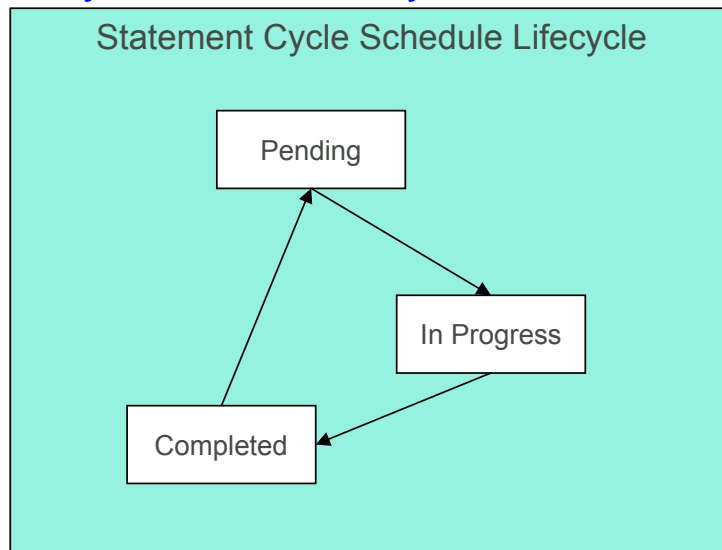
Persons who wish to receive statements will work with you to determine how often these statements should be produced. Some persons may want a monthly statement, some a quarterly and some annually. For each unique schedule that is designed for your various statement persons, you will set up a Statement Cycle and its schedule.

Designing Your Statement Cycles

The number of statement cycles is determined by a combination of the frequency that you will send statements to the statement persons and how many statement cycles you wish to manage within the same frequency.

So, for example, for all the statement persons who wish to receive a monthly statement, will you create only one monthly Statement Cycle so that all monthly statements are produced the same day? Or will you have several monthly statement cycles scheduled throughout the month? The answer will depend on the volume of statements being produced and on how you want to manage the statement production.

Lifecycle of a Statement Cycle Schedule



Pending

The statement cycle schedule is added in this state. The Create Statements background process finds records in this state to process on the appropriate date.

Refer to [Create Statements Background Process](#) for more information.

In Progress

Records in this state are currently being processed by the Create Statements background process.

Completed

Records in this state have already been processed by the Create Statements background process. If a problem occurs with the Create Statements background process and it needs to be rerun, simply change the status back to **pending** and rerun the process.

Setting Up Statement Cycles

A Statement Construct references a statement cycle. The statement cycle defines when the statement person will receive statements with financial information related to the accounts and service agreements linked to the statement construct. To define a statement cycle and its statement cycle schedule, open **Admin Menu, Statement Cycle**.

Description of Page

Enter a unique **Statement Cycle** and **Description** for every statement cycle.

Use the **Statement Cycle Schedule** collection to define when statements are produced for the persons with statement construct records in the given statement cycle. The following fields are required for each instance:

Processing Date

Specify the date on which the system should create statements for persons with statement construct records in the cycle.

Status

Indicates the status of the cycle schedule. Refer to [Lifecycle of a Statement Cycle Schedule](#) for more information.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI STM CYC](#).

The batch statement creation process uses this schedule to determine which statement cycles for it should create statements for the statement construct records. Refer to [Create Statements Background Process](#) for more information.

Defining Service Agreement Types

Every service agreement must reference a SA type. The SA type defines what you sell, how much you sell it for, to whom you sell it, how overdue debt is collected, and how sales will be booked in your general ledger.

Perfect foresight. In a perfect world, the other control tables would have been set up with perfect foresight of setting up your SA types. In reality, setting up your SA types may invalidate some of your earlier decisions. Don't feel bad if this happens, some amount of iteration is natural.

Contents

- [Designing SA Types For Service Agreements With Service Points](#)
- [Designing SA Types For SAs Without Service Points](#)
- [Designing SA Type For Other Segmentations](#)
- [SA Types And The Financial Design](#)
- [Setting Up SA Types](#)

Designing SA Types For Service Agreements With Service Points

Designing your SA types is an iterative process. To minimize some iterations, we recommend using the steps outlined in this section to complete the following table. When the table is complete, you're ready to set up your SA types.

Division/SA Type	Service Type

The topics in this section provide guidelines describing how to fill in this table for SA types associated with service agreements that charge for service point-oriented services.

Contents

- [CIS Division Segmentation](#)
- [Service Segmentation](#)
- [Receivable Segmentation](#)
- [Revenue Segmentation](#)
- [Rate Segmentation](#)
- [Service Point \(SP\) Type Segmentation](#)
- [Company Usage Segmentation](#)
- [Debt Class Segmentation](#)
- [Budget Billing Segmentation](#)

CIS Division Segmentation

A CIS division is typically associated with a jurisdiction. The definition of a jurisdiction is a geographic-oriented entity with unique business rules. For example, if you conduct business in California and Nevada, and each state has different collection rules, you will need a separate jurisdiction for each state. You must set up a CIS division for each jurisdiction in which you conduct business.

If we assume that you are located in a single jurisdiction – say California – we will need a single CIS division for all of our SA types.

CIS Division/SA Type
CA

Service Segmentation

At a minimum, you will have one SA type for every different type of utility service offered by your organization. If we assume you sell electricity, gas, water, wastewater and cable; your SA Types will be as follows:

CIS Division/SA Type	Service Type
CA/G	Gas service
CA/W	Water service
CA/E	Electric service
CA/WW	Waste water service
CA/CABLE	Cable

Non Utility Services. Earlier in this manual, service types are discussed in respect of meters and items. However, you may require additional service types if you have non-utility services. In the above table, we have only shown utility oriented services and their respective CIS Divisions. Later in this section, we will encounter a few more service types. Refer to [Setting Up Service Types](#) for more information.

Receivable Segmentation

Many organizations segregate their receivable balances in the general ledger. For example, the receivable amount associated with gas and water service may be maintained in separate GL accounts.

If your organization does this, you will probably have at least one SA type for each such receivable account because each SA type references a distribution code that typically contains the receivable account.

- The word *probably* is underlined because this is a rule of thumb. There are situations where the number of receivable accounts isn't directly related to the number of SA types. This happens when an organization maintains very detailed receivable accounts in the general ledger and maintaining a one-to-one relationship between SA types and distribution codes would lead to a massive proliferation of SA types (and you don't want this!). If your organization maintains very detailed receivable accounts, please speak to your implementers, they should be able to introduce a small customization to generate the appropriate receivable account rather than extract it from the distribution code.

- The word *typically* is underlined because there are several SA types that don't book to a receivable account when bill segments are generated. For example, company usage and charitable contributions. Refer to [Company Usage Segmentation](#) and [Charitable Contribution Segmentation](#) for examples of SA types that don't book to receivable accounts.

We'll simplify our example and assume your organization has one receivable account for all types of utility service. Given this, we won't need additional SA types to support receivable segmentation:

CIS Division/ SA Type	Service Type	Distribution Code
CA/G	Gas service	A/R-UTIL
CA/W	Water service	A/R-UTIL
CA/E	Electric service	A/R-UTIL
CA/WW	Waste water service	A/R-UTIL
CA/CABLE	Cable service	A/R-UTIL

Revenue Segmentation

Look at your rates and determine which rates can be used by each SA type. The following table shows the sample rates that can be used for each service:

CIS Division/ SA Type	Service Type	Distribution Code	Rates
CA/G	Gas service	A/R-UTIL	GALL-1
CA/W	Water service	A/R-UTIL	WALL-1
CA/E	Electric service	A/R-UTIL	ERES-1, ERES-2, ECOM-1, EIND-1, ELAMP-1
CA/WW	Waste water service	A/R-UTIL	WWALL-1
CA/CABLE	Cable service	A/R-UTIL	CABLE

Now, look at the rates' rate components GL Distribution window. You're looking for rate components whose GL distribution is affected by revenue class. If there are no revenue classes referenced on the rate components, this means that the revenue associated with the rate will be booked to a single GL account regardless of the type of customer. If you see revenue classes, this means that the revenue account associated with the rate component(s) differs depending on the SA type's revenue class. If revenue classes are used in the rates, you must create a different SA Type for every revenue class.

Let's assume the following:

- The gas rate (GALL-1) references the **RESIDENTIAL**, **COMMERCIAL** and **INDUSTRIAL** revenue classes in order to differentiate revenue based on the type of customer.
- None of the other rates differentiate revenue based on customer class.

Our SA types will now look as follows:

CIS Division/	Service	Distribution	Revenue	Rates
---------------	---------	--------------	---------	-------

SA Type	Type	Code	Class	
CA/G-RES	Gas service	A/R-UTIL	R – residential	GALL-1
CA/G-COM	Gas service	A/R-UTIL	C – commercial	GALL-1
CA/G-IND	Gas service	A/R-UTIL	I – industrial	GALL-1
CA/W	Water service	A/R-UTIL	N/A	WALL-1
CA/E	Electric service	A/R-UTIL	N/A	ERES-1, ERES-2, ECOM-1, EIND-1, ELAMP-1
CA/WW	Waste water service	A/R-UTIL	N/A	WWALL-1
CA/CABLE	Cable service	A/R-UTIL	N/A	CABLE

Notice that we created new SA types for gas in order to specify the respective revenue class. We didn't do this for the other services because it isn't necessary. However, you should feel free to do this if it feels right or if you need it for reporting purposes. For example, if you want to report on all *residential* service agreements, but you differentiate only gas by residential versus commercial, you'll be in trouble.

Rate Segmentation

Every SA Type whose service agreements have their charges calculated with a rate must have one or more rates linked to it. These define which rates can be linked to the SA type's service agreements. If different service agreements have different valid rate combinations, you will need to split the SA types accordingly.

For example, look at the electric rates. If we assume:

- the RES rates can only be used for residential customer,
- the IND rates can only be used for industrial customers,
- the COM rates can only be used for commercial customers,
- the LAMP rates can only be used for lamp customers

Then you might want to set up a new SA type to indicate such. The advantage of doing this is that you get more control over which rates can be used on a given SA type. The disadvantage is that you proliferate SA types. Unfortunately, it's really a question of taste.

CIS Division/ SA Type	Service Type	Distribution Code	Revenue Class	Rates
CA/G-RES	Gas service	A/R-UTIL	R – residential	GALL-1
CA/G-COM	Gas service	A/R-UTIL	C – commercial	GALL-1
CA/G-IND	Gas service	A/R-UTIL	I – industrial	GALL-1
CA/W	Water service	A/R-UTIL	N/A	WALL-1
CA/E-RES	Electric service	A/R-UTIL	N/A	ERES-1, ERES-2
CA/E-COM	Electric service	A/R-UTIL	N/A	ECOM-1
CA/E-IND	Electric service	A/R-UTIL	N/A	EIND-1
CA/LAMP	Electric service	A/R-UTIL	N/A	ELAMP-1

CA/WW	Waste water service	A/R-UTIL	N/A	WWALL-1
CA/CABLE	Cable service	A/R-UTIL	N/A	CABLE

Warning! Don't be too specific in your analysis in respect of rate segmentation because you could end up with a separate SA type for every rate (and you don't want this). We understand this is a very subjective warning, but we recommend that you start out with broad bands of rates that can be used on a SA type and narrow it down if you end up unhappy with the results. For example, you don't have to set up a separate SA type for low-income residential gas customers just because they have a special rate. Rather, you can leave your SA types as they are and treat low-income gas customers as a subset of your residential gas customers. **For more information**, refer to [Setting Up Start Options](#).

Rate override. If a service agreement of this type may be linked to a terms of service record, you must also consider whether or not the [rate schedule could be overridden](#) by a template SA linked to the terms of service record. Refer to [SA Type - Rate](#) for information about the possible values for this field.

Service Point (SP) Type Segmentation

Every SA Type whose service agreements exist to bill for service point-oriented service has one or more SP types. These define which type of service points can be linked to the SA type's service agreements. If different service agreements have different valid SP types, you will need to split the SA types accordingly.

For each service point-oriented SA type, determine if there are any restrictions in respect of the types of service points that can use the SA type's rates. For example, if we assume that only commercial SP types can be used by commercial customers, industrial SP types by industrial customers, residential SP types by residential customers, your SA types will be:

CIS Division/ SA Type	Service Type	Distribution Code	Revenue Class	Rates	SP Type
CA/G-RES	Gas service	A/R-UTIL	R – residential	GALL-1	G-RES
CA/G-COM	Gas service	A/R-UTIL	C – commercial	GALL-1	G-COM
CA/G-IND	Gas service	A/R-UTIL	I – industrial	GALL-1	G-IND
CA/W-RES	Water service	A/R-UTIL	N/A	WALL-1	W-RES
CA/W-COM	Water service	A/R-UTIL	N/A	WALL-1	W-COM
CA/W-IND	Water service	A/R-UTIL	N/A	WALL-1	W-IND
CA/E-RES	Electric service	A/R-UTIL	N/A	ERES-1, ERES-2	E-RES
CA/E-COM	Electric service	A/R-UTIL	N/A	ECOM-1	E-COM
CA/E-IND	Electric service	A/R-UTIL	N/A	EIND-1	E-IND
CA/WW-RES	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-RES,

					WW-RES
CA/WW-COM	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-COM, WW-COM
CA/WW-IND	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-IND, WW- IND
CA/CABLE	Cable service	A/R-UTIL	N/A	CABLE	CABLE

Notice the wastewater SA types reference both water and waste water service points. This is intentional as wastewater service uses the consumption from the water service to calculate some part of the wastewater charge.

Company Usage Segmentation

Up to now, we've discussed SA types associated with service agreements linked to your customers. The system has also been designed to keep track of the expenses associated with your company's use of power. If you want the system to do this, you must create at least one SA type for each service consumed by your organization.

For example, if we assume your organization consumes electric, gas, and water service; your SA types will now be as follows:

CIS Division/ SA Type	Service Type	Distribution Code	Revenue Class	Rates	SP Type	Bill Seg Type
CA/G-RES	Gas service	A/R-UTIL	R – residential	GALL-1	G-RES	SP-RATED
CA/G-COM	Gas service	A/R-UTIL	C – commercial	GALL-1	G-COM	SP-RATED
CA/G-IND	Gas service	A/R-UTIL	I – industrial	GALL-1	G-IND	SP-RATED
CA/W-RES	Water service	A/R-UTIL	N/A	WALL-1	W-RES	SP-RATED
CA/W-COM	Water service	A/R-UTIL	N/A	WALL-1	W-COM	SP-RATED
CA/W-IND	Water service	A/R-UTIL	N/A	WALL-1	W-IND	SP-RATED
CA/E-RES	Electric service	A/R-UTIL	N/A	ERES-1, ERES-2	E-RES	SP-RATED
CA/E-COM	Electric service	A/R-UTIL	N/A	ECOM-1	E-COM	SP-RATED
CA/E-IND	Electric service	A/R-UTIL	N/A	EIND-1	E-IND	SP-RATED
CA/WW-RES	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-RES, WW-RES	SP-RATED
CA/WW-COM	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-COM, WW-COM	SP-RATED
CA/WW-IND	Wastewater service	A/R-UTIL	N/A	WWALL-1	W-IND, WW-IND	SP-RATED
CA/CABLE	Cable service	A/R-UTIL	N/A	CABLE	CABLE	SP-RATED
CA/E-COY	Electric service	EXP-COMP	N/A	E CO USE	E-CO USE	COMPUSAG
CA/G-COY	Gas service	EXP-COMP	N/A	G CO USE	G-CO USE	COMPUSAG
CA/W-COY	Water service	EXP-COMP	N/A	W CO USE	W-CO	COMPUSAG

					USE	
--	--	--	--	--	-----	--

Notice the three company usage SA types do not reference an A/R account as their distribution code. This is because when bill segments are created for these types of service agreements, the system must debit an expense account (or contra-revenue account) rather than a receivable account.

Also notice we introduced a new column – Bill Segment Type. Notice that the customer-oriented SA types use the SP-RATED bill segment type and the company usage SA types use the COMPUSAG bill segment type. Different bill segment types are necessary because company usage SA types use a different algorithm to calculate their bill segment's financial transaction algorithm because they don't affect either payoff or current balance.

For more information, refer to [Designing and Defining Bill Segment Types](#).

Debt Class Segmentation

Every SA Type has a debt class. The debt class is used to categorize a service agreement's debt for the purpose of credit and collections (C&C) analysis. If a given SA Type has different categories of debt from C&C's perspective, you will have to split the SA Type.

For more information about debt class, refer to [Designing Your Collection Procedures](#).

If we assume that your residential services are regulated and your commercial and industrial services are deregulated, we won't have to introduce additional SA types.

CIS Division/ SA Type	Distribution Code	Revenue Class	Rates	SP Type	Bill Seg Type	Debt Class
CA/G-RES	A/R-UTIL	R – residential	GALL-1	G-RES	SP-RATED	REGU
CA/G-COM	A/R-UTIL	C – commercial	GALL-1	G-COM	SP-RATED	UNRE
CA/G-IND	A/R-UTIL	I – industrial	GALL-1	G-IND	SP-RATED	UNRE
CA/W-RES	A/R-UTIL	N/A	WALL-1	W-RES	SP-RATED	REGU
CA/W-COM	A/R-UTIL	N/A	WALL-1	W-COM	SP-RATED	UNRE
CA/W-IND	A/R-UTIL	N/A	WALL-1	W-IND	SP-RATED	UNRE
CA/E-RES	A/R-UTIL	N/A	ERES-1, ERES-2	E-RES	SP-RATED	REGU
CA/E-COM	A/R-UTIL	N/A	ECOM-1	E-COM	SP-RATED	UNRE
CA/E-IND	A/R-UTIL	N/A	EIND-1	E-IND	SP-RATED	UNRE
CA/WW-RES	A/R-UTIL	N/A	WWALL-1	W-RES, WW-RES	SP-RATED	REGU
CA/WW-COM	A/R-UTIL	N/A	WWALL-1	W-COM, WW-COM	SP-RATED	UNRE

CA/WW-IND	A/R-UTIL	N/A	WWALL-1	W-IND, WW-IND	SP-RATED	UNRE
CA/CABLE	A/R-UTIL	N/A	CABLE	CABLE	SP-RATED	UNRE
CA/E-COY	EXP-COMP	N/A	E CO USE	E-CO USE	COMPUSAG	No debt
CA/G-COY	EXP-COMP	N/A	G CO USE	G-CO USE	COMPUSAG	No debt
CA/W-COY	EXP-COMP	N/A	W CO USE	W-CO USE	COMPUSAG	No debt

Notice the three company usage SA types do not have a debt class. This is because their bill segment type's FT algorithm doesn't cause debt to be created and therefore there is no reason to have a debt class. However, you'll need to create a "dummy" debt class – call it N/A – for these SA types because every SA type must reference a debt class.

Budget Billing Segmentation

Many utilities offer their customers levelized payment plans to smooth out the seasonal bill variations. We call this levelized amount the *budget amount*.

Refer to [Budget Billing](#) for more information about budgets in general. Refer to [Billing – Current Balance versus Payoff Balance](#) for an example of budget billing accounting.

If we assume that you only allow budget billing on the electric and gas residential services, then you'll need to update your CA/G-RES and CA/E-RES SA types:

CIS Division/ SA Type	Dist Code	Rev Class	Rates	SP Type	Bill Seg Type	Debt Class	Recurring Charge Control Info
CA/G-RES	A/R-UTIL	R	GALL-1	G-RES	SP-RATED	REGU	Amount to bill is <i>Not Allowed</i> Amount is <i>Optional</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Budget Amount</i> :
CA/G-COM	A/R-UTIL	C	GALL-1	G-COM	SP-RATED	UNRE	
CA/G-IND	A/R-UTIL	I	GALL-1	G-IND	SP-RATED	UNRE	
CA/W-RES	A/R-UTIL	N/A	WALL-1	W-RES	SP-RATED	REGU	
CA/W-COM	A/R-UTIL	N/A	WALL-1	W-COM	SP-RATED	UNRE	
CA/W-IND	A/R-UTIL	N/A	WALL-1	W-IND	SP-RATED	UNRE	
CA/E-RES	A/R-UTIL	N/A	ERES-1, ERES-2	E-RES	SP-RATED	REGU	Amount to bill is <i>Not Allowed</i> Amount is <i>Optional</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Budget Amount</i> :
CA/E-COM	A/R-UTIL	N/A	ECOM-1	E-COM	SP-RATED	UNRE	

CA/E-IND	A/R-UTIL	N/A	EIND-1	E-IND	SP-RATED	UNRE	
CA/WW-RES	A/R-UTIL	N/A	WWALL-1	W-RES, WW-RES	SP-RATED	REGU	
CA/WW-COM	A/R-UTIL	N/A	WWALL-1	W-COM, WW-COM	SP-RATED	UNRE	
CA/WW-IND	A/R-UTIL	N/A	WWALL-1	W-IND, WW-IND	SP-RATED	UNRE	
CA/CABLE	A/R-UTIL	N/A	CABLE	CABLE	SP-RATED	UNRE	
CA/E-COY	EXP- COMP	N/A	E CO USE	E-CO USE	COMPUSAG	No debt	
CA/G-COY	EXP- COMP	N/A	G CO USE	G-CO USE	COMPUSAG	No debt	
CA/W-COY	EXP- COMP	N/A	W CO USE	W-CO USE	COMPUSAG	No debt	

Notice the following:

- We updated the two SA types to allow recurring charge information.
- The Recurring Charge Amount is **Optional**. Why? Because those customers who aren't on a budget plan won't have a recurring charge amount. Those customers on a budget will have a recurring charge amount.

Turn on Eligible for Budget. Besides indicating that these SA types use recurring charges, you also need to turn on the Eligible for Budget switch on the SA type to indicate that this type of SA participates in budget processing.

Override Budget Eligibility. You may plug in an override budget eligibility algorithm on an SA type that is configured to be Eligible for Budget if certain service agreements of this type are not eligible.

Refer to the Description of Page under [SA Type – Billing](#) for the definition of the recurring charge attributes.

Designing SA Types For SAs Without Service Points

The topics in this section provide guidelines describing how to design the SA types associated with your service agreements that don't have service points.

Contents

[Overpayment Segmentation](#)

[Write Off Segmentation](#)
[Connection Charge Segmentation](#)
[Charitable Contribution Segmentation](#)
[Payment Arrangement Segmentation](#)
[Merchandise Segmentation - Installment Billing](#)
[Deposit Segmentation - Installment Billing](#)
[Billable Charge Segmentation](#)
[Over/Under Cash Drawer Segmentation](#)
[Payment Upload Error Segmentation](#)
[CIAC Segmentation](#)
[Loan Segmentation](#)
[Non-billed Budget Segmentation](#)

Overpayment Segmentation

When a customer pays more than they owe, you must decide what to do with the excess money. The following points describe two possibilities:

- You could create a new service agreement to hold the excess (let's call it an overpayment SA). The credit would be transferred from this service agreement to the billable service agreements when the next bill is completed. This means that all billable service agreements have the same opportunity to receive the overpayment when they are billed in the future.
- You could amalgamate the excess payment on one of the existing, billable service agreements. For example, if a customer has both electric and gas service, the excess funds could be kept on either the gas or the electric SA. This would result in the following:
 - The service agreements that do NOT receive the overpayment will have debt when they are next billed.
 - The service agreement that receives the overpayment could have its future debt offset by the overpayment (meaning that it could have a credit balance until the service agreement's future bill segments offset the overpayment amount).

The above situation is not desirable unless the customer intentionally overpaid one service agreement. The first method (keeping the overpayment on a separate service agreement) obviates this potential problem. Obviously, if your organization sells a single service (and therefore your customers have a single service agreement) you would choose the second method.

You control which method is used by plugging in the appropriate **Overpayment Distribution** algorithm on each [Customer Class](#) (i.e., you can choose a different method for different customer classes). If you choose to hold overpayments on a separate SA, then you must set up an SA Type as described in the following table:

CIS Division/ SA Type	Service Type	Distrib. Code	Eligible for Billing	Debt Class	Pay Seg Type	Do Not Overpay	One- time
CA/OVERPAY	Other	A/P – OVER	Not billed	N/A	Normal	No	Yes

Notice the following about the new overpayment SA type:

- It has an interesting distribution code. This is because when a payment segment is created for this type of service agreements, the system must credit a liability (an overpayment is a liability).

- It's important to indicate that the overpayment SA is a one-time service agreement. Why? Because this means that the system will automatically close the SA when it's balance falls to zero (i.e., when all of the overpayment has been used to satisfy future bills).
- A bill segment type is not needed because the system never creates bill segments for such service agreements (they exist only to hold excess credits).
- You may also want to turn on the alert message
- **Important!** You must plug-in a bill completion algorithm on this SA type. This bill completion algorithm will transfer the credit balance to the account's other service agreements when the bill is completed. Refer to [The Credit Transfer Algorithm](#) for more information about this algorithm.
- You must also reference this overpayment SA type as the parameter value on your overpayment algorithm (this algorithm is plugged in on the desired customer classes). Refer to [Overpayment Algorithm](#) for more information about this algorithm.

If overpayment means charitable contribution. Some organizations sponsor a program that works as follows – if a customer overpays a bill by a given amount (say \$5), this amount is assumed to be a charitable contribution. If you have this requirement, you will create another SA type to hold a customer's charitable contributions. This SA type will look similar to the one described below (see [Charitable Contribution Segmentation](#)) except it is not billable. The funds will be credited to this service agreement by creating a new overpayment algorithm that is similar to the base package [Overpayment Algorithm](#). This new algorithm will be very similar to the existing algorithm. The main difference will be that it will have to check if the overpayment amount is an exact value (say \$5). If so, it will create a payment segment for the charitable contribution SA type; otherwise it will create a payment segment for the overpayment SA.

Write Off Segmentation

When you write off non-collectable debt, you transfer the receivable from a “normal” service agreement onto one or more write-off service agreements. When the debt is transferred to a write-off service agreement, the distribution code on the “normal” service agreement is credited (typically an A/R GL account), and the distribution code on the write-off service agreement is debited.

You will almost always need a write-off service agreement whose distribution code is the write-off expense. However, you probably don't book all of the write-off amount to a write-off expense account. Why? Because the debt that you're writing off typically contains both revenue and liabilities. At write-off time, you want to book the written off revenue to a write-off expense account and you want to reduce the liabilities (you don't owe the liability if you don't get paid). This means you'll need another SA type for the liabilities. Refer to [The Ramifications of Write Offs in the General Ledger](#) for a complete explanation.

The following table contains the minimum number of SA Types that you'll need to hold your write-offs.

CIS Division/ SA Type	Service Type	Distrib. Code	Bill Seg Type	Debt Class	Pay Seg Type	Do Not Overpay
CA/WO-STD	Other	EXP-W/O	Not billed	WO	Normal	Yes

CA/WO-LIA	Other	LIA- General	Not billed	WO	Normal	Yes
-----------	-------	-----------------	------------	----	--------	-----

Notice the following about the new write-off SA types:

- They have interesting distribution codes. This is because when debt is transferred to these types of service agreements, the system must debit either an expense account (i.e., write-off expense) or a liability account. It's important to note that in [The Ramifications of Write Offs in the General Ledger](#) we explain how this liability account may be overwritten with the liability account that was originally booked.
- Neither needs a bill segment type because the system never creates bill segments for such service agreements (they exist only to hold uncollectable debt)
- Even though the debt is not collectable, it still has a debt class. Why? Because the system shows a customer's debt on many inquiries by debt class and it's important to show write-off debt on these queries.
- The combination of Payment Segment Type and Do Not Overpay are important. Refer to [The Ramifications of Write Offs in the General Ledger](#) for a complete explanation.

Note. The adjustment type used to set the offending service agreement's current balance equal to its payoff balance is defined on each write-offable SA type. The adjustment type used to transfer the delinquent debt to the write-off service agreement is defined on the write-off SA type.

An Alternative. If you have a limited number of liability accounts, you may decide to have a separate write-off service agreement for each liability account. Doing this would proliferate the number of service agreements created at write-off time. However, it would simplify the remittance of payment to the taxing authority if the reversed liability is ever paid.

Connection Charge Segmentation

If you levy connection charges, you have two options:

- You can create a SA type that exists purely to handle connect charge debt. After doing this, you'd create a start option for this SA type that causes an adjustment to be levied as part of the start service process. This adjustment would contain your standard connection charge. This approach would be used by a utility that had multiple services (e.g., a combined electric, gas, water utility) that only levies a single connection charge regardless of the number of services started. If you use this approach, make sure to indicate the SA type is non billable.
- You can levy a start adjustment on one of your existing SA types (e.g., CA/E-RES). The easiest way to do this is with a start option. On the start option you'd indicate an adjustment to be levied as part of the start service process. This adjustment would contain your standard connection charge. This approach would be used by a utility that had a single service offering (e.g., an electric-only utility). Refer to [Setting Up Start Options](#) for more information.

In the table below, we show what would be necessary if you want to have a separate service agreement for the connection charge.

CIS Division/ SA Type	Service Type	Distrib. Code	Bill Seg Type	Debt Class
--------------------------	-----------------	------------------	------------------	------------

CA/CONNECT	Other	A/R-UTIL	Not billed	REGU
------------	-------	----------	------------	------

Notice the following about the new connection charge SA type:

- It has a normal receivable distribution code.
- It doesn't need a bill segment type because the system never creates bill segments for such service agreements (its charged via an adjustment).
- The debt class is interesting – REGU (for regulated). We are intentionally linking the connection charge debt to the same debt class as the regulated debt from which it originates. This way, the C&C process will consider the connection charge debt the same as regulated debt and therefore perform the regulated collection (which results in the severance of all regulated service agreements).

Charitable Contribution Segmentation

If your organization accepts charitable contributions made by your customers, you must create a SA type to hold these contributions.

CIS Division/ SA Type	Service Type	Distrib. Code	Debt Class	Bill Seg Type	Recurring Charge Control Info
CA/CHARITY	Other	A/P-CHAR	CHAR	RECUR	Amount to bill is <i>Not Allowed</i> Amount is <i>Required</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Contrib. Amount</i>

Notice the following about the new charitable contribution SA type:

- It has an interesting distribution code. This is because when a payment is distributed to these types of service agreements, the system must credit a payable account (i.e., charitable contribution payable) rather than a receivable account. Note well, we have assumed a receivable is not incurred when the bill segment for the charitable contribution is created.
- It uses an interesting bill segment type – RECUR. This bill segment type was set up to create recurring charges that don't automatically stop at some point in time.
- The debt class is interesting – CHAR (for charity). This is done so that past due charitable contribution debt is treated separately from other types of debt.
- The recurring charge control information is set up as defined.

Refer to the Description of Page under [SA Type – Billing](#) for the **definition** of the recurring charge attributes. Refer to [Start Option Considerations For SA Types That Use Recurring Charges](#) for **how** you can use start options to automatically populate a service agreement's recurring charge fields with appropriate values when service is started.

Warning! It's important that you assign the charitable contribution SA type with a payment segment type that only affect current balance (as opposed to affecting current AND payoff balance). This is because there is no receivable recognized when the contribution is billed and therefore there is no payoff balance to relieve when it's paid.

Payment Arrangement Segmentation

If your organization allows customers to payoff outstanding debt using payment arrangements (e.g., current bill plus \$X), you will need a new SA type for every debt class that can have a payment arrangement. If we assume you can have payment arrangements for both regulated and unregulated debt, then you'll need at least two more SA types (you may have more SA types if you need to segregate the payment arrangement receivable amount by utility type (or some other type)).

CIS Division/ SA Type	Service Type	Distrib .Code	Debt Class	Bill Seg Type	Recurring Charge Control Info
CA/PA-REGU	Other	A/R- ARRG	REGU	RECUR-AS	Amount to bill is <i>Not Allowed</i> Amount is <i>Required</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Arrange Amount</i>
CA/PA-UNRE	Other	A/R- ARRG	UNRE	RECUR-AS	Amount to bill is <i>Not Allowed</i> Amount is <i>Required</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Arrange Amount</i>

Notice the following about the new payment arrangement SA types:

- They have an interesting distribution code. This is because when funds are transferred to these types of service agreements, the system must debit a receivable (i.e., payment arrangement receivable).
- They use an interesting bill segment type – RECUR-AS. This bill segment type was set up to create recurring charges that stop when the customer no longer has a payoff balance.
- Each new SA type references the debt class whose debt it will pay off. We are intentionally linking the payment arrangement debt to the same debt class as the regulated debt from which it originates. This way, the C&C process will consider the arrangement debt as the same as regulated debt and therefore perform the regulated collection (which results in the severance of all regulated service agreements).
- The recurring charge control information is set up as defined.

Refer to the Description of Page under [SA Type – Billing](#) for the definition of the recurring charge attributes. Refer to [Start Option Considerations For SA Types That Use Recurring Charges](#) for how you can use Start Options to automatically populate a service agreement's recurring charge fields with appropriate values when service is started.

Merchandise Segmentation - Installment Billing

If your organization allows customers to purchase merchandise using an installment plan, you must create a SA type for this.

No installments. If the customer must pay for the merchandise in one lump amount, you'd create an SA type similar to the [Connection Charge Segmentation](#) example.

CIS Division/ SA Type	Service Type	Distrib. Code	Debt Class	Bill Seg Type	Recurring Charge Control Info
CA/MERCH-I	Merch	A/R-MRCH	UNRE	RECUR- AS	Amount to bill is <i>Not Allowed</i> Amount is <i>Required</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Install Amount</i>

Notice the following about the new merchandise SA type:

- It has a normal receivable distribution code.
- It uses an interesting bill segment type – RECUR-AS. This bill segment type was set up to create recurring charges that stop when the customer no longer has a payoff balance.
- The recurring charge control information is set up as defined.

Refer to the Description of Page under [SA Type – Billing](#) for the definition of the recurring charge attributes. Refer to [Start Option Considerations For SA Types That Use Recurring Charges](#) for how you can use start options to automatically populate a service agreement's recurring charge fields with appropriate values when service is started. Refer to [Start Option Considerations For SA Types That Use Initial Adjustments](#) for how you can use start options to automatically populate a service agreement's recurring charge fields with appropriate values when service is started.

Deposit Segmentation - Installment Billing

If your organization allows customers to pay deposits using an installment plan, you must create an SA type for this.

No installments. If the customer must pay for the deposit in one lump amount, you'd create an SA type similar to the [Connection Charge Segmentation](#) example. Just make sure the adjustment that's levied to charge for the deposit amount doesn't affect payoff balance (when you bill a deposit, the customer doesn't really owe anything because it's not a true receivable from an accountant's perspective).

CIS Division/ SA Type	Service Type	Distrib. Code	Debt Class	Bill Seg Type	Recurring Charge Control Info
--------------------------	-----------------	------------------	---------------	------------------	-------------------------------

CA/DEP-I	Other	A/P-DEPO	DEP	RECURAT B	Amount to bill is <i>Required</i> Amount is <i>Required</i> Frequency is <i>Monthly</i> Recurring Amount Label is <i>Install Amount</i>
----------	-------	----------	-----	--------------	--

Notice the following about the new deposit SA type:

- It has an interesting distribution code. This is because when a payment is distributed to these types of service agreements, the system must credit a payable account (i.e., deposit payable) rather than a receivable account. Note well, we have assumed a receivable is not incurred when the bill segment for the deposit is created.
- It uses an interesting bill segment type – RECURATB. This bill segment type was set up to create recurring charges that stop when the system has billed the Total Amount to Bill.
- The debt class is interesting – DEP (for deposit). This is done so that past due deposit “debt” is treated separately from other types of debt.
- The recurring charge control information is set up as defined. Note well, the Amount to bill is *Required*.

Refer to the Description of Page under [SA Type – Billing](#) for the definition of the recurring charge attributes. Refer to [Start Option Considerations For SA Types That Use Recurring Charges](#) for how you can use start options to automatically populate a service agreement’s recurring charge fields with appropriate values when service is started.

Bill messages on receipt of deposit in full. The base package includes a special FT Freeze algorithm that can be specified on deposit SA Types. It recognizes when a deposit has been paid in full, and creates a bill messages to inform the customer. Refer to algorithm *DEP PIF MSG* in [Algorithm Types](#) for more information.

Billable Charge Segmentation

You create a billable charge whenever a customer should be charged for a service that occurs outside the normal course of business. For example, you would create a billable charge to charge a contractor for the repair of a ruptured gas line. You can also use billable charges to “pass through” other bill ready charges generated outside the system, by another application, or by a 3rd party supplier.

A billable charge must reference a service agreement. This service agreement behaves just like any other service agreement:

- **Bill segments are created for the service agreement.** Whenever billing is performed for an account with billable charge service agreements, the system creates a bill segment for each service agreement with unbilled charges. If multiple unbilled charges exist for a given service agreement, only one bill segment will be created and it will contain details about all of the billable charges.
- **Payments are distributed to the service agreement.** Payments made by an account are distributed to its billable charge service agreements just like any other service agreement.

- **Overdue debt is monitored.** The credit and collections process monitors billable charge service agreements for overdue debt and responds accordingly when overdue debt is detected.

Therefore, you must set up at least one SA type to hold your billable charge debt. You may have multiple charges based on billing frequencies, A/R booking, debt monitoring, etc. It's really up to you.

The easiest way to determine how many billable charge SA types you'll need is to define every conceivable billable charge (which you should have done when you designed your [billable charge templates](#)). Then ask yourself if they have the same billing and payment behavior, if so, you'll have one SA type. If not, you'll need one SA type for each combination.

We will assume your billable charges are all used to levy unusual one-off charges that can be collected in the same way, therefore we'll need one SA type.

CIS Division/ SA Type	Service Type	Distrib Code	Debt Class	Bill Seg Type	Billable Charge Templates	Rate
CA/ONETIME	Other	A/R-UTIL	UNRE	BILLCHRG	TREETRIM DAMAGE	None
CA/PASSTHRU	Electricity	A/R-OTHER	UNRE	BILLCHRG	None	None
CA/ADDON	Electricity	A/R-OTHER	EXTERNAL	BILLCHRG	None	TAXES

Notice the following about the new one time SA type:

- It has a normal receivable distribution code.
- Its debt class is unregulated.
- It uses an interesting bill segment type – BILLCHRG. This bill segment type was set up to create bill segments using billable charges.
- It references the valid billable charge templates that can be used on this SA type.

One Time Charge. The **ONE TIME** example shown above implies this SA type exists to hold one-time charges. Because of this, you should turn on the One Time Charge switch on the SA type so that service agreement's of this type are automatically closed when final payment is received. You don't have to do this because a customer could have a single billable charge service agreement that is perpetually active for pass through charges (i.e., it doesn't have a stop date). If you do this, the system will create a bill segment for this service agreement whenever it finds an unbilled billable charge linked to the service agreement.

Notice the following about the pass through SA Type:

- It doesn't use the normal distribution code or debt class. This is done so that the debt and receivable can be tracked separately. If these charges were being pass through from another system, you might want to track these financial values separately.
- It still uses the normal bill segment type – BILLCHRG. From a billing perspective, there is no difference between this and the one time SA Type.
- Templates are not relevant – these charges are not created on-line using templates, but are loaded via the [Billable Charge Upload Staging](#).

Notice the following about the add on charges SA Type:

- This is an example of bill-ready charges (similar to pass through) to which the system adds on other charges, for example, taxes.
- It still uses the normal bill segment type – BILLCHRG. From a billing perspective, there is no difference between this and the one time SA Type.
- It also uses a Rate. In this case, the bill creation algorithm (specified on the bill segment type) will take any billable charge lines and attach them to a bill. In addition, these billable charges will include billable charge service quantities (SQs). These service quantities will also be swept onto the bill segment, and the Rate (TAXES in this example) will be applied. In order for taxes to be calculated, the billable charge SQs must include the total taxable amount – the system is not able to apply the rate on top of the other billable charges. But, it can apply the tax rate to the SQs that are supplied.
- You can also use this technique to bill other rate-ready service quantities, like kWh, CCF, *etc.* This is a way to process rate-ready data for which you have a contract, but you do not know the meter (and therefore, cannot collect real meter reads).
- If the rate has SQ Rules, these will be applied as well.

For more information about billable charge templates, refer to [Setting Up Billable Charge Templates](#).

Over/Under Cash Drawer Segmentation

In order to balance a tender control that is out-of-balance, your organization must set up an account with a service agreement whose SA type references the over/under expense account. You will probably only have one service agreement that references this SA type, but you still must have it if you remit funds via a cash drawer.

For more information about over/under processing, refer to [How To Get An Unbalanced Tender Control In Balance \(Fixing Over/Under\)](#).

CIS Division/ SA Type	Service Type	Distrib.Code	Debt Class	Bill Seg Type
CA/OVR UNDR	Other	EXP-OV/UND	N/A	Not billed

Notice the following about the new SA type:

- It has an interesting distribution code. This is because when a payment segment is applied to this type of service agreement, the system must debit an expense account for under amounts (and credit it for over amounts).
- It doesn't need a bill segment type because the system never creates bill segments for such service agreements (it only has over/under payment segments linked to it).
- It uses the N/A debt class because the credit and collections process should never consider debt associated with service agreements of this type (because it's not really debt).

Payment Upload Error Segmentation

If the payment upload process detects an invalid account on a payment upload record, it will create a payment for the suspense service agreement defined on the upload process' tender source (see [Setting Up Tender Sources](#)). You should create a special SA type for this service agreement.

For more information about the payment upload process, refer to [Phase 3 – Create Payment Events, Tenders, Payments and Payment Segments](#).

CIS Division/ SA Type	Service Type	Distrib.Code	Debt Class	Bill Seg Type
CA/SUSPENSE	Other	EXP-MISC	N/A	Not billed

Notice the following about the new SA type:

- It has an interesting distribution code. This code should probably be a suspense account. All payment segments that are created for this service agreement will eventually be transferred to a "real" service agreement and therefore this GL account's balance should be zero when no payments are in suspense.
- It doesn't need a bill segment type because the system never creates bill segments for such service agreements (it only has invalid account payment segments linked to it).
- It uses the N/A debt class because the credit and collections process should never consider debt associated with service agreements of this type (because it's not really debt).

CIAC Segmentation

If your company bills and refunds Contribution In Aid of Construction (CIAC) contracts, you must create one or more SA types. CIAC contracts are typically used to levy charges associated with line extensions. These types of service agreements are different from other service agreements because the initial amount charged is refunded to the original payee when new properties (or extensions) are added to the extension (or when the new properties are subsequently billed for service).

Billable charges are used for the original CIAC service agreement. CIAC SA types are always **Billable Charge** SA types because an operator must specify the exact amount to charge the contractor using a billable charge. Refer to [Billable Charge Segmentation](#) for more information.

The following points describe how CIAC processing is implemented in the system:

- When a new line extension is build, you will create a new service agreement that references your CIAC SA type. This service agreement should be linked to a service point associated with the extension. The easiest way to create this service agreement and link it to the service point is by using the Start Account transaction. It's important to remember to define your CIAC SA type as a valid SA type for the SP type used to represent the line extension.

- When a new premise / service point is built that “hangs” off the original extension, the new service point should be linked to the original service point. This can be done by referencing the original service point as the “parent” service point of the new service point using a [foreign key value characteristic](#).
- The system periodically monitors the original CIAC service agreement. The process that performs this monitoring is referenced on the original CIAC service agreement’s SA type. The exact processing that takes place during this monitoring is up to your organization’s specific business requirement. For example, if you refund 10% of every bill produced for the “downstream” service points to the original contractor, you would have logic in your CIAC monitoring process that looks for recent bill segments produced for the “downstream” service agreements and then creates an adjustment for the master service agreement.

CIAC refunds are idiosyncratic. Because CIAC refund processing is idiosyncratic, we do not supply any CIAC monitoring processes in the base package. This is because the likelihood that they could be used is extremely low because of your organization’s unique requirements.

Therefore, if your organization performs CIAC processing, you should create a special SA type.

CIS Division/ SA Type	Service Type	Distrib.Co de	Debt Class	CIAC Process	Bill Seg Type
CA/CIAC	Other	A/R – CIAC	UNRE	Refund based on percent of future bills	Billable Charge

Loan Segmentation

If you loan money to customers that is recouped using an amortization schedule, you need to set up an SA type for the loan service agreement.

CIS Division/ SA Type	Service Type	Distrib.Code	Loan A/R Distrib. Code	Debt Class	Bill Seg Type
CA/LOAN	Other	A/R - LOAN	A/R - STLN	UNRE	LOAN

Refer to [Setting Up The System To Enable Loans](#) for more information.

Non-billed Budget Segmentation

If you allow your customers to pay set amounts at specified intervals (e.g. every two weeks), you need to set up SA types for non-billed budget service agreements. Non-billed budgets are typically used when your company bills on an infrequent basis and you want to provide your customers with a mechanism to make smaller payments more frequently. You may implement two types of non-billed budgets monitored and unmonitored, each type requiring a different SA type. You may also implement different renewal options for non-billed budgets.

CIS Division/ SA Type	Service Type	Distrib. Code	Non Billed Budget Monitoring	Debt Class	Renewal	Bill Seg Type
CA/NBB-MON	Other	A/P - OVPY	Monitored	NBB	Optional	Not billed
CA/NBB-UNMON	Other	A/P - OVPY	Unmonitored	N/A	Optional	Not billed

For more information about monitored and unmonitored non-billed budgets, refer to [Defining Non-billed Budget Options](#).

Designing SA Type For Other Segmentations

The earlier parts of this discussion described the most common factors that cause the creation of SA Types. However, many obscure factors could cause the introduction of more SA Types. In this section, we explain these more obscure factors.

Warning! We strongly recommend not being too pedantic when considering the factors described in this section. If you can only think of a few strange situations that would necessitate a SA type, think carefully before you introduce it. It's better to be a little less than perfect than end up with large number of obscure SA types.

Contents

- [Cash Distribution Segmentation](#)
- [Adjustment Profile Segmentation](#)
- [Late Payment Charge Segmentation](#)
- [Debt Classification Segmentation](#)
- [Allow Estimates Segmentation](#)
- [Severance Criteria Segmentation](#)
- [Deposit Class Segmentation](#)
- [Sub SA Types](#)
- [Financial Settlement SA Types](#)
- [Interval Billing SA Types](#)
- [Usage Request SA Types](#)
- [Initial Consumption Period Considerations](#)
- [Processing Sequence Considerations](#)

Cash Distribution Segmentation

Every SA Type has a payment segment type. The payment segment type defines the cash account to which the SA type's payments should be booked. If different service agreements have different cash accounts, you will need to split the SA types accordingly.

Adjustment Profile Segmentation

Every SA Type has one or more adjustment profiles. These profiles define the valid adjustment types that can be booked to the SA type's service agreements. If different service agreements within an SA type have different mixtures of valid adjustment types, you must split the SA types accordingly.

Late Payment Charge Segmentation

An option exists on SA Type that causes the system to generate a late payment charge if payment is not received on time. If you don't levy late payment charges on all service agreements, you will need to determine when you do and design your SA types accordingly.

In addition, if you levy late payment charges, the percentage levied and the algorithm that defines the amount of the outstanding balance subject to the charge is defined on the SA type.

Debt Classification Segmentation

Every SA Type has a debt class. The debt class is used to categorize a service agreement's debt for the purpose of credit and collections (C&C) analysis. If a given SA Type has different categories of debt from C&C's perspective, you must split the SA Type.

For more information about debt class, refer to [Designing Your Collection Procedures](#).

Write Off Debt Class vs. Normal Debt Class. An SA type references both write off debt class and normal debt class. An SA type's write-off debt class controls the write-off rules imposed on service agreements of a given type. An SA type's normal debt class controls the collection rules imposed on service agreements of a given type. Refer to [Different Collection Criteria For Different Customers And Different Debt](#) for more information about collection rules. Refer to [Different Write-Off Criteria For Different Customers And Different Debt](#) for more information about write-off rules.

Allow Estimates Segmentation

Every SA Type has a switch that controls whether the system estimates consumption if meter reads are missing at billing time. If a given SA Type has different situations when the system should and should not estimate, you will have to split the SA Type.

Override Note. You can override the value of the SA Type's estimation switch on an individual service agreement. This means that if only a few service agreements don't abide by the SA Type's estimation switch, you can change the switch value of these service agreements.

For more information about estimation, refer to [Setting Up Consumption Estimation Parameters](#).

Severance Criteria Segmentation

Every SA Type has severance criteria. The severance criteria define the severance process used to sever service if the customer doesn't pay. You can have multiple severance processes if different conditions warrant a different process. For example, you may have a different severance process if the customer has life support.

If you have a SA Type that requires different severance conditions other than those currently supported, you can make a programmatic change to introduce the additional conditions OR you can split the SA Type.

For more information about severance, refer to [Designing Your Severance Procedures](#).

Deposit Class Segmentation

Every SA Type that exists to hold a cash deposit will reference a deposit class. The deposit class defines the business rules that control various functions including interest calculation and refund criteria. You will need multiple deposit SA Types if any of the deposit class' rules / conditions differ for different types of deposits. For example, if residential customers use a different recommended deposit algorithm as compared to commercial customers, you'd need one SA type for residential deposits and another for commercial deposits (where the residential deposit SA type will reference the residential deposit class and the commercial deposit SA type will reference the commercial deposit class).

You will need additional deposit SA types if your customers can have multiple deposits where each deposit is restricted to a specific type of debt. For example, if separate deposits are held for regulated and unregulated debt (and a customer could hold a combination of regulated and unregulated debt), you'd need one SA type for regulated deposits and another for unregulated deposits.

For more information about deposit class, refer to [Designing and Defining Deposit Classes](#).

Sub SA Types

If you operate in a deregulated environment AND if you provide billing services for other service providers, you will need to create SA types to handle the billing of the service providers' charges.

Refer to [Designing Your SA Types And Start Options For Sub SA's](#) for more information.

Financial Settlement SA Types

If you operate in a deregulated environment, you may have to create financial settlement service agreements for the service providers. As explained in [Service Providers Have Service Agreements Too](#), **We Bill For Them** and **They Bill For Us** service providers require a service agreement to hold adjustments used to increase how much you owe the service provider (or how much they owe you).

Refer to [Designing SA Types For Service Provider Financial Settlements](#) for more information.

Interval Billing SA Types

If you have customers with interval meters, the SA types for these customers will require special setup.

Refer to [Designing Your SA Interval Billing Options](#) for more information.

Usage Request SA Types

If your organization uses a meter data management system to store meter reading information and the MDM is responsible for calculating bill determinants during billing, the SA types for these customers will require special setup.

For more information about usage requests, refer to [The Big Picture Of Usage Requests](#).

Initial Consumption Period Considerations

Bill segments produced for a service agreement have two time periods:

- The bill segment period. The bill segment period defines the entire period of time covered by a bill segment's charges.
- The consumption period. The consumption period defines the period of time used to calculate the number of days for daily charges.

The consumption period almost always starts one day after the bill segment period. The consumption period always ends on the bill segment's end date. For example, a bill segment period that spans 5-Jan-2002 through 6-Feb-2002 will almost always have a corresponding consumption period of 6-Jan-2002 through 6-Feb-2002. The reason that the start dates don't match is because a bill segment's start date equals the end date of the prior bill segment (i.e., the start date was already counted in the previous bill segment's consumption period and we don't want to count it twice).

The only time when the previous paragraph isn't true is the first bill segment that's produced for a new service agreement. This is because different utilities count the first day of a new service agreement differently than others. Because of this, a flag exists on SA Type called **Initial Start Date Option**. This flag controls whether the service agreement's start date is included in the consumption period in a service agreement's first bill segment.

The following table describes the ramifications of the options you can set for this flag.

Flag Value	Consumption Period Calculation	Use This Option When
<i>Add 1 Day Always</i>	The consumption period's start date is calculated by adding 1 day to the service agreement's start date. (The SA start date is never included in the consumption period for the first bill segment.)	You want the initial bill and all subsequent bills to have a consistently calculated consumption period (i.e., the consumption period for the first and all other bills is always one day less than the

		bill segment's period).
<i>Add 1 Day for Back-to-back</i>	<p>A back-to-back occurs when any service point for this service agreement was previously linked to a different service agreement that was stopped on the same date that the new service was started (i.e., there is no gap in the service).</p> <p>If a back-to-back is detected, the consumption period start date is calculated by adding 1 day to the SA start date.</p> <p>If no back-to-back is detected, the start date of the consumption period is the SA start date.</p>	<p>You want to flexibly handle consumption period calculation. If you start customers on the same date as the stop date of the previous customer, billing <u>does not include</u> the SA start date in the consumption period. However, if you start a new customer one day (or more) after the stop date of the previous customer, billing <u>includes</u> the first day of the service agreement in the consumption period.</p>
<i>Include First Day</i>	<p>The start date of the consumption period is the service agreement start date. (The SA start date is always included in the consumption period.)</p>	<p>You want to always include the first day AND you will never encounter a back-to-back situation.</p>

The example below shows how the consumption period would be calculated with the various options for a customer who starts service on January 1.

	First Bill Meter Read: Jan 31 Bill Segment Period: Jan 1 to Jan 31	Second Bill Meter Read: Feb 28 Bill Segment Period: Jan 31 to Feb 28	Third Bill Meter Read: Mar 31 Bill Segment Period: Feb 28 to Mar 31
Consumption period using <i>Add 1 Day Always</i>	Jan 2 to Jan 31 (30 days)	Feb 1 to Feb 28 (28 days)	Mar 1 to March 31 (31 days)
Consumption period using <i>Add 1 Day for Back-to-back</i> when back-to-back is detected	Jan 2 to Jan 31 (30 days)	Feb 1 to Feb 28 (28 days)	Mar 1 to March 31 (31 days)
Consumption period using <i>Add 1 Day for Back-to-back</i> when back-to-back is NOT detected	Jan 1 to Jan 31 (31 days)	Feb 1 to Feb 28 (28 days)	Mar 1 to March 31 (31 days)
Consumption period using <i>Include First Day</i>	Jan 1 to Jan 31 (31 days)	Feb 1 to Feb 28 (28 days)	Mar 1 to March 31 (31 days)

There may be SA types for which the value of this flag does not affect the consumption period calculation and still other SA types where this flag is never used. For example,

- For billable charge service agreements, the consumption period is equal to the start and end dates on the billable charge and therefore this flag is not applicable.
- A sub SA either inherits the consumption period from the master SA or it uses billable charges. As a result, billing does not use this flag.

- For some service agreements, the charges on the rate are not affected by the consumption period. For example, if you have a customer with a simple meter and a simple usage-based charge, billing calculates the amount of consumption between the start reading and end reading and applies the rate (i.e., the number of days in the consumption period doesn't impact the charges in some rates).
- Some service agreements are not billed, for example, [overpayment service agreements](#). For this type of service agreement and other types of service agreements that are not billed, this flag is not applicable. However, the system does not prevent a value from being entered in these cases to allow for an implementation process to use the flag if needed.

Setting this flag to an appropriate value is significant for certain types of service agreements.

- For services whose rate includes daily charges, the configuration of this flag may impact the first bill segment for the service agreement. For example, if the first bill period is October 1st through October 31st, do you consider consumption period to be 31 days or 30 days? How you want to bill the customer on the first bill determines how you set this flag.
- For interval billing service agreements, the interval usage recordings typically begin on the first day of service. If you add 1 to the start date, you miss billing for intervals on the first day. In this case, you should set the value to **Add 1 Day for Back-to-back**.

Refer to [Determine the Consumption Period](#) for more information.

Processing Sequence Considerations

You may have customers with a complex collection of contracts such that the calculation for one bill segment relies on information calculated by another bill segment for the same account. For example, perhaps you need to process your bill segments as follows:

- Step 1: Calculate bill segments for all the account's "single site" service agreements (i.e., each service agreement related to a single premise)
- Step 2: Calculate the bill segment for additional charges for the "head office" service agreement where the charges are based on the aggregated consumption of all premises from all bill segments (calculated in the first step).

To create the "head office" bill segment for the account, you must control the order in which the system creates the bill segments for each service agreement.

The SA type allows you to indicate a billing processing sequence that controls the order in which the service agreements are processed. The processing sequence is optional and service agreements are processed in ascending order of their SA type's billing processing sequence. As a result, SA Types with a processing sequence of zero are processed first, then 1, then 2, etc.

Note. If you have sub SAs linked to a master SA, the billing processing sequence is used first to order the creation of bill segments for the master SAs. If you populate a processing sequence on an SA type for a sub SA, it is used to control the order in which the sub SAs for a given master SA are processed relative to each other. Refer to [Designing Your SA Types And Start Options For Sub SAs](#) for more information.

The billing processing sequence also controls the order of service agreements in the following other processes:

- Execution of pre bill completion algorithms. The system processes each service agreement in the billing processing sequence order. Within each service agreement, the pre bill completion algorithms on its [SA type](#) are processed in the order of the algorithm's sequence.
- Execution of bill completion algorithms. The system processes each service agreement in the billing processing sequence order. Within each service agreement, the completion algorithms on its [SA type](#) are processed in the order of the algorithm's sequence.
- Interval Data Creation. For interval service agreements linked to interval profiles with profile creation algorithms defined, the system processes each service agreement in the processing sequence order. Within each service agreement, the [data creation algorithms](#) are processed in the order of the creation priority on the [profile type](#).

SA Types And The Financial Design

In this section, we provide an example of how our SA Types map to Bill Segment Types, Payment Segment Types, and Adjustment Profiles. This example is meant to help solidify the power of the financial model, it is not necessarily indicative of how your specific implementation will look.

Warning! If you are not comfortable with the topics described in [Defining Financial Transaction Options](#), the following table will not make sense.

Division/SA Type	Distribution Code	Bill Segment Type	Payment Segment Type	Adjustment Profiles
CA/G-RES	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES, BUDGET
CA/G-COM	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/G-IND	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/W-RES	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/W-COM	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/W-IND	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/E-RES	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES, BUDGET
CA/E-COM	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/E-IND	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/WW-RES	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/WW-COM	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/WW-IND	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/CABLE	A/R-UTIL	SP-RATED	NORMAL	BALXFER, MISCFEES
CA/E-COY	E-COMP	COMPUSAG		

Division/SA Type	Distribution Code	Bill Segment Type	Payment Segment Type	Adjustment Profiles
CA/G-COY	E-COMP	COMPUSAG		
CA/W-COY	E-COMP	COMPUSAG		
CA/WO-STD			NORMAL	BALXFER
CA/WO-LIA			NORMAL	BALXFER
CA/CONNECT			NORMAL	BALXFER, MISCFEES
CA/CHARITY	A/P-CHAR	RECUR	CHARITY	CHARITY
CA/PA-REGU	A/R-ARRG	RECUR-AS	NORMAL	BALXFER, MISCFEES, DPA
CA/PA-UNRE	A/R-ARRG	RECUR-AS	NORMAL	BALXFER, MISCFEES, DPA
CA/MERCH-I	A/R-MRCH	RECUR-AS	NORMAL	BALXFER, MISCFEES, MERCH
CA/DEP-I	A/P-DEPO	RECUR-AS	NORMAL	BALXFER, MISCFEES, DEPOSIT
CA/ONETIME	A/R-UTIL	BILLCHRG	NORMAL	BALXFER, MISCFEES
CA/OVR UNDR	EXP-OV/UND		NORMAL	
CA/OVERPAY	A/P-OVER		NORMAL	BALXFER
CA/SUSPENSE	A/R-SUSP		NORMAL	
CA/NBB	A/P-OVPY		NORMAL	BALXFER, MISCFEES, NBB

If you operate in a deregulated environment, you will also have additional SA types as described under [Designing Your SA Types And Start Options For Sub SA's](#) and [Designing SA Types For Service Provider Financial Settlements](#) you may have additional SA types.

Setting Up SA Types

In the previous section, Designing SA Types, we presented a case study that illustrated a mythical organization's SA types. In this section, we explain how to use the windows on the SA Type window group to maintain your SA Types.

When a new SA type is added. When you add a SA type whose service agreements use service points, you must update the respective SP types if the new SA type is defaulted when service is initially started at the service points of a given SP type. Refer to [SP Type – SA Type](#) for more information.

Contents

- [SA Type - Main Information](#)
- [SA Type - Detail](#)
- [SA Type - Billing](#)

[SA Type - Rate](#)
[SA Type - SP Type](#)
[SA Type - Adjustment Profiles](#)
[SA Type - C&C](#)
[SA Type - Billable Charge Template](#)
[SA Type - Characteristics](#)
[SA Type - Algorithms](#)
[SA Type - Billable Charge Overrides](#)
[SA Type - Contract Option Type](#)
[SA Type - Interval Info](#)
[SA Type - NBB Recommendation Rule](#)
[Setting Up Start Options](#)
[Start Options Merge](#)

SA Type - Main Information

Open **Admin Menu**, **SA Type** and navigate to the **Main** tab to define core information about your SA Types.

Warning! Every SA Type is owned by a CIS Division. This Division controls many values that can be referenced on the SA Type. If you don't understand Divisions and their place in the application, do NOT attempt to set up your SA Types. Rather, refer to [Setting Up CIS Divisions](#) before proceeding.

Description of Page

Enter a unique combination of **CIS Division** and **SA Type** for every service agreement type.

Enter a **Description** for the SA type.

Service Type defines the type of service associated with the SA type. If the SA type has rates, only rates belonging to this service type may be linked to the SA type.

For more information about service types, refer to [Setting Up Service Types](#).

Select the **Distribution Code** and **GL Division** that defines the receivable account for receivable-oriented service agreements. For non-receivable oriented service agreements, this distribution code is typically as follows:

- Charitable contributions. The distribution code is a charity payable account.
- Deposits. The distribution code is a deposit payable account.
- Non-billed budgets. The distribution code is an overpayment payable account.
- Company usage. The distribution code is a company usage expense account.
- Write off. The distribution code is a write-off expense account.
- Payment arrangements. The distribution code is a payment arrangement receivable account.

For more information about GL accounts, refer to [The Source Of GL Accounts On Financial Transactions](#).

Select the **Revenue Class** associated with the SA Type (and its service agreements). The revenue class may affect the revenue account(s) generated by the service agreement's rate.

Refer to [Rate Component – GL Distribution](#) for more information about revenue class.

Turn on **Start Options Required** if you want to force a customer service rep to choose a start option when they start service for this SA Type (on the Start Account window). If this switch is off and a rate is required for the SA Type, the system defaults the SA type's default rate on new service agreements. The default rate is defined on the SA Type – Rate page.

Refer to [Setting Up Start Options](#) for more information about the pros and cons of requiring start options.

Select the **Pay Segment Type** that defines how payment segments linked to service agreements of this type affect:

- The service agreement's payoff and current balances

For more information about payment segment types, refer to [Setting Up Payment Segment Types](#).

When a tender is canceled, a cancellation reason must be supplied. If the cancellation reason indicates a NSF (non sufficient funds) charge should be levied, the system invokes the Levy an NSF Charge algorithm specified on the tender's account's [customer class](#). Because adjustments must be linked to a service agreement, the algorithm must determine the appropriate service agreement to use to levy the adjustment based on business rules. The charge is levied using the **NSF Adjustment Type** of the appropriate service agreement's SA type.

Warning! You must specify adjustment type profiles on the SA type (on the Adjustment Type window) before adjustment types will appear in the above drop downs.

For more information about adjustment types, refer to [Setting Up Adjustment Types](#). For more information about cancellation reasons, refer to [Setting Up Payment Cancellation Reasons](#).

Select the **Payment Priority**. This field is available for use by the algorithms that distribute partial payments amongst an account's service agreements. Higher priority service agreements will have their debt relieved before lower priorities. Refer to [Distribution Based on Payment Priority](#) and [Delinquent Payment Distribution Algorithm](#) for information about payment distribution algorithms that use this field.

Note. The values for this field are customizable using the Lookup table. This field name is PAY_PRIORITY_FLG.

For more information about distribution priority, refer to [Distributing A Payment Amongst An Account's Service Agreements](#).

Select the **Delinquent Payment Priority**. This field is available for use by the algorithms that distribute partial payments amongst an account's service agreements. Higher priority service agreements will have their debt relieved before lower priorities. Refer to [Delinquent Payment Distribution Algorithm](#) for information about a payment distribution algorithm that uses this field.

Note. The values for this field are customizable using the Lookup table. This field name is DEL_PRIORITY_FLG.

Turn on **Do Not Overpay** if the system is not allowed to distribute an overpayment to this type of service agreement (i.e., the service agreement is not allowed to have a system-created credit balance). This field is available for use by algorithms that distribute overpayments. Refer to [Overpayments Held On Highest Priority Service Agreement](#) for information about an overpayment algorithm that uses this field.

Turn on **Late Payment Charge** if the system should generate a late payment charge for this type of service agreement if payment is not received on time. If this is turned on, you must define the **LPC Calc. Algorithm** used to calculate the late payment charge amount. Refer to [Defining Late Payment Charge Options](#) for more information about late payment charges. Examples of algorithm types used for calculating late payment charges are [BILPC-SPRC](#) and [BILPC-TOTAL](#).

Define the **Adjustment Type (Synch Curr)** that will be used to synchronize (make equal) the current amount with the payoff amount on a service agreement of this type. This type of processing happens as follows:

- Most [write-off](#) algorithms that perform financial efforts (e.g., writing off debt), will issue an adjustment of this type if the service agreement's current and payoff balances are not equal.
- If a user stops a customer on a [budget plan](#), the system issues adjustments of this type to synchronize the customer's current and payoff balances.
- If a user stops a service agreement covered by a [non-billed budget](#), the system issues adjustments of this type to synchronize the customer's current and payoff balances.
- If a [cancellation of a bill segment](#) occurs after a customer has stopped participating in a budget plan, an adjustment of this type is issued to synchronize the imbalance created when the bill segment's financial transaction is canceled.

Turn on **CIAC SA Type** and specify an appropriate **CIAC Refund Process** if service agreements of this type are used to bill for Contribution In Aid of Construction (CIAC) charges. Refer to [CIAC Segmentation](#) for more information.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI SA TYPE](#).

SA Type - Detail

Open **Admin Menu**, **SA Type** and navigate to the **Detail** tab to define additional details about a given SA type.

Description of Page

Turn on **Display As Alert** if Control Central should display an alert if an account has a service agreement of this type that isn't **Closed** or **Canceled**. If this switch is on, also enter the **Alert Information** to appear on Control Central. We recommend only using this feature on unusual SA types (e.g., payment arrangements, write-offs) so that a CSR is not presented with an alert for every SA type.

If this SA Type is used for any of the **Special Roles** defined in the drop down, indicate which one. Valid values are: **Billable Charge**, **Cash Deposit**, **Interval**, **Loan**, **Non-billed Budget**, **Payment Arrangement**, **Write Off**, **Bill Determinants Required**. This information is used on windows with functionality that can only be used by service agreements used for specific roles. For example, the Billable Charge window group can only reference **Billable Charge** service agreements.

If Special Role is **Cash Deposit**, you must define the **Deposit Class** of the deposit. You should also define a **Deposit Class** on every SA type to which a given deposit can be distributed.

Refer to [What Do Deposit Classes Do?](#) for more information.

If the Special Role is **Loan**, you must also define the following fields:

- Use the **Interest Bill Factor** to define the bill factor code for the loan interest rate.
- Use **Override Interest Flag** to indicate whether the interest rate defined on the interest bill factor may be overridden at the SA level. If you select **Allowed**, the interest rate may be overridden by a contract value on a start option or the SA.
- Use the **Loan A/R Distribution Code** to define the distribution code to be used when posting the short-term receivable amount to the general ledger (the normal distribution code is used for the long-term receivable). If the normal distribution code is the same as the **Loan A/R Distribution Code**, the SA type does not differentiate between long- and short-term receivables. If the two distribution codes are different, the SA type differentiates between long- and short-term receivables.

Refer to [Defining Loan Options](#) for more information about **Interest Bill Factor**, **Override Interest Flag** and **Loan A/R Distribution Code**.

If the Special Role is **Non-billed Budget**, you must also define the following:

- Use **Adjustment Type (Xfer)** to specify the type of adjustment used to the transfer accumulated credit from the non-billed budget SA to the SAs covered by the non-billed budget when the account is billed or the non-billed budget SA is stopped.
- Use the **Non-billed Budget Monitoring** to specify whether the non-billed budget is monitored by the account debt monitor.

If the SA type is defined as **Eligible for Non-billed Budget**, you must also define the following:

- Use **Adjustment Type (Current=0)** to specify the type of adjustment used to set the service agreement's current balance to zero when a service agreement of this type is linked to an active, monitored non-billed budget.

If the Special Role is **Payment Arrangement**, you must also define two adjustment types:

- Use **Adjustment Type (Xfer)** to specify the type of adjustment used to transfer funds from the delinquent service agreements to the payment arrangement service agreement.

- Use **Adjustment Type (Current=0)** to specify the type of adjustment used to set the payment arrangement's current balance to zero after funds have been transferred.

If the Special Role is **Write Off**, you must also define the following adjustment types:

- Use **Adjustment Type (Xfer)** to specify the type of adjustment used to transfer funds from the uncollectable service agreements to the write off service agreement.

Warning! You must specify adjustment type profiles on the SA type (on the Adjustment Type window) before adjustment types will appear in the above drop downs.

The **Stop Option Flag** can be used to automatically stop a service agreement when all other service agreements of an account are stopped. Valid values are: **Automatically Stop SA**. When all service point related service agreements of an account are stopped then all additional service agreements where the **Stop Option** is set to **Automatically Stop SA** will also be stopped. This is useful, for example, with charitable contribution service agreements. The charity service agreement should be stopped when all utility related services are stopped.

Refer to [The Lifecycle of a Service Agreement](#) for more information on the **Stop Option**.

Turn on **One Time Charge** if this SA type is used for one-time invoices. When a one-time invoice service agreement is created, the system sets the stop date of the SA to be equal to the start date.

Turn on **Sub SA** if this SA type is used to define the business rules for sub service agreements.

Refer to [Sub Service Agreements](#) for more information about sub service agreements.

Renewal of SAs of this type may be **Optional**, **Not Allowed** or **Required** depending on your business processes. If renewal is not allowed, the SA expires on the expiration date. Renewal treatment is an important consideration for SAs that require an expiration date, such as [non-billed budget SAs](#).

If renewal is required or optional, specify the **Days Before Expiration for Renewal**. Note that currently this is only used by non-billed budgets to calculate the renewal date based on the expiration date.

If the Special Role is **Non-billed Budget**, **Non-billed Budget Monitoring** must indicate whether the non-billed budget is monitored by the account debt monitor.

Refer to [Credit and Collections and Non-billed Budgets](#) for more information about monitoring non-billed budgets.

Where Used

The alert information is used by Control Central to alert a CSR when unusual service agreements exist for an account. Refer to [Control Central – Main](#) for more information.

Only SA types designated as being **Billable Charge** may have billable charges linked to them. Refer to [Maintaining Billable Charges](#) for more information.

Only SA types designated as being **Cash Deposit** are processed by the various deposit-related background processes (e.g., interest calculation, automatic refund, etc.). Refer to [The Big Picture Of Deposits](#) for more information.

Only SA types designated as being **Interval** may define Contract Option Types, Profile Relationship Types and TOU Map Types. Refer to [Designing Your SA Interval Billing Options](#) for more information. This role also ensures that a service agreement of this type defines the cutoff time and start day option required by billing. Refer to [Start and End Times for Billing](#) for more information.

Only SA types designated as **Loan** are used to define the loan terms for a loan SA. Refer to [Loans](#) for more information.

Only SA types designated as **Non-billed Budget** may be used to set up non-billed budgets. Refer to [Non-billed Budgets](#) for more information.

Only SA types designated as being **Payment Arrangement** may be used on the payment arrangement window group. Refer to [Setting Up Payment Arrangements](#) for more information.

Only SA types designated as being **Write Off** may be specified as the write off SA type on distribution codes. Refer to [Setting Up Distribution Codes](#) for more information.

Only service agreements whose SA type is designated as being **Write Off** appear on the Write Off SA's query. Refer to [Write Off – Write Off SAs](#) for more information.

SA Type - Billing

Open **Admin Menu, SA Type** and navigate to the **Billing** tab to define how the system manages bill segments for service agreements of a given SA type.

Description of Page

Turn on **Eligible for Billing** if the system should create bill segments for service agreements of this type. This will typically be turned on for all service agreements except for those used to hold write-off amounts or to levy one-off adjustments.

Define the minimum number of days a bill segment (other than the final segment) must span using **Minimum Days for Billing**. This is useful to prevent initial bill segments that span only a few days.

For more information about minimum days, refer to [Preventing Short Bill Segments](#).

Select the **Bill Segment Type** that controls both how bill segments for this SA Type will be created and how the related financial transaction affects the general ledger and the customer's debt.

For more information about bill segment types, refer to [Setting Up Bill Segment Types](#).

Use **Default Description on Bill** to define the verbiage that should print on the customer's bill.

Rates overwrite this description. The Default Description on Bill is not applicable for service agreements whose charges are calculated using a rate. Why? Because the description that appears on the bill segment is defined on the rate schedule's rate version.

Billable charges overwrite this description. The Default Description on Bill is not applicable for service agreements whose charges are calculated using a billable charge. Why? Because the description that appears on the bill segment is defined on the billable charge.

Use the **Billing Processing Sequence** if you need to control the order in which service agreements linked to this SA type are processed by billing and interval data creation processes.

Refer to [Processing Sequence Considerations](#) for more information.

Use **Bill Print Priority** to define the order in which the SA type's bill segments should appear on bills (relative to the other SA types that appear on a bill).

Note. The values for this field are customizable using the Lookup table. This field name is BILL_PRT_PRIO_FLG.

Use **Max Bill Threshold** if you want the system to generate a bill error when a bill segment is produced in batch that exceeds a given value. These bill errors will appear on the standard billing queries and To Do lists. If, after reviewing the high value bill segment, an operator truly intends to send the bill out, they should regenerate the bill. Refer to [How To Correct A Bill Segment That's In Error](#) for more information.

Warning! The value entered in this field will DEFAULT onto service agreements of this type when they are first created. An operator may change the default value on a service agreement in case a specific customer has unusually high bills that continually error out. It's important to be aware that if you change the value of High Bill Amount on an SA type and there already exist service agreements of this type, the existing service agreements will contain the original value (the new value on the SA type will not be propagated on the existing service agreements).

Use **Graph Unit Of Measure** to define the unit of measure of the graphed consumption on the bill (if any).

Turn on **Allow Estimates** if the system is allowed to generate estimated consumption if meter reading(s) cannot be found at billing time. This value is defaulted onto service agreements and can be overridden on an individual service agreement.

For more information about estimated consumption, refer to [The Theory Behind Consumption Estimation](#).

Turn on **Characteristic Premise Required** if a characteristic premise must be linked to the service agreement when the service agreement is activated. The characteristic premise is used to define the taxing authorities associated with the service agreement's bill segments. It is also used to identify where the service agreement's service is located on various windows.

For more information about how characteristic premise is used, refer to [An Illustration Of A Bill Factor And Its Characteristics](#).

Use the **Initial Start Date Option** to control how billing should calculate the consumption period for the very first bill for service agreements of this type. This field is not applicable for sub SA types or SA types with a special role of **Billable Charge**. Valid values are **Include First Day**, **Add 1 Day Always** and **Add 1 Day for Back-to-back**. Refer to [Initial Consumption Period Considerations](#) for more information.

Non-metered service agreements may have the end date of their bill segments defined on a user-maintained bill period schedule. This option is used when bill segments must fall on strict calendar boundaries (e.g., quarterly bills that end on the last day of the quarter). If this SA type should be billed like this, select **Use Bill Period** in the **Use Calendar Billing** field. When this option is used, you must define the **Bill Period** whose schedule defines the bill segment end dates.

For more information about bill period schedules, refer to [Designing Bill Periods](#). For more information about other bill end date methods, refer to [Ways To Control The End Date Of A Bill Segment](#).

Instead of the **Use Bill Period** method, non-metered service agreements may have their bill segment end date based on the first day of service. For example, if service started on the 16th of some month, the ongoing bill segments will start on roughly the 16th of each month. This option is frequently used to bill for garbage or cable service. If this SA type should be billed like this, select **Anniversary Future Billing** or **Anniversary Past Billing** in the **Use Calendar Billing** field. When either option is used, you must define the **Anniversary Bill Frequency**. This frequency defines the amount of time between bill segments.

For more information about anniversary billing, refer to [Using The Anniversary Method](#). For more information about other bill end date methods, refer to [Ways To Control The End Date Of A Bill Segment](#).

Total Bill Amount indicates whether service agreements of this type can use the total amount to bill field on the service agreement page. Valid values are **Not Allowed** and **Required**. Only SA types used to bill for deposits or loans should have this field set to **Required**.

If **Required** is selected, you must enter the **Total Amount To Bill Label**. The **Total Amount To Bill Label** defines the label that prefixes the total bill amount on the service agreement page for service agreements of this SA type.

For more information about total amount to bill and deposit service agreements, refer to [Total Amount To Bill](#). For more information about total amount to bill and loan service agreements, refer to [Setting Up The System To Enable Loans](#).

Recurring Charge indicates whether service agreements of this type can use the recurring charge field on the service agreement window. Valid values are **Not Allowed**, **Optional** and **Required**. If either **Optional** or **Required** are used, you must enter:

- **Recurring Chg Amt Label**. This defines the label that prefixes the recurring charge amount on the service agreement window for service agreements of this SA type.
- **Recurring Charge Frequency**. This defines the following:
 - Specifies the frequency at which the Recurring Charge Amount specified on service agreements of the SA Type is to be billed.

- Serves as the basis for proration of the Recurring Charge Amount.
- Specifies the frequency at which service agreements of the SA Type without a rate and/or meters will be billed.

For more information about how to use the recurring charge information, refer to [Charitable Contribution Segmentation](#), [Merchandise Segmentation – Installment Billing](#), [Deposit Segmentation – Installment Billing](#), [Payment Arrangement Segmentation](#), [Budget Billing Segmentation](#), [The Terms Of A Loan Are Stored On A Service Agreement](#).

Turn on **Eligible for Budget** if service agreements of this type can participate in budget billing. If this switch is turned on, then you must define the **Adjustment Type (Synch Current)** that will be used to synchronize (make equal) the current amount with the payoff amount on a service agreement of this type when a budget is cancelled. (The Adjustment Type (Synch Current) field is on the main page.)

Refer to [Budget Billing](#) for more information about budgets in general. Refer to [Budget Billing Segmentation](#) and [Designing and Defining Budget Plans](#) for more information.

Set the **Eligible for Non-billed Budget** flag to **Eligible for Non-billed Budget** if you want SAs of this type to be eligible to be covered by a non-billed budget. If this flag is set to **Eligible for Non-billed Budget**, you must also define the **Adjustment Type (Current = 0)** field (on [SA Type - Detail](#)).

Refer to [Current Amount For SAs Covered By A Non-billed Budget](#) and [SA Types for SAs Covered by Non-billed Budgets](#) for more information.

Contents

[Require Total Amount Switch versus Bill Segment Algorithm](#)
[Allow Recurring Charge Switch versus Bill Segment Algorithm](#)

Require Total Amount Switch versus Bill Segment Algorithm

The following table shows valid combinations of the SA type's required total amount switch and the bill segment creation algorithm defined on the SA type's bill segment type. If N/A appears in a cell, the combination is not supported in the system. Otherwise, we list typical types of service agreements that will use a combination.

SA Type Require Total Amount Switch →	Not allowed	Required
Bill Segment Create Algorithm ↓		

Apply Rate	Metered services. Lamp services. Misc item services. Company usage. Misc recurring charges whose value is specified in a rate or is taxable.	N/A
Recurring Charge With Auto Stop	Payment arrangements. Merchandise installment plans. Zero-interest loans.	N/A
Recurring Charge For Amount To Bill	N/A	Deposit installment plans.
Recurring Charge	Charitable contributions.	N/A
Billable Charge	One time invoices. Pass through charges	N/A
Loan	N/A	Loans.

Allow Recurring Charge Switch versus Bill Segment Algorithm

The following table shows valid combinations of the SA type's allow recurring charge switch and the bill segment creation algorithm defined on the SA type's bill segment type. If N/A appears in a cell, the combination is not supported in the system. Otherwise, we list typical types of service agreements that will use a combination.

SA Type Recurring Charge Switch → Bill Segment Create Algorithm ↓	Not allowed	Optional	Required
Apply Rate	Metered services – no budget Lamp services Misc item services. Company usage. Misc recurring charges whose value is specified in a rate or is taxable.	Metered services – budget optional	Metered services – budget required
Recurring Charge With Auto Stop	N/A	Payment arrangements. Merchandise installment plans. Zero-interest loans. SEE NOTE!	Payment arrangements. Merchandise installment plans. Zero-interest loans. SEE NOTE!

Recurring Charge For Amount To Bill	N/A	Deposit installment plans. SEE NOTE!	Deposit installment plans. SEE NOTE!
Recurring Charge	Charitable contributions	N/A	N/A
Billable Charge	One time invoices. Pass through charges	N/A	N/A
Loan	N/A	N/A	Loans.

Note. Most recurring charge SA types require a recurring charge amount on their service agreements. However, the above matrix indicates you can have recurring charge SA types where this value is optional. Why? A special algorithm exists in billing that says if the recurring charge amount is 0 (zero) the system will bill the remaining payoff balance or total amount to bill. This algorithm exists so that you can easily bill the amount in one lump sum (i.e., don't bill it in installments).

Where Used

The billing information is used when the system creates a bill segment for service agreements of this type.

SA Type - Rate

Open **Admin Menu**, **SA Type** and navigate to the **Rate** tab to define the rates that may be referenced on service agreements of a given type.

Description of Page

Turn on **Rate Required** if the bill segment creation algorithm for the SA type expects a rate schedule to be referenced on service agreements of this type.

For more information, refer to [Rates](#).

Define the date the system uses when selecting an effective-dated rate (from the service agreement's rate history) using **Rate Selection Date**. Selecting **Bill Start Date** will cause the system to use the rate effective on the first day of the bill segment's [consumption period](#). Selecting **Bill End Date** will cause the system to use the rate effective on the last day of the bill period. Selecting **Accounting Date** will cause the system to use the rate effective on the accounting date associated with the bill.

If the **Contract Management** module is not [turned off](#), which includes [umbrella agreement management](#), indicate the **Rate Source**. **Check SA Only** indicates that the rate schedule currently in effect for the service agreement should always be used. **Check TOS First, then SA** indicates that if the service agreement is linked to a terms of service record, the service agreement's rate could be [overridden by one linked to the terms of service](#) record.

The information in the **Rate Schedules** collection defines the rates that may be referenced on service agreements of this type. The following fields are required for each SA Type:

Rate Schedule

Specify the primary rate schedule; its description is displayed adjacent.

Use As Default

Turn on this switch for the rate to be defaulted on new service agreements.

Rate Required versus Bill Segment Algorithm

The following table shows appropriate combinations of the SA type's rate required switch and the bill segment creation algorithm defined on the SA type's bill segment type. If N/A appears in a cell, the combination is not applicable. Otherwise, we list typical types of service agreements that will use a combination. Be aware that no cross validation exists between the rate required switch and the bill segment creation algorithm when setting up the SA type.

SA Type Rate Required Switch →	Not allowed	Rate required on SA
Bill Segment Create Algorithm ↓		
Apply Rate	N/A	Metered services. Lamp services. Misc item services. Company usage.
Recurring Charge With Auto Stop	Payment arrangements. Merchandise installment plans. Zero-interest loans.	N/A
Recurring Charge For Amount To Bill	Deposit installment plans.	N/A
Recurring Charge	Charitable contributions.	N/A
Billable Charge	One time invoices.	Billable charges that require a rate to add-on extra charges (like taxes) or billable charges where the consumption is interfaced and the system is responsible for calculating the charges.
Loan	Loans.	N/A

Where Used

This information is used to default and validate the rate specified on a service agreement. Refer to [Service Agreement – Rate Info](#) for more information.

SA Type - SP Type

Open **Admin Menu**, **SA Type** and navigate to the **SP Type** tab to define the service point types that may be referenced on service agreements of a given type.

Description of Page

Turn on **Service Points Required** if at least one service point should be linked to service agreements of this type in order to properly bill the service agreements.

The information in the **SP Types** collection defines the service point (SP) types that may be referenced on this SA type's service agreements. The following fields are required for each SA Type:

SP Type	Specify the SP type; its description is displayed adjacent.
----------------	---

For more information about SP types, refer to [Designing SP Types](#).

Where Used

This information is used to validate the types of service points linked to a service agreement. Refer to [Service Agreement – SA / SP](#) for more information.

SA Type - Adjustment Profiles

Open **Admin Menu**, **SA Type** and navigate to the **Adj Profile** tab to define the adjustment profiles that define adjustment types that may be referenced on service agreements of a given type.

Description of Page

Define the **Adjustment Type Profiles** that, in turn, define adjustment types that may be referenced on service agreements of a given type.

For more information about adjustment type profiles, refer to [Setting Up Adjustment Type Profiles](#).

Where Used

This information is used to validate the adjustments linked to the service agreement. Refer to [Adjustments – Main Information](#) for more information.

SA Type - C&C

Open **Admin Menu**, **SA Type** and navigate to the **C&C** tab to maintain attributes that affect how the system severs the service agreement when collection attempts fail.

Refer to [Designing Your Severance Procedures](#) for more information.

Description of Page

Select the **Debt Class** associated with the SA Type. Any debt on a service agreement of this SA Type will be categorized under this debt class.

Select the **Write Off Debt Class Code** associated with the SA Type. Any debt on a service agreement of this SA Type will be categorized under this debt class during write-off processing.

Write Off Debt Class vs. Regular Debt Class. It's important to be aware that a SA type references both a regular debt class and a write-off debt class. The regular debt class controls the collection criteria applied against an account's service agreements. The regular debt class is also used to segregate an account's outstanding balance on several queries in the system. The write-off debt class controls the write-off criteria applied against an account's stopped service agreements. The reason the system supports two different debt classes is because you may categorize your service agreements differently when you try to collect overdue debt versus when you write-off debt. Refer to [The Big Picture Of Write Off Processing](#) for more information.

The information in the **Severance Criteria** collection defines the SA Type's severance criteria. Severance criteria define the severance process to be executed for service agreements of a given SA type. The severance process may differ depending on some attribute of the customer or premise. For example, you may have a different severance process if the customer has life support equipment.

The following information is not intuitively obvious. Refer to [Designing Your Severance Procedures](#) for more information.

The following fields are required for each instance:

Priority

The priority controls the order in which the system determines if the severance process should be applied (the first severance process whose algorithm applies is used). Higher priorities are checked before lower priorities.

Note. The values for this field are customizable using the Lookup table. This field name is CRIT_PRIO_FLG. Be aware that this field is used for multiple tables: [Collection Class Control](#), [Severance Criteria](#), [Write Off Control](#) and [Workflow Process Profiles](#).

Severance Criteria Algorithm

Select the algorithm to be used to check if the severance process should be initiated for service agreements of this type. If a condition is met, a severance process is created using the associated severance process template.

If you haven't done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that determines if the severance of a service agreement should be processed using the associated **Severance Process Template**. Click [here](#) to see the algorithm types available for this plug-in spot.

Important! You must have at least one entry in this collection otherwise the system will not start a severance process when a service agreement of this type needs to be stopped due to non payment. This entry should have the lowest priority code and should reference a **Severance Criteria Algorithm** that references the [SV CRIT DFLT](#) algorithm type.

Severance Process Template

Specify the severance process template to use to sever the service agreement; its description is displayed adjacent.

Where Used

The debt class has multiple uses:

- The system summarizes an account's debt by debt class on [Account – Main Information](#) and [Account – Financial Balances](#).
- Debt class is one component that controls how the system analyzes an account's overdue debt (the others are the account's collection class and currency). Refer to [Different Collection Criteria For Different Customers And Different Debt](#) for more information.
- Write off debt class is one component that control how the system writes off an account's stopped service agreements. Refer to [Different Write-Off Criteria For Different Customers And Different Debt](#) for more information.

The severance criteria are used when a collection event is activated that indicates that service should be severed.

SA Type - Billable Charge Template

Open **Admin Menu**, **SA Type** and navigate to the **BC Template** tab to define the billable charge templates that can be used on service agreements of a given type.

Only billable charges have billable charge templates. Only service agreements that are defined as Billable Charges (in the Special Role on the Details window) may use the grid on this window.

Description of Page

The information in the **Billable Charge Template** collection defines the SA Type's permissible billable charge templates. A billable charge template contains the default bill lines, amounts and distribution codes used to levy a one-off charge. The following fields are required for each template:

Billable Charge Template

Specify the billable charge template. Its description is displayed adjacent.

Use As Default

Turn on this switch for the template to be defaulted on new billable charges linked to service agreements of this type (if any).

For more information about billable charge templates, refer to [Setting Up Billable Charge Templates](#).

Where Used

This information is used to limit the billable charge templates that can be used for a given SA type.

SA Type - Characteristics

To define characteristics common to all service agreements of a given type, open **Admin Menu**, **SA Type** and navigate to the **Characteristics** tab.

Description of Page

Use the **Characteristics** collection to define characteristics common to all service agreements of this type. Turn on the **Required** switch if the **Characteristic Type** must be defined on service agreements of a given type. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on. Use **Sequence** to control the order in which characteristics are defaulted.

Note. You can only choose characteristic types defined as permissible on a SA Type record. Refer to [Setting Up Characteristic Types & Their Values](#) for more information.

SA Type - Algorithms

Open **Admin Menu**, **SA Type** and navigate to the **Algorithm** tab to define the algorithms that should be executed for service agreements of a given type.

Description of Page

The grid contains **Algorithms** that control important functions in the system. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (descriptions of all possible events are provided below).
- Specify the **Sequence** and **Algorithm** for each system event. You can set the **Sequence** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

Warning! These algorithms are typically significant processes. The absence of an algorithm may prevent the system from operating correctly.

The following table describes each **System Event** for which you can define algorithms.

System Event	Optional / Required	Description
<i>Bill Completion</i>	Optional	These algorithms are executed whenever a bill is completed for an account that contains a non-canceled service agreement of this type. The following situations necessitate the definition of a completion algorithm on an SA type:

		<p>- As explained under Technical Implementation of A/R Transfer and Technical Implementation of Routing Billable Charges To Service Providers, when a bill is completed, the system needs to set up the data necessary to interface any "master" SA's charges to the service provider and to transfer the receivable balance from the customer to the service provider. The system will only do this if you specify an appropriate algorithm on the master SA types.</p> <p>- As explained under Billing For SAs Covered By The Non-billed Budget, when a bill is completed for accounts that have a non-billed budget SA, the system needs to distribute the non-billed budget's credit balance to the covered SAs. The system will only do this if you specify an appropriate algorithm on the NBB SA types.</p> <p>- As explained under Overpayment Segmentation, when a bill is completed, the system may apply an excess credit from a prior overpayment to an account's service agreements.</p> <p>Note. Algorithms of this type are called for all non-<i>Canceled</i> service agreements, regardless of whether or not they are billed. If your algorithms should only be processed under certain conditions (for example, only process this algorithm for <i>Active</i> service agreements), then it is the responsibility of the algorithm to check the conditions before continuing.</p> <p>Click here to see the algorithm types available for this system event.</p>
Break NBB SA	Optional	<p>These algorithms are executed when a non-billed budget is manually stopped via the non-billed budget maintenance page.</p> <p>Click here to see the algorithm types available for this system event.</p> <p>Note that the <i>Payment Arrangement</i> algorithm and the <i>Break Pay Arrangement</i> algorithm are mutually exclusive.</p>
Break Pay Arrangement	Optional	<p>These algorithms are by executed by severance events when the event is created for payment arrangement SAs. This algorithm should be specified on SA types with a special role of <i>Payment Arrangement</i> to perform special actions that take place when a customer breaks a payment arrangement. Refer to Monitoring Payment Arrangements for more information about breaking payment arrangements.</p> <p>Click here to see the algorithm types available for this system event.</p>
Budget Eligibility	Optional	<p>These algorithms are executed when determining in a service agreement is eligible for budget. Algorithms of this type are only applicable on SA types that are marked as eligible for budget and may be used to override that setting and indicate that the service agreement is not eligible.</p> <p>For example, maybe service agreements in a certain rate are not eligible. Or perhaps service agreements with a given characteristic value are not eligible.</p> <p>Click here to see the algorithm types available for this system event.</p>
Cut Process Rule	Optional	<p>These algorithms are executed to create a cut process for service agreements of this type. Refer to The Big Picture Of Cut Processes for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
FT Freeze	Optional	<p>These algorithms are executed whenever a financial transaction is frozen that is linked to a service agreement of this type. The following situations necessitate the definition of an FT freeze algorithm:</p>

		<p>- As explained under Technical Implementation of Routing Consumption To Service Providers, when a master SA's bill segment is frozen, the system must check if there are any service providers who need the bill segment's consumption. If so, it sets up the data necessary to interface the master SA's consumption (snapshot on the bill segment) to the service provider(s). The system will only do this if you specify an appropriate FT Freeze Algorithm on the master SA types.</p> <p>- As explained under Technical Implementation of Paying The Service Provider, when a financial transaction (FT) is frozen that is associated with a sub SA, the system must check if this FT should trigger the "payment" of a service provider. If so, it has to create an adjustment to increase how much we owe the service provider. The system will only do this if you specify an appropriate FT Freeze Algorithm on the sub SA types.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Landlord Reversion</i>	Optional	<p>These algorithms are used to create, update or cancel a service agreement for a landlord when service is started, stopped or updated for a premise that references a landlord agreement.</p> <p>Algorithms of this type are called:</p> <ul style="list-style-type: none"> - When starting a new SA (start initiation) - When stopping an SA (stop initiation) - When canceling a pending start SA - When changing the stop / start dates of back-to-back service agreements <p>Click here to see the algorithm types available for this system event.</p>
<i>Loan Interest Charge</i>	Optional	<p>These algorithms are executed whenever the interest charge needs to be calculated for a loan, such as when the loan amortization schedule is created and when a bill segment is created for a loan SA. This algorithm should be specified on SA types with a special role of <i>Loan</i>. Refer to Defining Loan Options for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Loan Periods and Amount</i>	Optional	<p>These algorithms are executed whenever a user clicks the Calculate button on Loan - Main or on the Start SA Confirmation dialog for a loan SA. It calculates either the number of periodic payments or the payment amount (depending on whether the user specifies the number of payments or the payment amount as input). This algorithm should be specified on SA types with a special role of <i>Loan</i>. Refer to Defining Loan Options for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Loan Schedule</i>	Optional	<p>These algorithms are executed whenever the system needs to create a loan amortization schedule for a loan, such as when you renegotiate the terms of a loan on Loan - Main. This algorithm should be specified on SA types with a special role of <i>Loan</i>. Refer to Defining Loan Options for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Payment Arrangement</i>	Optional	<p>These algorithms are by executed to handle the creation, breaking and canceling of payment arrangement SAs. This algorithm should be specified on SA types with a special role of <i>Payment Arrangement</i> to perform special actions that take place during the lifecycle of a payment arrangement. Refer to Monitoring</p>

		<p>Payment Arrangements for more information about payment arrangements.</p> <p>Note that the <i>Payment Arrangement</i> algorithm and the <i>Break Pay Arrangement</i> algorithm are mutually exclusive.</p>
<i>Payment Freeze</i>	Optional	<p>These algorithms are executed whenever a payment is frozen. The following situations necessitate the definition of such an algorithm:</p> <ul style="list-style-type: none"> - For a loan SA, such an algorithm is required to create a frozen adjustment that transfers any credit balance resulting from an overpayment to the loan's principal balance. Refer to Defining Loan Options for more information. <p>Click here to see the algorithm types available for this system event.</p>
<i>Pre-Bill Completion</i>	Optional	<p>These algorithms are executed immediately prior to bill completion when a bill contains a bill segment for a service agreement whose SA type has such an algorithm. The following situations necessitate the definition of such an algorithm:</p> <ul style="list-style-type: none"> - If you want to delete a bill segment that's in error on the last night of a bill cycle when there are other bill segments that aren't in error, use such an algorithm. <p>Click here to see the algorithm types available for this system event.</p>
<i>Process NBB Scheduled Payment</i>	Optional	<p>These algorithms are executed by the NBB Scheduled Payment Processing background process whenever a scheduled payment is due. If the non-billed budget SA is unmonitored, this algorithm is not called. This algorithm should be specified on non-billed budget SA types to create the necessary adjustments for the non-billed budget SA.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Proposal SA Acceptance</i>	Optional	<p>These algorithms are executed when a proposal service agreement is accepted. Refer to Enabling The Creation Of A Real Service Agreement for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Proposal SA Bill Segment Generation</i>	Optional	<p>These algorithms are executed to generate simulated bills segments for a proposal service agreement. Refer to Enabling The Generation Of Simulated Bill Segments for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>Proposal SA Creation</i>	Optional	<p>These algorithms are executed when a proposal service agreement is created. Refer to Enabling The Automatic Generation Of Billing Scenarios for more information.</p> <p>Click here to see the algorithm types available for this system event.</p>
<i>SA Activation</i>	Optional	<p>These algorithms are executed when a service agreement status changes from <i>Pending Start</i> to <i>Active</i>. It performs any additional activities that are necessary to activate an SA. The following situations necessitate the definition of such an algorithm:</p> <ul style="list-style-type: none"> - If you want to create a customer contact to indicate that a non-billed budget has been activated, use such an algorithm. <p>Click here to see the algorithm types available for this system event.</p>
<i>SA Cancel</i>	Optional	<p>These algorithms are executed when a service agreement status changes to <i>Canceled</i>. It performs any additional activities that are necessary to cancel an SA.</p>

		<p>An example of when you may use this algorithm is that perhaps your business rules dictate that the creation of a payment arrangement should create a credit rating history transaction. When a payment arrangement SA is canceled, the credit rating should be updated with an end date.</p> <p>Click here to see the algorithm types available for this system event.</p>
SA Creation	Optional	<p>These algorithms are executed when a service agreement is created. The following situations necessitate the definition of such an algorithm on an SA type:</p> <ul style="list-style-type: none"> - If you want to create a To Do entry whenever a new service agreement of a given type is added, specify such an algorithm. - If you want to automatically activate SAs of a given type (instead of waiting for the background SA activation process to run), specify such an algorithm. - If you want to create a Workflow Process when a service agreement of a given type is added, specify such an algorithm. <p>Click here to see the algorithm types available for this system event.</p>
SA Information	Optional	<p>We use the term "SA information" to describe the basic information that appears throughout the system to describe a service agreement. The data that appears in "SA information" is constructed using this algorithm.</p> <p>Plug an algorithm into this spot to override the "SA information" algorithm on installation options or the system default "SA information" if no such algorithm is defined on installation options.</p> <p>Click here to see the algorithm types available for this system event.</p>
SA Renewal	Optional	<p>These algorithms are executed by the Service Agreement Renewal background process whenever an SA is due for renewal or when the user clicks the Renew button (for non-billed budgets). It performs any activities that are necessary to renew an SA and returns the new renewal and expiration dates for the SA.</p> <p>Click here to see the algorithm types available for this system event.</p>
SA Stop	Optional	<p>These algorithms are executed whenever a service agreement's status changes from <i>pending stop</i> to <i>stopped</i>. The following situations necessitate the definition of a SA stop algorithm:</p> <ul style="list-style-type: none"> - For non-billed budgets to distribute any remaining credit balance from the non-billed budget SA to the covered SAs, you must specify such an algorithm. - For service credit memberships that have a refundable membership fee, an SA Stop algorithm attempts to refund the fee if this is the last SA linked to the membership that is being stopped. <p>Click here to see the algorithm types available for this system event.</p>
SA Stop Initiation	Optional	<p>These algorithms are executed whenever a service agreement's status becomes <i>pending stop</i>. The following situations necessitate the definition of a stop initiation algorithm on an SA type:</p> <ul style="list-style-type: none"> - As explained under Finalizing Pending Stops, service agreements are normally transitioned from <i>pending stop</i> to <i>stopped</i> by a background process (or manually). For non-billed budget SAs to transition to <i>stopped</i> automatically (without waiting for the background process), you must specify such an algorithm. <p>When does a SA become pending stop? Service agreements typically become <i>pending stop</i> when a user initiates a request to stop service on Start</p>

		<p>Stop - Main. A severance process with an Expire SA severance event causes a service agreement to become pending stop (when the event is executed). Additionally, the Stop Expired Service Agreements background process starts the process to initiate the stop of an SA when the expiration date is on or before the process date.</p> <p>Click here to see the algorithm types available for this system event.</p>
Start Stop Field Work	Optional	<p>These algorithms are executed to create the field activities necessary to start and stop service. Refer to Starting Service and Field Activities and to Stopping Service and Field Activities for a description of when algorithms of this type are called. The following situations necessitate the definition of a start stop fieldwork creation algorithm:</p> <ul style="list-style-type: none"> - If a service agreement has field activities created to start and stop service at its service points, its SA type must have an appropriate start stop field work creation algorithm. <p>Click here to see the algorithm types available for this system event.</p>

SA Type - Billable Charge Overrides

The [BCU2 – Create Billable Charge](#) background process is responsible for creating billable charges for each billable charge upload staging record interfaced into the system. This process will override the values of the various switches referenced on a bill charge upload staging line if the respective service agreement's SA type has an override value for the bill charge upload staging line's billable charge line type.

This information is optional. If you don't need to override the values of a [Billable Charge Line Type](#) you don't need to set up this information.

Open **Admin Menu, SA Type** and navigate to the **BC Upload Override** tab to define override values for a given SA Type / Billable Charge Upload Staging Line Type.

Description of Page

Use the **Billable Charge Overrides** collection to define values to be overridden on billable charge lines uploaded from an external system (refer to the description above for the details). The following switches may be overridden for a given **SA Type** and **Billable Charge Line Type**.

- Use the **Show on Bill** switch to define the value to be defaulted into the Show on Bill indicator on billable charge upload lines that reference this line type.
- Use the **Appear in Summary** switch to define the value to be defaulted into the App in Summary indicator on billable charge upload lines that reference this line type.
- Use **Memo Only, No GL** switch to define the value to be defaulted into the Memo Only, No GL indicator on billable charge upload lines that reference this line type.
- Use **Distribution Code** to define the value to be defaulted into the Distribution Code on billable charge upload lines that reference this line type.

SA Type - Contract Option Type

Open **Admin Menu**, **SA Type** and navigate to the **Contract Option Type** tab to define the contract option types, which are valid for service agreements of a given type.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Contract Options are used by the system to define special options under which certain calculations supporting a contract's rate may be overridden or altered occasionally for specific periods of time. Refer to [Contract Option Background Topics](#) for more information.

Description of Page

If the SA Type's special role is **Interval**, you may define the **Contract Option Types** that are valid for contract options linked to a service agreement of this type.

For more information, refer to [Setting Up Contract Option Types](#).

Where Used

This information is used to validate the contract options linked to the service agreement. Refer to [Service Agreement - Contract Options](#) for more information.

SA Type - Interval Info

Open **Admin Menu**, **SA Type** and navigate to the **Interval Info** tab to define the interval profile relationship types and TOU map relationship types, which are valid for service agreements of a given type.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Description of Page

If the SA Type's special role is **Interval**, you may define the **Interval Profile Relationship Types** that may be linked to service agreements of this type.

For more information about interval profile relationship types, see [Designing Interval Profile Relationship Types](#).

If the SA Type's special role is **Interval**, you may define the **TOU Map Relationship Types** that may be linked to service agreements of this type.

For more information about TOU map relationship types, refer to [Designing TOU Map Relationship Types](#).

Where Used

The interval profile information is used to validate the interval profile relationship types linked to the service agreement. Refer to [Service Agreement - Interval Info](#) for more information.

The TOU map information is used to validate TOU map relationship types linked to a service agreement. Refer to [Service Agreement - Interval Info](#) for more information.

SA Type - NBB Recommendation Rule

Open **Admin Menu**, **SA Type** and navigate to the **NBB Rec'n Rule** tab to define the recommendation rules that are valid for non-billed budget SAs of this type.

Description of Page

If the SA Type's special role is **Non-billed Budget**, you may define the **Recommendation Rules** that are valid on non-billed budget SAs of this type. Check the **Use As Default** box to indicate the default recommendation rule for service agreements of this type.

For more information about non-billed budgets, refer to [Defining Non-Billed Budget Options](#).

Where Used

The non-billed budget recommendation rules are used to recommend the payment amount and payment schedule for non-billed budget service agreements. Refer to [Maintaining Non-billed Budgets](#) for more information.

Setting Up Start Options

Start options save users time and prevent data entry errors because they default many values on a service agreement (e.g., the rate schedule, recurring charge amount, contract riders, contract terms, characteristics, terms and conditions, etc. can all be defaulted onto a service agreement from a start option).

A SA type may have zero or more start options.

- A SA type without start options is usually one that has a very limited number of options. For example, if a SA type has a single valid rate and no customer-specific contract values, you don't need to setup a start option (the SA's default rate can default based on the information defined when you setup the SA type).
- A SA type with multiple start options is one where many different permutations are possible. For example, a SA type that can have multiple rates and each rate can have multiple riders is a good candidate for start options (where each start option will default, for example, a specific rate and set of contract riders).

When the [Start/Stop](#) transaction is used to start service AND the service being started uses a SA type with start options, the user is asked to select one of the start options. The service agreement that's created is populated with fields from the start option.

When the [Order](#) transaction is used to start service, the user selects start options, but only indirectly. It works like this:

- A user selects a "package" of services to start service for a customer.

- A “package” contains one or more start options.
- When a user selects a package, the system creates a service agreement for each start option on the package (and defaults the information on each service agreement from the respective start option).

A start option’s default values may change over time (i.e., the information on a start option is effective-dated). The start service logic uses the version of the start option that is effective on the day service starts.

Start options can cause a great deal of information to be populated on a service agreement. There are several ways to change this default information:

- A user may override this information using the [Service Agreement](#) transaction.
- If the service agreement is in the **pending start** state, you can use the [Start/Stop - Pending SAs](#) page to change the service agreement’s SA type and/or start option.
- If the service agreement is in **active** or **pending stop** states, a button appears on [Service Agreement - Main](#) called **Apply New Start Option**. When pressed, the user is allowed to define a start option and the date its terms become effective on the service agreement. Refer to [Changing A Start Option](#) for the details of this functionality.

The topics in this section describe how to setup start options.

The merge transaction can save setup time. The [Start Options Merge](#) transaction can be used to construct a start option by copying pieces from other start options.

Contents

[Start Option Considerations - Rate-Oriented SA Types](#)
[Start Option Considerations - SA Types That Use Recurring Charges](#)
[Start Option Considerations - Initial Adjustment SA Types](#)
[Start Option Considerations - Interval SA Types](#)
[Start Option - Main](#)
[Start Option - Rate Info](#)
[Start Option - Characteristics & Qty](#)
[Start Option - Contract Option](#)
[Start Option - Interval Info](#)
[Start Option - TOU Contract Value](#)
[Start Option - Terms and Conditions](#)

Start Option Considerations - Rate-Oriented SA Types

To understand the following discussion, you should be familiar with the following concepts:

- SA types that use rates have one or more valid rate schedules. Only these rate schedules may be defined on service agreements of a given type.
- One of a SA type’s rate schedules may be designated as the “default” rate. The system assigns the “default” rate to new service agreements when a CSR doesn’t choose a start option.

- Start options may be used by customer service reps when a service is started for a customer. A start option causes the customer's new service agreement to be populated with a specific rate and contract terms (e.g., contract riders, contract values). The use of start options is not allowed if the service's SA type does not require a start option. The use of start options is required if the service's SA type requires a start option.

Whether or not a SA Type has start options is dependent on the following factors:

- If a SA Type has only one valid rate and the rate doesn't use customer-specific contract terms (e.g., contract riders, contract values), the SA Type does not need any start options. Why? Because the system default's the SA type's default rate on new service agreement when no start option is used at start time.
- If a SA Type has only one valid rate, but under unusual circumstances, it uses customer-specific contract terms, you'll want an option for every situation (both the standard one and the unusual ones).
- If a SA Type has a single rate with a variety of contract terms (which, by definition, are different for each customer), then you'd want a start option for each permissible combination of contract terms. You'd also want to turn on the SA type's Start Option Required switch to make your CSR's pick one of the start options when service is started (rather than let the system use the SA type's default rate).
- If a SA type has multiple valid rates and a variety of contract terms (a combination of the previous two points), you'd need a start option for each permissible combination. You'd also want to turn on the SA type's Start Option Required switch to make your CSR's pick one of the start options when service is started (rather than let the system use the SA type's default rate).

Start Option Considerations - SA Types That Use Recurring Charges

To understand the following discussion, you should be familiar with the following concepts:

- Many SA types use the recurring charge algorithms to generate the bill segments. For example, if you let a customer pay for a \$900 heat pump in 3 installments of \$300, you'd have an SA type called CA/MERCH-I and indicate it uses the recurring charge algorithm.
- When a CSR creates an SA type that uses recurring charge algorithms, they can enter the recurring charge amounts on [Start/Stop Service – Start Confirmation](#) window OR they can specify a start option on this window and let the system populate the recurring charge amount. For example, if you let a customer pay for a \$900 heat pump in 3 installments of \$300, you can set up a start option called HP 3PAY with an installment amount of \$300.
- Depending on the type of service being started, you might also need to generate an adjustment when service is started in order to initialize the total debt. For example, when a customer buys a heat pump we'll need an adjustment issued to realize the entire \$900 of revenue.

The following table provides examples of recurring charge SA types with several typical start options:

CIS Division/ SA Type	Start Option	Adjustment Type	Install Amount	Comments
CA/CHARITY	DONATE \$5		5	This causes a charitable contribution to be created with an installment amount of \$5.

CA/CHARITY	DONATE \$10		10	This causes a charitable contribution to be created with an installment amount of \$10.
CA/PA-REGU	PAY \$10 PM		10	This causes a payment arrangement to be created with an installment amount of \$10.
CA/PA-REGU	PAY \$20 PM		20	This causes a payment arrangement to be created with an installment amount of \$20.
CA/PA-UNRE	PAY \$10 PM		10	This causes a payment arrangement to be created with an installment amount of \$10.
CA/PA-UNRE	PAY \$20 PM		20	This causes a payment arrangement to be created with an installment amount of \$20.
CA/MERCH-I	HP 3PAY	HEATPUMP	300	This causes a merchandise service agreement to be created with an installment amount of \$300. It also causes an adjustment to be issued to realize the \$900 of revenue.
CA/MERCH-I	BBQ 3PAY	BBQ	250	This causes a merchandise service agreement to be created with an installment amount of \$250. It also causes an adjustment to be issued to realize the \$750 of revenue.
CA/DEP-I	PAY \$40 PM		40	This causes a deposit service agreement to be created with an installment amount of \$40. The CSR would be required to define the total deposit amount to be billed over the life of the service agreement on the secondary SA window on the Start Account window.

Start Option Considerations - Initial Adjustment SA Types

To understand the following discussion, you should be familiar with the following concepts:

- Some SA types depend on an adjustment to book their initial debt. For example, the CA/MERCH-I SA type requires an initial adjustment to book the payoff amount for the entire cost of the merchandise. Another example would be a loan service agreement (refer to [Booking The Principal Amount Using An Adjustment](#) for the details).
- When a CSR starts an SA type that requires an initial adjustment, they can create the adjustment immediately after starting service OR then can specify a start option when they start service and let the system generate the adjustment. You could let the system create the adjustment to book the \$900 associated with the heat pump as shown in the previous section.

The following table provides examples of SA types that use adjustments with several typical start options:

CIS Division/ SA Type	Start Option	Adjustment Type	Install Amount	Comments
--------------------------	--------------	--------------------	-------------------	----------

CAMERCH-I	HP 3PAY	HEATPUMP	300	This causes a merchandise service agreement to be created with an installment amount of \$300. It also causes an adjustment to be issued to realize the \$900 of revenue.
CAMERCH-I	BBQ 3PAY	BBQ	250	This causes a merchandise service agreement to be created with an installment amount of \$250. It also causes an adjustment to be issued to realize the \$750 of revenue.
CA/CONNECT	CONNECT	CONNECT		This causes an adjustment to be issued to realize the connection charge.

Start Option Considerations - Interval SA Types

Refer to [Designing Your SA Interval Billing Options](#) for information about setting up start options for interval service agreements.

Start Option - Main

Open **Admin Menu, SA Type Start Option** and navigate to the **Main** tab to define a SA type's start options.

Description of Page

Every start option is uniquely identified by the following fields:

CIS Division & SA Type Enter the Division and SA type to which the start option is linked.

Start Option Enter the unique identifier of the option. Pick something easy to recognize as this will be used by CSRs to pick an option when they start service.

Effective Date Enter the earliest effective date. It should be the same as the earliest effective date of the start option's rate (although it doesn't hurt for it to be earlier). The date defaults to the current date. (The status, below, should be **Active**.)

The remaining fields further describe a start option.

Enter a **Description** for the start option.

Indicate its **Status**. For new start options, the status should be **Active**. When it's no longer applicable, change it to **Inactive**.

Enter the primary **Rate Schedule** that should be defaulted onto service agreements created using this option. Refer to [Start Option Considerations For SA Types That Use Rates](#) for more information.

Note. Only rates that meet the following criteria may be specified: 1) the rate must be defined as valid for the Division / SA type, and 2) the rate must have at least one **Finished** rate version.

For more information about a service agreement's rates, refer to [Service Agreement – Rate Info](#).

Enter the **Adjustment Type** that should be generated, if any, when service is started using this option. Refer to [Start Option Considerations For SA Types That Use Initial Adjustments](#) for more information.

Enter the **Recurring Charge Amount** that should be defaulted onto service agreements created using this option. This field is only visible when the SA type allows recurring charges. In addition, the prompt for this field is defined on the SA type table on the billing window (e.g., it could appear as Payment Amount, Budget Amount, or Installment Amount). Refer to [Start Option Considerations For SA Types That Use Recurring Charges](#) for more information.

Enter the **Currency Code** in which monetary amounts are denominated.

Default Note. The currency code defaults from the [installation record](#).

Enter the **Total Amount to Bill** that should be defaulted onto service agreements created using this option. This is useful to initiate either a loan or a deposit. The prompt for this field is defined on the SA type table on the billing window (e.g., it could be Deposit Amount or Loan Amount).

Use **Number of Payment Periods** to default the number of payment periods of service agreements created using this option. This field is only allowed for SA types with special role of **Loan**. Refer to [The Terms Of A Loan Are Stored On A Service Agreement](#) for more information.

If your SA Type has a special role of **Interval**, then you must enter the **Cutoff Time** and **Start Day Option** that should be defaulted onto service agreements created using this option. For start day option, you may choose **Current Day** or **Previous Day**. Refer to [Start and End Times for Billing](#) for more information about how these fields are used.

If your SA Type has a special role of **Billable Charge**, then you can setup the start option to automatically create a billable charge when a service agreement is created using this start option. For example, you might have a start option that automatically creates a “one-time invoice” service agreement along with a “tree trimming” billable charge. To use this feature you should turn on the **Create Billable Charge** switch and specify the **Billable Charge Template** that will be used to create the billable charge. These fields are only allowed for SA types with special role of **Billable Charge**.

Refer to [Setting Up Billable Charge Templates](#) for more information about templates. Refer to [An Easier Way To Create One Time Charges](#) for an example of how you can setup a campaign with packages that use this type of start option.

Note. The duplicate action in the action button bar enables you to copy another start option. Refer to [Duplicate Button](#) in the system wide standards document for more information.

Start Option - Rate Info

Open **Admin Menu**, **SA Type Start Option** and navigate to the **Rate Info** tab to define the start option's default values for contract riders and contract values.

Description of Page

The information in the **Contract Riders** collection defines the contract riders to be defaulted onto service agreements created using this start option. The following fields are required for each instance:

Bill Factor

The bill factor defines the type of rider. You may only reference bill factors designated as being applicable for contract riders.

Number of Days

The number of days the rider should be in effect. This value is used by the system to set the stop date on the service agreement's contract rider. If the rider has no expiration, set this field to 0. Default note: this field will be set to 0 if left blank.

For more information about a rate's contract riders, refer to [Defining General Bill Factor Information](#). For more information about a service agreement's contract riders, refer to [Service Agreement – Contract Riders](#).

The information in the **Contract Values** collection defines the contract values to be defaulted onto service agreements created using this start option. The following fields are required for each event:

Bill Factor

The bill factor defines the type of value. You may only reference bill factors designated as allowing values in contract terms.

Number of Days

The number of days the value should be in effect. This value is used by the system to set the stop date on the service agreement's contract value. If the value has no expiration, set this field to 0.

Value

The amount of the contract value.

For more information about a rate's contract values, refer to [Defining General Bill Factor Information](#). For more information about a service agreement's contract values, refer to [Service Agreement – Contract Values](#).

Start Option - Characteristics & Qty

Open **Admin Menu**, **SA Type Start Option** and navigate to the **Characteristics & Qty** tab to define the start option's default values for characteristics and contract quantities.

Description of Page

The information in the **Characteristics** collection defines the characteristics to be defaulted onto service agreements created using this start option. The following fields are required for each instance:

Characteristic Type	This defines the type of characteristic. Note: you may only define characteristics valid on service agreements.
Characteristic Value	This defines the characteristic value that will be defaulted.

For more information about a service agreement's characteristics, refer to [Service Agreement – Characteristics](#).

The information in the **Contract Quantity** collection defines the contract quantities to be defaulted onto service agreements created using this start option. The following fields are required for each instance:

Contract Quantity Type	This defines the type of contract quantity.
Contract Quantity	The amount of the contract quantity.

For more information about a service agreement's contract quantities, refer to [Service Agreement – Contract Quantity](#).

Start Option - Contract Option

Refer to [Designing Your SA Interval Billing Options](#) for information about setting up start options for interval service agreements.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Open **Admin Menu**, **SA Type Start Option** and navigate to the **Contract Option** tab to define the start option's contract option default values.

Description of Page

The collection of contract option types will be used to link contract options to a service agreement that will be created with this start option. If the contract option to create for the service agreement will be **SA Specific**, simply indicate the **Contract Option Type**. This will cause a new contract option to be created with this contract option type and the new contract option will be linked to the new SA at start time. If the contract option to create for the service agreement will be **Shared**, indicate the contract option type and the **Contract Option ID**. This will cause the specified contract option to be linked to the new SA at start time.

For more information about contract option types, see [Designing Your Contract Option Types](#).

Start Option - Interval Info

Refer to [Designing Your SA Interval Billing Options](#) for information about setting up start options for interval service agreements.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Open **Admin Menu, SA Type Start Option** and navigate to the **Interval Info** tab to define the start option's interval profile and TOU map default values.

Description of Page

The collection of interval profile information will be used to link interval profiles to a service agreement that will be created with this start option. The **Interval Profile Relationship Type** indicates the value that will be linked to each new SA Profile record. For each interval profile relationship type, define either an **Interval Profile Type** or an **Interval Profile ID** depending on whether the Profile to be linked to the new service agreement should be **SA Specific** or **Shared**. For SA Specific, indicate an interval profile type. This will cause a new interval profile to be created with this interval profile type and the new interval profile will be linked to the new SA at start time. For Shared profiles, indicate the Interval Profile ID. This will cause the specified Profile to be linked to the new SA at start time.

If a **Derivation Algorithm** is linked to the interval profile type, then the algorithm and its **Creation Priority** are displayed.

For more information about interval profile relationship types, see [Designing Interval Profile Relationship Types](#). For more information about Shared profiles vs. SA Specific profiles, see [Common Profiles vs. SA Owned Profiles](#).

The collection of TOU map information will be used to link TOU Maps to a service agreement that will be created with this start option. The **TOU Relationship Type** indicates the value that will be linked to each new SA TOU Map record. For each TOU relationship type, define either a **TOU Map Type** or a **TOU Map ID** depending on whether the map to be linked to the new service agreement should be **SA Specific** or **Shared**. For SA Specific, indicate a TOU map type. This will cause a new TOU map to be created with this TOU map type and the new TOU map will be linked to the new SA at start time. For Shared map, indicate the TOU map id. This will cause the specified map to be linked to the new SA at start time.

If a TOU map creation algorithm is linked to the TOU Map type, then the **Derivation Algorithm** and its **Creation Priority** are displayed.

For more information about TOU map relationship types, see [Designing Your TOU Map Relationship Types](#).

Start Option - TOU Contract Value

Refer to [Designing Your SA Interval Billing Options](#) for information about setting up start options for interval service agreements.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Open **Admin Menu, SA Type Start Option** and navigate to the **TOU Contract Value** tab to define the start option's default TOU contract values.

Description of Page

The **TOU Contract Value** scroll contains information to define when you would like TOU contract values to be defaulted onto service agreements created using this start option

Indicate the **Bill Factor** associated with the TOU contract values. You may only reference bill factors designated as "TOU" bill factor types and where the value may be in contract terms.

Indicate the **TOU Group**, which contains the appropriate collection of TOU codes for service agreements created using this start option.

Indicate the **Number of Days** the value should be in effect. This value is used by the system to set the stop date on the service agreement's contract value. If the value has no expiration, set this field to 0.

Use the Time of Use collection to indicate the TOU **Value** associated with each **Time of Use** codes for the TOU Group.

For more information about TOU contract values, refer to [Customer Specific TOU Values](#).

Start Option - Terms and Conditions

Open **Admin Menu, SA Type Start Option** and navigate to the **Terms and Conditions** tab to define the start option's default terms and conditions.

Description of Page

The information in the grid defines the terms and conditions to be defaulted onto service agreements created using this start option. The following fields are required for each instance:

Terms and Conditions

This is the code that identifies a term and condition (T&C).

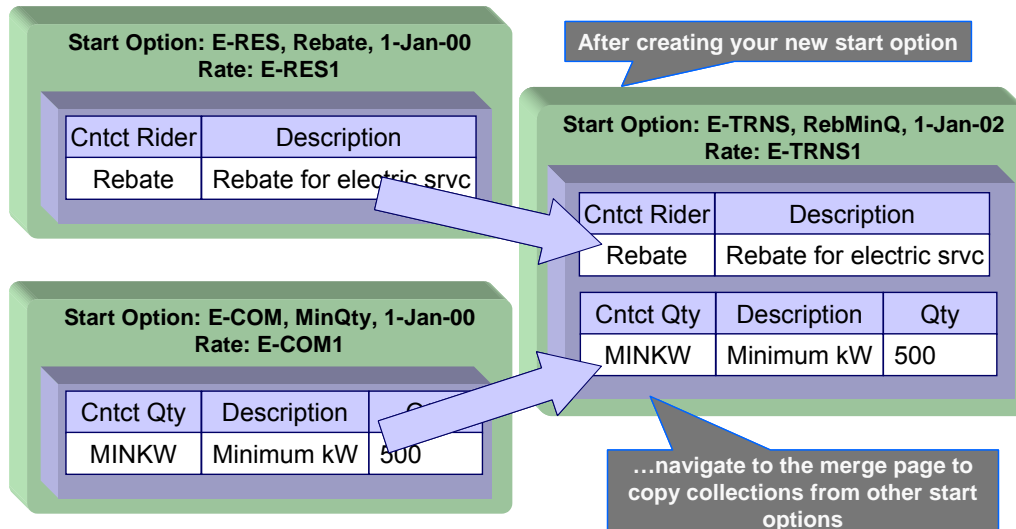
Number of Days

The number of days the T&C should be in effect. This value is used by the system to set the end date on the service agreement's T&C. If the T&C has no expiration, set this field to 0. Default note: this field is set to 0 if left blank.

Start Options Merge

Use this page to modify an existing start option by copying information from other start options. This page may be used to copy records from the contract rider, contract value, contract quantity, characteristic, interval profile, TOU map, contract option and TOU contract value collections from one or more existing start options to another.

Note. The target start option must exist prior to using this page. If you are creating a new start option, you must first go to the [Start Option](#) page to add the new start option and then navigate to the merge page to copy collection information.



Duplicate versus Merge. The [Start Option](#) page has [Duplication](#) capability. You would duplicate a start option if you want to a) create a new start option AND b) populate it with all the information from an existing start option. You would use the start option merge page if you want to build a start option using pieces of one or more start options.

Contents

- [Start Options Merge - Main](#)
- [Start Options Merge - Characteristics and Quantities](#)
- [Start Options Merge - Contract Option](#)
- [Start Options Merge - Interval Info](#)
- [Start Options Merge - Terms and Conditions](#)

Start Options Merge - Main

Open **Admin Menu, SA Type Start Options Merge**.

Description of Page

Select the **Original Start Option**, which is the target for merging the start option collection information.

Select the **Merge From Start Option**, which is your template start option to copy the collections from.

Note. You may only copy information from one Merge From start option at a time. If you wish to copy information from more than one start option, select the first Merge From start option, copy the desired records, Save, then select the next Merge From start option.

The left portion of the page will display any existing records in the collections for the original start option. The right portion of the page will display the existing records in the collections for the Merge From start option.

You may use the **Copy All** button to copy all the records in all the collections from the Merge From start option to the Original start option. If you do not choose to copy all, you may copy records individually as described below.

The left portion of the **Contract Riders** collection initially displays existing contract riders linked to the original start option. In the **Merge Type**, you will see the word **Original**, for any of these records. The **Bill Factor** and **Number of Days** for each contract rider is displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

The left portion of the **Contract Values** collection initially displays existing contract values linked to the original start option. In the **Merge Type**, you will see the word **Original**, for any of these records. The **Bill Factor**, **Number of Days** and **Value** for each contract value is displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

The topics, which follow, describe how to perform common maintenance tasks:

Contents

- [Removing A Row From A Grid](#)
- [Adding A New Row To A Start Option](#)
- [Removing An Uncommitted Row From A Start Option](#)

Removing A Row From A Grid

If you wish to remove a record linked to the Original start option, click the “-” button to the left of the record.

Adding A New Row To A Start Option

You may move any of the records from the Merge From start option to the original start option by selecting the left arrow adjacent to the desired row. Once a record is moved it will disappear from the Merge From information and appear in the Original information with the word **Merge** in the Merge Type column.

Removing An Uncommitted Row From A Start Option

If you have copied a row across by mistake, you may remove it by clicking on the right arrow adjacent to the appropriate record.

Refer to [Editable Grid](#) in the system wide standards documentation for more information about adding records to a collection by selecting from a list.

Start Options Merge - Characteristics and Quantities

Open **Admin Menu, SA Type Start Options Merge**, and navigate to the **Characteristics and Quantities** tab to copy rows in the characteristic and contract quantity collections.

Description of Page

The left portion of the **Characteristics** collection initially displays existing characteristics linked to the original start option. In the **Merge Type**, you will see the word **Original**, for any of these records. The **Characteristic Type** and **Characteristic Value** for each characteristic are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

The left portion of the **Contract Quantity** collection initially displays existing contract quantities linked to the original start option. In the **Merge Type**, you will see the word **Original**, for any of these records. The **Contract Quantity Type** and **Contract Quantity** for each contract quantity are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

Refer to [Start Options Merge - Main](#) for more information about how to perform common maintenance tasks for the grids displayed on this tab page.

Start Options Merge - Contract Option

Open **Admin Menu, SA Type Start Options Merge**, and navigate to the **Contract Option** tab to copy rows in the contract options collection.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Description of Page

The left portion of the **Contract Option** collection initially displays existing contract option information linked to the original start option. In the **Merge Type**, you will see the word **Original**, for any of these records. The **Contract Option Type**, **Contract Option ID** and **Description** of the contract option for each contract option row are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

Refer to [Start Options Merge - Main](#) for more information about how to perform common maintenance tasks for the grids displayed on this tab page.

Start Options Merge - Interval Info

Open **Admin Menu, SA Type Start Options Merge**, and navigate to the **Interval Info** tab to copy rows in the interval profile, and TOU map and TOU contract quantity collections.

This tab may not appear. This tab is suppressed if the interval billing **Complex Billing** module is [turned off](#).

Description of Page

The left portion of the **Interval Profiles** collection initially displays existing interval profile information linked to the original start option. In the **Merge Type**, you will see the word *Original*, for any of these records. The **Interval Profile Relationship Type**, **Interval Profile Type** and **Interval Profile ID** for each interval profile row are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

The left portion of the **TOU Maps** collection initially displays existing TOU map information linked to the original start option. In the **Merge Type**, you will see the word *Original*, for any of these records. The **TOU Map Relationship Type**, **TOU Map Type** and **TOU Map ID** for each TOU map row are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

The left portion of the **TOU Contract Values** collection initially displays existing TOU contract values linked to the original start option. In the **Merge Type**, you will see the word *Original*, for any of these records. The **Bill Factor** and **TOU Group** for each TOU contract value are displayed. In the right portion of the collection, the existing records in the merge from start option are displayed initially.

Refer to [Start Options Merge - Main](#) for more information about how to perform common maintenance tasks for the grids displayed on this tab page.

Start Options Merge - Terms and Conditions

Open **Admin Menu**, **SA Type Start Options Merge**, and navigate to the **Terms and Conditions** tab to copy terms and conditions (T&Cs).

Description of Page

The left side of the **Terms and Conditions** grid initially displays the T&Cs linked to the original start option. On the right side, the T&Cs linked to the merge from start option are displayed initially.

Refer to [Start Options Merge - Main](#) for a description of how to perform common maintenance tasks for the grids displayed on this tab page.

Background Processes Addendum

This chapter is an addendum to the general [Defining Background Processes](#) chapter. This addendum describes the background processes that are provided with Oracle Utilities Customer Care and Billing.

Contents

- [The System Background Processes](#)
- [Batch Process Dependencies](#)
- [How To Set Up A New Extract Processes](#)
- [The Big Picture of Sample & Submit](#)

The System Background Processes

The topics in this section describe functionality that is common to system background processes.

Contents

- [Process What's Ready Processes](#)
- [Monitor Processes](#)
- [Extract Processes](#)
- [Adhoc Processes](#)
- [To Do Entry Processes](#)
- [Object Validation Processes](#)
- [Referential Integrity Validation Processes](#)
- [Conversion Processes](#)
- [Conversion Processes Executed In The Staging Database](#)
- [Purge Processes](#)
- [ConfigLab Processes](#)
- [Archive and Purge Processes](#)
- [Column Descriptions](#)

Process What's Ready Processes

Some background processes create and update records that are “ready for processing”. The definition of “ready” differs for every process. For example,

- The bill cycle process produces bills for all accounts belonging to open bill cycles.
- The account debt monitor process analyzes the debt associated with all accounts whose review date is on or before the business date.
- The process that activates pending stop and pending start SA's attempts to activate all service agreements that aren't already activated.

Processes of this type tend to use a business date in their determination of what's ready. For example, the bill cycle process creates bills for all bill cycles whose bill window is open (i.e., where the business date is between the bill cycle's start and end date). If the requester of the process does not supply a specific business date, the system assumes that the current system date should be used. If you need to use a date other than the current date, simply supply the desired date when you request the batch process.

The following table lists every background process that processes all data that is “ready”.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
ACTVTAPY	CIPPAAPB	<p>This process marks each auto pay download staging record with the batch control associated with its auto-pay source's route type. It also stamps the respective batch control's current run number on each record.</p> <p>Note: The APAYACH/C1-APACH background processes use the information on this staging table to create the flat file that is used to interface information to the ACH. The BALAPY background process uses the information on this staging table to create automatic payment tender controls.</p> <p>Refer to Activating Automatic Payments for more information.</p>	Yes	MAX-ERRORS	Yes	200/15
ADM	CIPLDMB	<p>The account debt monitor analyzes all accounts whose C&C review date is on or before the supplied business date. Refer to The C&C Monitors for more information.</p> <p>The input parameter controls how the trigger date is set on collection events that are created by this process. Refer to Calendar vs Work Days for more information about your date arithmetic options.</p>	Yes	ADD-WORK-DAYS (Y or N) MAX-ERRORS	Yes	100/15
ADM2	CIPLDM2B	<p>The account debt monitor analyzes all accounts who have not been analyzed in the last X days (where X is the Days Between Review defined on the account's customer class). Refer to The C&C Monitors for more information.</p> <p>The input parameter controls how the trigger date is set on collection events that are created by this process. Refer to Calendar vs Work Days for more information about your date arithmetic options.</p>	Yes	ADD-WORK-DAYS (Y or N) MAX-ERRORS	Yes	100/15
ANLYZSAR	CIPCCSSB	<p>The analyze SA relationship process looks at newly activated SA relationships and creates the sub service agreement(s) for them if we provide billing service for the service provider (both rate ready and bill ready). In addition, this process also:</p> <ul style="list-style-type: none"> - Activates pending start sub SAs when their 	Yes	MAX-ERRORS	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>master SA is active</p> <ul style="list-style-type: none"> - Stops pending stop sub SAs when their master SA is stopped <p>Refer to Automatic Creation of Sub SAs for more information.</p>				
APAYCRET	CIPPACRB	<p>This process creates automatic payments for bills whose automatic payment creation has been deferred until the extract date. This extract date is stamped on the bill and is used by this background process to select all bills whose automatic payment extract date is on or before the supplied business date. It calls the automatic payment creation algorithm plugged in on the installation record to create the automatic payments. Note that the algorithm supplied does not distribute and freeze the automatic payments that are created. This is handled by the complementary background process APAYDSFR.</p> <p>Refer to Installation Options - Billing and Automatic Payments for more information.</p>	Yes	MAX-ERRORS	Yes	300/15
APAYDSFR	CIPPADFB	<p>This process distributes and freezes automatic payments whose distribution date (indicated on the download staging record) is on or before the supplied business date. Payments that have been distributed (e.g., manually) are frozen if the above criterion is satisfied.</p> <p>This job complements the APAYCRET background process and the PPAPAY background process when the Autopay Creation Option on the installation record is set to Create on Extract Date.</p> <p>Refer to Installation Options - Billing and Automatic Payments for more information.</p>	Yes	MAX-ERRORS	Yes	300/15
ASSGNSBN	CIPBASBB	<p>This process allocates completed bills a sequential bill number. You need only schedule this job if your organization assigns sequential bill numbers. Please refer to Sequential Bill Numbers for important</p>	No	MAX-ERRORS	Yes	1/ not applicable

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		information about this job and why it may not be necessary if you single-thread the BILLING background process.				
BALAPY	CIPPBAPB	This process creates a new tender control (with an associated deposit control) for each batch control and run number encountered for extracted automatic payments that are not already linked to a tender control. Afterwards, this process balances the open tender and deposit control records. Note: Automatic payment staging records are activated by the ACTVTAPY process and extracted by either APAYACH or C1-APACH. Refer to Creating Automatic Payment Tender Controls for more information.	No	MAX-ERRORS	Yes	200/15
BCASSIGN	CIPFBCAB	This process assigns the Pending balance control group to new FT's (i.e., those without a balance control group). Refer to The Big Picture of Balance Control for more information.	Yes	MAX-ERRORS	No	200/15
BCGNEW	CIPFBCGB	This process creates a Pending balance control group if one doesn't already exist. Refer to The Big Picture of Balance Control for more information.	No	MAX-ERRORS	No	N/A (only 1 record is inserted)
BCGSNAP	CIPFBCSB	The balance control snapshot and verification process has two functions: 1. It summarizes the number and value of the financial transactions on the current Pending balance control group record. 2. It verifies the financial integrity of your system. The value of the VERIFY-ONLY-SW parameter controls which of these functions is performed: - If VERIFY-ONLY-SW = "N", the system summarizes the new financial transactions under the current Pending balance control and verifies that the balances summarized on <u>every</u> historical balance control group are	No	VERIFY-ONLY-SW MAX-ERRORS	No	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>consistent with the financial transactions associated with this balance control group (i.e., it checks the financial integrity of the system).</p> <p>- If VERIFY-ONLY-SW = "G", the system only summarizes the new financial transactions under the current <i>Pending</i> balance control (i.e., the verification step is not performed).</p> <p>- If VERIFY-ONLY-SW = "Y", the system verifies that the balances summarized on <u>every</u> historical balance control group are consistent with the financial transactions associated with this balance control group (i.e., it checks the financial integrity of the system).</p> <p>Note: You may want to use the VERIFY-ONLY-SW parameter to improve system performance. For example, you can generate the balance control summary nightly (run the process with the switch set to "G") and validate the balance control summaries weekly (run the process with the switch set to "Y").</p> <p>Refer to The Big Picture of Balance Control for more information.</p>				
BCU1	CIPCBC1B	<p>The first phase of the billable charge upload staging process validates and defaults information on to billable charge upload staging records.</p> <p>Refer to Billable Charge Upload Background Processes for more information.</p>	No	MAX-ERRORS	No	N/A
BCU2	CIPCBC2B	<p>The second phase of the billable charge upload staging process creates billable charges for the new billable charge upload staging records.</p> <p>Refer to Billable Charge Upload Background Processes for more information.</p>	Yes	MAX-ERRORS	No	200/15
BILLING	CIPBBILB	<p>The bill cycle process creates bills for accounts with an "open" bill cycle.</p> <p>Refer to Batch Billing for more information.</p>	Yes	MAX-ERRORS	Yes	100/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
BUDMON	CIPGMBGB	The budget monitor analyzes all customers with a budget plan and highlights those where the current budget amount is out-of-sync with the recommended budget amount. Refer to Budget Billing for more information.	Yes	MAX-ERRORS	Yes	200/15
BUDTRUP	CIPGTUPB	The budget true up process periodically trues up customers on a budget plan. Refer to Budget Billing for more information.	Yes	MAX-ERRORS	Yes	200/15
C1-ADMOV	CIPLOVMB	The overdue monitor uses your overdue rules to collect overdue debt. Refer to How Does The Overdue Monitor Work? for more information.	Yes	MAX-ERRORS	Yes	2000/15
C1-CSTRS	CIPQTRCB	The case scheduled transition process transitions cases to a nominated next status or transition condition at a scheduled time. The process selects all open cases whose current status is linked to the process' batch control code and are allowed to transition from their current status to the chosen next status or condition (i.e. where a corresponding transition rule exists for the case type/status combination) based on the input algorithm parameters.	Yes	NEXT-STATUS-CD (Next Status Code) NEXT-TR-COND-FLG (Next Transition Condition) MAX-ERRORS	Yes	200/15
C1-ODET	CIPLOETB	The overdue / cut event manager activates all overdue and cut events whose trigger date is on or before the supplied business date. Refer to How and When Events Are Activated for more information. This process also has the responsibility of recursively activating later events that are dependent on the completion of earlier events. For overdue or cut events that are in the <i>Wait</i> state, this process runs the associated waiting algorithm for the event type to determine if the object the event is waiting for is complete (and then triggering the dependent events when it completes). Populate an Overdue Process Template in	Yes	OD-PROC-TMP-CD (Optional) MAX-ERRORS	Yes	2000/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		the input parameter to limit the processing to overdue processes for this template.				
C1-PEPL1	CIPPEL1B	<p>This is the first of three background processes that load the contents of the payment event upload staging records into the various payment tables.</p> <p>It first creates new deposit and tender control records, and then updates the payment event upload staging records with the corresponding Tender Control ID.</p> <p>Next, it processes each <i>incomplete</i> record as follows: It updates the record's Tender Account ID with the account ID returned by the <i>Determine Tender Account</i> algorithm defined on the distribution rule. If the Pay Event Process ID field is not populated, it is set equal to the tender account ID. If no error was encountered, it transitions the record from to <i>Pending</i>.</p> <p>Refer to Interfacing Payments Using Distribution Rules for more information.</p>	Yes	MAX-ERRORS	Yes	300/15
C1-PEPL2	CIPPEL2B	<p>This is the second of three background processes that load the contents of the payment event upload staging records into the various payment tables.</p> <p>The responsibility of this process is to create payment events, payment tenders and payments and transition the corresponding staging records from <i>Pending</i> to <i>Complete</i>.</p> <p>Refer to Interfacing Payments Using Distribution Rules for more information.</p>	Yes	MAX-ERRORS	Yes	300/15
C1-PEPL3	CIPPEL3B	<p>This is the last of three background processes that load the contents of the payment event upload staging records into the various payment tables.</p> <p>The responsibility of this process is to update the status of the related deposit and tender controls from <i>open</i> to <i>balanced</i>.</p> <p>Refer to Interfacing Payments Using Distribution Rules for more information.</p>	Yes	MAX-ERRORS	No	300/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
C1-WFSUB	CIPWWETB	<p>This process does two things:</p> <ul style="list-style-type: none"> It sets the trigger date of workflow events for the input workflow process template that are dependent on the completion of earlier workflow events. It activates all workflow events for the input workflow process template whose trigger date is on or before the supplied business date. <p>This background process is the same code used for WFET. It is used for the batch scheduling functionality.</p> <p>Refer to Workflow Event Dependencies for more information.</p>	Yes	WF-PROC-TMPL-CD (Workflow Process Template) MAX-ERRORS	No	200/15
CAREPRO G	CIPCCRCB	<p>This batch process is responsible for creating customer contacts (letters) for SA characteristics that are about to expire.</p> <p>It finds SAs with the indicated Characteristic Type and Value that will expire within Threshold Days and creates the indicated type of customer contact.</p> <p>Note. The Threshold Days are calendar days.</p> <p>(In California this can be used to create letters for CARE, California Alternate Rates for Energy, SAs that are about to expire.)</p>	No	CHAR-TYPE-CD (Characteristic Type Code) CHAR-VAL (Characteristic Type Value) THRES-DAYS (Threshold Days) CC-CLASS (Customer Contact Class) CC-TYPE (Customer Contact Type) MAX-ERRORS	Yes	200/15
CASETRAN	CIPQCSTB	<p>This batch process is responsible for calling the algorithm that determines if a case should be transitioned to a new state. Refer to Automatic Transition Rules for the details.</p> <p>If Restrict To Case Type Code is specified, only cases of this type will be analyzed to determine if they should be transitioned to a new state.</p>	Yes	CASE-TYPE-CD (Restrict To Case Type Code) CASE-STATUS-CD (Restrict To Case Status)	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		If Restrict To Case Status Code is specified, only cases in this status will be analyzed to determine if they should be transitioned to a new state. Note, if this parameter is specified, a Restrict To Case Type Code must also be defined.		Code) MAX-ERRORS		
CET	CIPLCETB	The collection event trigger activates all collection events whose trigger date is on or before the supplied business date. Refer to How Are Collection Events Completed for more information. The input parameter controls how the trigger date is set on severance events that are created by collection events. Refer to Calendar vs Work Days for more information about your date arithmetic options.	Yes	ADD-WORK-DAYS (Y or N) MAX-ERRORS	Yes	100/60
CLOSEQTE	CIPCUQEB	The close quotes process closes all quotes whose expiration date is on / before the business date. The PROP-SA-ACTION parameter controls whether the proposal SAs linked to the quote's quote details should be marked as declined or cancelled. If you indicate the proposal SAs should be declined, you must also use PROP-DCL-RSN-CD to define the declination reason code to be updated on the SA's.	Yes	PROP-SA-ACTION (DECL or CANCEL) PROP-DCL-RSN-CD MAX-ERRORS	Yes	300/15
CPM	CIPLCPMB	The collection process monitor removes service agreements from collection processes when they have sufficient credits. It will also cancel a collection process when all of its service agreements have been removed. Refer to The C&C Monitors for more information.	Yes	MAX-ERRORS	Yes	100/60
DEPINTRF	CIPDINTB	The deposit interest refund process calculates the deposit amount for service agreements whose SA type has a special role of "cash deposit". Refer to Deposit Interest for more information.	Yes	MAX-ERRORS	Yes	200/15
DEPRFND	CIPDRFNB	The deposit refund process refunds deposits	Yes	MAX-ERRORS	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		to a customer when the customer satisfies the refund criteria. Refer to Deposit Refunds for more information.				
DEPRVW	CIPDRVWB	The deposit review process highlights accounts that require an additional deposit. Provide an input Deposit Class to optionally restrict the review to accounts that have SAs belonging to that class. Refer to Deposit Review for more information.	Yes	DEP-CL-CD (Optional) MAX-ERRORS	Yes	200/15
FACOMPL	CIPOFOCB	The field activity completion process completes field activities and field orders using the records in the various field activity staging tables. Refer to Uploading Field Order Completion Information for more details.	Yes	MAX-ERRORS	No	100/10
FACT	CIPOFACB	The field activity remark activation process executes the algorithm associated with field activities that have <i>pending</i> field activity remarks that reference an activation algorithm. Refer to Field Activity Remark Exceptions for more information.	Yes	MAX-ERRORS	No	200/15
FANRMRO	CIPOSSRB	The complete field activity using a recent read process looks for pending start / stop field activities (using the input FA-TYPE-CD) for which a meter read was recently taken (within NO-OF-DAYS). It prorates the read (Usage only) into an end read and a start read for the start and stop SAs. The new reads are created with meter read type <i>system prorated</i> , and are linked to the SA/SP. A meter read remark is also created for the reading (using the input READER-REM-CD). If this process finds a peak demand type reading or a consumption type reading, it posts a To Do entry using the TD-TYPE-CD and ROLE-ID parameters. Refer to How To Start Service Using A Scheduled Meter Read for more information.	Yes	TD-TYPE-CD (To Do Type Code) ROLE-ID (To Do Role) READER-REM-CD (MR Remark Code) FA-TYPE-CD (Field Activity Type) NO-OF-DAYS (Number of Days) MAX-ERRORS	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>The base package is set up with a To Do Type <i>D/C REG</i>, which should be used for the TD-TYPE-CD parameter, unless you've set up your own.</p> <p>This process will only be useful if you've created field activities of a special type. A base package field activity creation algorithm support this; refer to the algorithm SASP FW CRE for more information.</p>				
FDS	CIPOSTDB	<p>The create field order download staging process creates a download staging record for every field order that is ready for download.</p> <p>This process populates the field order download staging table with the unique ID of all field orders to be routed to field work personnel. This process marks each staging record with the batch control associated with the field order's dispatch group. It also stamps the respective batch control's current run number on each record.</p> <p>Note: The FODL background process uses the information on this staging table to create the flat file that is used to interface information to your field order print software.</p> <p>Refer to Downloading Field Orders for more information.</p>	No	MAX-ERRORS	Yes	200/15
FOD	CIPOADPB	<p>This batch process automatically dispatches eligible field activities X days in advance of their scheduled date (where X is defined by the parameter <i>DISPATCH-DAYS</i>). By dispatch, we mean that this process creates field orders and links the field activities to the field order.</p> <p>Eligible field activities are:</p> <ul style="list-style-type: none"> - in the <i>pending</i> state - have a field activity type that is <i>dispatchable</i> - have a scheduled date / time that is within the specified DISPATCH-DAYS 	No	DISPATCH-DAYS (This is an optional parameter. If it is not entered the system will select all field activities with a scheduled date up to and including the business date. If it is entered, the number of	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<ul style="list-style-type: none"> - if the PRIORITY parameter has been specified, the field activity's field activity type must have a priority that matches the value of the parameter <p>The dispatched field activities are grouped by premise, dispatch group (or alternate dispatch group) and date. Refer to Dispatch Field Activities Automatically for more information on dispatching field activities.</p>		<p>DISPATCH-DAYS will be added to the business date and this process will select records up to and including this new date.)</p> <p>PRIORITY (This is an optional parameter. If specified, it must equal one of the field activity type priority values – in the base package, these are 10, 20, 30, 40, 50, 60, 70, 80, 90)</p> <p>MAX-ERRORS</p>		
GLASSIGN	CIPFGLAB	The GL account number assignment process assigns GL account numbers to the GL details associated with financial transactions. Refer to The GL Interface for more information.	Yes	MAX-ERRORS	Yes	200/15
GLS	CIPFGLEB	<p>The create general ledger download staging process creates a download staging record for every financial transaction that is ready for download.</p> <p>This process populates the FT / Batch Process table with the unique ID of all financial transactions to be interfaced to the general ledger. This process marks each staging record with the GL interface's batch process ID (defined on the installation record). It also stamps the respective batch control's current run number on each record.</p>	Yes	MAX-ERRORS	No	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>Note: The GLDL background process uses the information on this staging table to create the consolidated journal entries that are interfaced to your general ledger.</p> <p>Refer to The GL Interface for more information.</p>				
IB-SPDB	CIPISPDB	<p>This process derives interval data for accounts in the system. Only accounts that have at least one interval SA with derivable profiles linked to it are processed. A 'derivable' profile is an SA Owned profile where this SA is the owner AND the profile type indicates an "Interval Data Creation" derivation algorithm. Interval data for SAs linked to the Account are derived in Process Priority order as defined on their SA Type. For each SA, the Interval Data Creation algorithms are executed in creation priority order.</p> <p>The standard batch parameter business date will be used by the system to determine until what date to generate data for.</p> <p>Use the Force Derive Switch parameter to indicate that you are forcing derivation. Use the Force Derive Start Date to indicate the starting point for the forced derivation.</p> <p>Refer to Creation of Profile Data through Data Derivation for more information.</p>	Yes	<p>FORCE-DERIVE-SW (Indicates whether this is a forced derivation - optional.)</p> <p>FRCE-DRV-START-DT (Start date YYYY-MM-DD for forcing derivation – optional)</p> <p>MAX-ERRORS</p>	No	1/15
IB-STDB	CIPISTDB	<p>This process derives TOU map data for accounts in the system. Only accounts that have at least one interval SA with derivable TOU maps linked to it are processed. A 'derivable' map is an SA Owned map where this SA is the owner AND the map type indicates a "TOU Data Creation" algorithm. TOU map data for SAs linked to the Account are derived in Process Priority order as defined on their SA Type. For each SA, the TOU Map Creation algorithms are executed in creation priority order.</p> <p>The standard batch parameter business date</p>	Yes	<p>FORCE-DERIVE-SW (Indicates whether this is a forced derivation - optional.)</p> <p>FRCE-DRV-START-DT (Start date YYYY-MM-DD for forcing derivation –</p>	No	1/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>will be used by the system to determine until what date to generate data for.</p> <p>Use the Force Derive Switch parameter to indicate that you are forcing derivation. Use the Force Derive Start Date to indicate the starting point for the forced derivation.</p> <p>Refer to Automatic Creation of TOU Map Data for more information.</p>		<p>optional)</p> <p>MAX-ERRORS</p>		
IPDSDVb	CIPIPDVB	<p>This process is used to validate interval profile data. It processes interval profiles that were created up to the cutoff date/time and executes their validation algorithms, if any, defined on their profile type. The algorithms are executed one after the other in their predefined sequence order.</p> <p>Refer to Validation of Profile Data for more information.</p>	Yes	MAX-ERRORS	No	50/15
IPDSIDB	CIPIPIDB	<p>The Determine Profile For Profile Datasets process attempts to link interval profile data sets to an appropriate profile. It tries to find an interval profile with the same external ID as the one defined on the dataset. Use the Start External ID and End External ID parameters if you only want to process records in that range of Ids.</p> <p>Only profile data sets in pending status that are not already associated with a profile are processed.</p> <p>Refer to Upload Interval Data for more information.</p>	Yes	<p>START-EXT-ID (Start External ID - optional)</p> <p>END-EXT-ID (End External ID- optional)</p> <p>MAX-ERRORS</p>	No	50/15
IREGDVB	CIPIRDVB	<p>The Interval Register Data Validation process is used to validate interval register data. It processes interval registers that were created up to the cutoff date/time and executes their validation algorithms, if any, defined on their interval register type. The algorithms are executed one after the other in their predefined sequence order.</p> <p>Refer to Validation of Register Data for more information.</p>	Yes	MAX-ERRORS	No	50/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
IREGIDB	CIPIRIDB	<p>The Determine Register For Register Data Sets process attempts to link interval register data sets to an appropriate interval register. It tries to find an interval register with the same external ID as the one defined on the dataset. Use the Start External ID and End External ID parameters if you only want to process records in that range of Ids.</p> <p>Only interval register data sets in pending status that are not already associated with a register are processed.</p> <p>Refer to Upload Interval Register Data for more information</p>	Yes	START-EXT-ID (Start External ID - optional) END-EXT-ID (End External ID- optional) MAX-ERRORS	No	50/15
LATEPYMT	CIPBLPCB	<p>The late payment generator creates late payment charges when an account doesn't pay a bill by the end of the grace period.</p> <p>Refer to How Late Payment Charges Get Calculated for more information.</p>	Yes	MAX-ERRORS	Yes	100/15
MDS	CIPMMDSB	<p>The create meter read download staging process creates a download staging record for every meter on every route that is ready for download. This process uses the route information on the service cycle schedule. The MSR process must run before this process runs.</p> <p>This process marks each staging record with the batch control associated with the meter's route. It also stamps the respective batch control's current run number on each record.</p> <p>Note: the MDL background process uses the information on the staging table to create the flat file that is used to interface information to your meter read software.</p> <p>Refer to Downloading Meter Reads for more information.</p>	No	MAX-ERRORS	No	200/15
MRRA	CIPMRRUB	<p>The meter read remark activation process executes the algorithm associated with recently added meter reads that have meter read remark codes that reference an activation algorithm.</p>	No	MAX-ERRORS	No	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		Refer to Meter Read Remark Exceptions for more information.				
MSR	CIPMMSRB	The prepare meter read route for download process looks for all routes that are scheduled to be downloaded and populates them on the service cycle schedule. The MDS process must run after this process in order to populate the meter read download staging table. Refer to Downloading Meter Reads for more information.	No	MAX-ERRORS	No	200/15
MUP1	CIPMUP1B	The first phase of the meter read upload staging process defaults meter configuration ID's for the badge number on newly added meter read upload staging records. If unique geographic type/value information is available, the meter configuration ID is populated even if the badge number is not unique. Refer to Uploading Meter Reads for more information.	Yes	MAX-ERRORS	No	200/15
MUP2	CIPMUP2B	The second phase of the meter read upload staging process creates meter and register reads for the new meter read upload staging records. Refer to Uploading Meter Reads for more information.	Yes	MAX-ERRORS	No	200/15
NBBAPAY	CIPGACRB	The NBB scheduled payment autopay create process creates autopay records for any scheduled non-billed budget payments where the account is set up for autopay and the non-billed budget is not excluded from autopay. For more information about non-billed budget auto payments, refer to Automatic Payment and Non-billed Budgets .	Yes	MAX-ERRORS	Yes	300/ not applicable
NBBPS	CIPGNPSB	The non-billed budget scheduled payment process performs processing for scheduled payment records with a payment date on or before the process business date.	Yes	MAX-ERRORS	Yes	300/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>If the scheduled payment is for a monitored non-billed budget the process runs the NBB process scheduled payment algorithm.</p> <p>If the scheduled payment processing is successful or was not required (i.e. for unmonitored non-billed budgets), the process deletes the current scheduled payment.</p> <p>For more information, refer to Scheduled and Actual Payments on the Non-billed Budget.</p>				
PAYSPR	CIPPSPRB	<p>The pay service provider process creates the adjustment used to increase the amount owed to service providers (when you provide billing services for the service provider). This process uses all FT / Process associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process uses all FT / Process records marked with the current run number.</p> <p>Refer to The Technical Implementation Of Paying Service Providers for more information.</p> <p>Note: records are written to the FT / Process table when an FT is frozen that is associated with a sub service agreement.</p>	Yes	MAX-ERRORS	Yes	200/15
PPAPAY	CIPCPPAB	<p>This process creates automatic payments on the scheduled payment date by calling the automatic payment creation algorithm plugged in on the installation record. Note, automatic payments are only created if:</p> <ol style="list-style-type: none"> 1) the account has indicated that they pay automatically 2) the payment method on the pay plan indicates automatic payment should be performed. <p>Note that the automatic payment creation algorithm supplied does not distribute and freeze the automatic payments that are created if the Autopay Creation Option on the installation record is set to Create on Extract Date. The background process</p>	Yes	MAX-ERRORS	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		APAYDSFR handles this. Refer to The Big Picture Of Pay Plans and Automatic Payments for more information.				
PPM	CIPCMPPB	This batch process is responsible for monitoring all active payment plans. It looks for payments made by the pay plan's payor and for SAs in the same debt class as the pay plan's debt class. It uses these payments to logically offset the pay plan's scheduled payments. This batch process determines if a pay plan has been kept, broken or remains active. An ADM trigger is stored for those accounts whose pay plans have been broken. Refer to The Pay Plan Monitor for more information.	Yes	MAX-ERRORS	Yes	200/15
PSASPM	CIPCSSMB	The create field activities for pending start and pending stops process creates field activities for pending start and stop service agreements shortly before the start/stop date if field activities have not already been created. Refer to When Field Activities Are Created To Start Service and When Stop Field Activities Are Created for more information.	Yes	MAX-ERRORS	Yes	100/60
PUPL	CIPPUPLB	The upload payments process creates payment events, payments, and tenders using the records in the various payment staging tables. Refer to Interfacing Payments From External Sources for more information.	Yes	MAX-ERRORS	No	100/15
PY-RPE	CIPPRPEB	The resolve payments in error process attempts to resolve the following payment errors automatically: <ul style="list-style-type: none">A valid account was found but no active SA exists. Refer to Resolving Exceptions Automatically for more information.	Yes	MAX-ERRORS	No	200/15
REACH	CIPBCHRB	This batch process accumulates the total	No	SA-TYPE-CD.	No	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>amount paid towards charity service agreements in the past year (for each account). If the resultant amount is greater than zero, a temporary bill message will be added to the account.</p> <p>SA-TYPE-CD defines the SA type of your charitable contribution service agreements. Note, this is the SA Type code without the Division. All service agreements with this SA Type, regardless of Division, will be processed.</p> <p>START-DT is the start date of the financial year in which payments should be accumulated.</p> <p>END-DT is the end date of the financial year in which payments should be accumulated.</p> <p>BILL-MSG-CD is the code of the bill message that will be added to the customer's bill. Note, your bill message code should have two parameters: the tax year and the total amount of contributions. For example, <i>Thank you for your charitable contributions in %1 for %2, please keep this bill for tax purposes</i>. The tax year is derived from the year in the START-DT parameter.</p> <p>ACCT-ID should be zero if this processing should happen for all accounts in the system. If this parameter is non-zero, this process will be limited to the supplied ACCT-ID</p> <p>Use ADJ-TYPE-CD1 and CD2 to indicate adjustments whose FTs should be ignored when calculating the contribution amount. This allows you to make adjustments to the charitable contribution SA that are not included in the calculation.</p> <p>(Note: in California, this program is referred to as REACH – Relief for Energy Assistance through Community Help.)</p>		<p>This is the SA type of the charitable contribution service agreements.</p> <p>START-DT. It should be entered in the format YYYY-MM-DD.</p> <p>END-DT. This should be entered in the format YYYY-MM-DD.</p> <p>BILL-MSG-CD.</p> <p>ACCT-ID</p> <p>MAX-ERRORS</p> <p>ADJ-TYPE-CD1</p> <p>ADJ-TYPE-CD2</p>		
REDSAAMT	CIPFFTRB	This process looks for financial transactions linked to each SA that are older than X days (where X is defined on the installation record)	Yes	MAX-ERRORS	Yes	100/10

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		that sum to zero. If it finds such FTs, it marks them as "redundant". Redundant FTs do not have to be accessed by the various SQL statements that accumulate an account or SA's balance.				
SAACT	CIPCSATB	The service agreement activation process updates pending start and pending stop service agreements. Refer to The System Activates Most Service Agreements Behind The Scenes for more information.	Yes	MAX-ERRORS	Yes	200/15
SAEXPIRE	CIPCOSVB	The stop expired service agreement process initiates the stop for all active service agreements where the expiration date is on or before the process date. For more information, refer to Expiring Service Agreements .	Yes	MAX-ERRORS	Yes	300/ not applicable
SARENEW	CIPCSARB	The service agreement renewal process renews all active SAs that are due for renewal (i.e. where the renewal date is populated and is less than or equal to the process date). For more information, refer to Renewing Service Agreements .	Yes	MAX-ERRORS	Yes	300/15
SASP	CIPCSAPB	The find read for SA/SP process updates SA/SP records of <u>active</u> service agreements with the appropriate start or stop read. This read is derived from the completed field activity linked to the SA/SP. Only billable reads linked to the field activity are used for the start or stop read. Refer to SAACT for the process that handles SAs that are pending start or pending stop.	Yes	MAX-ERRORS	Yes	200/15
SEC	CIPLSECB	The severance event completion process completes field activity oriented severance events when their field activity is completed. Refer to The Big Picture Of Severance Events for more information.	No	MAX-ERRORS	No	100/15
SED	CIPLSEDB	The severance event set dependency date sets the trigger date of later severance events that are dependent on the completion of	Yes	ADD-WORK-DAYS	Yes	100/60

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>earlier severance events. Refer to The Big Picture Of Severance Events for more information.</p> <p>The input parameter controls how the trigger date is set. Refer to Calendar vs Work Days for more information about your date arithmetic options.</p>		MAX-ERRORS		
SET	CIPLSETB	<p>The severance event trigger activates all severance events whose trigger date is on or before the supplied business date.</p> <p>Refer to The Big Picture Of Severance Events for more information.</p>	Yes	MAX-ERRORS	Yes	100/60
STMPRD	CIPBSTCB	<p>This process creates statements for statement construct records with a pending statement cycle whose processing date has been reached.</p> <p>Refer to Create Statements for more information.</p>	No	MAX-ERRORS	Yes	200/15
TREND	CIPMTRNB	<p>The meter read trend process updates meter read trends using consumption from newly added reads. Only register readings that have not failed high/low and whose read type is <i>Customer Read</i>, <i>Regular</i> or <i>Verified</i> are considered by this process.</p> <p>In addition, a To Do entry is created for each meter read where the SP is not linked to a SA and where daily consumption exceeds a trickle consumption threshold, computed as a percentage of the billable average daily consumption derived from the trend.</p> <p>The base package is set up with a To Do Type <i>NCFSSPS</i>, which should be the first parameter to this process, unless you've set up your own.</p> <p>Refer to Trends Are Updated Behind The Scenes for more information.</p>	Yes	TD-TYPE-CD (To Do Type) ROLE-ID (To Do Role) THRESHOLD (Threshold Trickle Percent) MAX-ERRORS	No	200/15
UARENEW	CIPKURNB	<p>This is the umbrella agreement renewal process. It selects all the umbrella agreements in <i>Complete</i> status with a renewal date on or before the given business</p>	Yes	MAX-ERRORS	Yes	

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>date.</p> <p>For each umbrella agreement found, it calls the renewal algorithm on the umbrella agreement's UA type.</p> <p>Refer to Umbrella Agreement Renewal for more information</p>				
WAITCOM	CIPWTMCB	<p>This workflow waiting process <i>Completes</i> a workflow event that has been in the <i>Waiting</i> state for longer than X days (X is defined in a parameter supplied to the background process).</p> <p>Refer to Waiting Events And Their Waiting Processes for more information.</p>	Yes	WAIT-DAYS MAX-ERRORS	Yes	200/15
WAITFA	CIPWTFAB	<p>This workflow waiting process monitors the state of a field activity created as a result of a workflow event. When the field activity is <i>Complete</i>, the associated workflow event is moved from the <i>Waiting</i> to the <i>Complete</i> state. This process will <i>Fail</i> the workflow event if the event has been in the <i>Waiting</i> state for longer the X days (X is defined in a parameter supplied to the background process).</p> <p>Refer to Waiting Events And Their Waiting Processes for more information.</p>	Yes	TIMEOUT-DAYS MAX-ERRORS	Yes	200/15
WAITMAN	CIPWTMNB	<p>This workflow waiting process <i>Fails</i> a workflow event that has been in the <i>Waiting</i> state for longer than X days (X is defined in a parameter supplied to the background process).</p> <p>Refer to Waiting Events And Their Waiting Processes for more information.</p>	Yes	TIMEOUT-DAYS MAX-ERRORS	Yes	200/15
WAITNT	CIPWTNTB	<p>This workflow waiting process monitors the Notification Upload Staging Response (stored as a context entry) of a notification download staging record that was created as a result of a workflow event. If a Response of Accept appears, the associated event is marked as <i>Complete</i>. If a Response of Reject appears, the associated event is marked as <i>Failed</i>. This process will <i>Fail</i> the workflow event if the</p>	Yes	TIMEOUT-DAYS MAX-ERRORS	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generate s To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>event has been in the Waiting state for longer the X days (X is defined in a parameter supplied to the background process).</p> <p>Refer to Waiting Events And Their Waiting Processes for more information.</p>				
WET	CIPLWETB	<p>The write-off event trigger activates all write-off events whose trigger date is on or before the supplied business date.</p> <p>Refer to The Big Picture Of Write-Off Events for more information.</p>	Yes	MAX-ERRORS	Yes	200/15
WFET	CIPWWETB	<p>This process does two things:</p> <ul style="list-style-type: none"> It sets the trigger date of workflow events that are dependent on the completion of earlier workflow events. It activates all workflow events whose trigger date is on or before the supplied business date. If the input Workflow Process Template is populated, only events for workflow processes with that template are processed. <p>Refer to Workflow Event Dependencies for more information.</p>	Yes	WF-PROC-TMPL-CD (Workflow Process Template - optional) MAX-ERRORS	No	200/15
WFPRINIT	CIPWNUSB	<p>The workflow process initiation process creates a workflow process to handle a notification upload staging record.</p> <p>Refer to How Are Workflow Processes Created for more information.</p>	Yes	MAX-ERRORS	Yes	200/15
WPM	CIPLWMOB	<p>The write-off monitor analyzes all accounts with finalized, unpaid service agreements. Refer to The Write Off Monitor for more information.</p> <p>The input parameter controls how the trigger date is set on write-off events that are created by this process. Refer to Calendar vs Work Days for more information about your date arithmetic options.</p>	Yes	ADD-WORK-DAYS MAX-ERRORS	Yes	100/15

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Monitor Processes

A periodic monitor batch process is provided for any maintenance object whose business object defines a [lifecycle](#). In addition deferred monitor batch process is provided if a business object supplied in the base product required a deferred process for one of its states.

Refer to [Monitoring Batch Processes](#) for more information.

Extract Processes

Extract processes extract information that is interfaced out of the system. Processes of this type typically extract records marked with a given run number. If the requester of the process does not supply a specific run number, the system assumes that the latest run number should be extracted. If you need to re-extract an historical batch, you can – simply supply the respective run number when you request the batch process.

Business Intelligence Extracts. If you are using Oracle Utilities Business Intelligence, there will be many other extract processes. These additional processes are responsible for extracting the data sent to the data-warehouse. Please see the product's fact and dimension chapter of the business intelligence documentation for a description of each extract process.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
APAYACH	CIPPXAPB	The automatic payment ACH (automated clearing house) download extraction process creates the flat file that is interfaced to the ACH. This process downloads all auto pay download staging records associated with its batch control ID that are marked with a supplied run number. If a run number is not supplied, the process extracts all automatic payment download records marked with the current run number. Note: the ACTVTAPY process updates auto pay download records on their extract date so that they will be downloaded by this process. Refer to Downloading Automatic Payments for more information.	Yes	FILE-PATH = directory path into which output should be placed FILE-NAME = name of file into which output should be placed MAX-ERRORS	NA
C1-APACH	CIPPX2B	The automatic payment ACH (automated clearing house) download extraction process creates the flat file that is interfaced to the ACH. This process downloads all auto pay download staging records associated with its	No	FILE-PATH = directory path into which output should be placed	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>batch control ID that are marked with a supplied run number. If a run number is not supplied, the process extracts all automatic payment download records marked with the current run number.</p> <p>The NBR-DAYS parameter is added to the scheduled extract date to determine the draft date when the payment amount should be withdrawn from the customer's account. If specified, the value must be a non-negative numeric value.</p> <p>The BUS-OR-CAL-DAYS parameter is required when the NBR-DAYS parameter is specified, otherwise it's not allowed. This parameter is used when adding the NBR-DAYS parameter to the scheduled extract date. Valid values are:</p> <p>B - Business days C - Calendar days</p> <p>Note: the ACTVTAPY process updates auto pay download records on their extract date so that they will be downloaded by this process.</p> <p>Refer to Downloading Automatic Payments for more information.</p>		<p>FILE-NAME= name of file into which output should be placed</p> <p>NBR-DAYS= number of days until withdrawal.</p> <p>BUS-OR-CAL-DAYS= business or calendar days.</p> <p>MAX-ERRORS</p>	
APDL	CIPADAPB	<p>The A/P download process creates the flat file that is interfaced to your accounts payable software (to cut checks).</p> <p>The process that is delivered has skeletal logic and must be customized by your organization to satisfy the needs of your accounts payable software.</p> <p>In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. 	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<ul style="list-style-type: none"> Introduce logic to format the downloaded records into your specific format. If you need assistance, please contact the implementation support group. <p>This process uses all adjustment extract records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process uses all A/P request extract records marked with the current run number.</p> <p>Refer to The A/P Interface for more information.</p>			
DSGPFODL	CIPODGFB	<p>The dispatch group FO print production process invokes the field order extract algorithm defined on the dispatch group associate to the batch control. The extract algorithm determines the field order print software to use and report or field order template to use to produce the field orders from the field order download staging table associated with the batch control and run number for a range of field orders specified.</p> <p>This process is similar to the existing FO Print Production Process (FODL). The differences are:</p> <ul style="list-style-type: none"> FODL calls the extract algorithm defined in the Dispatch Group for each field order associated to the Dispatch Group, while this process only calls the distinct extract algorithm associated with the batch control. FODL produces a flat file that is interfaced to a FO print software, while this process calls the FO print software for every distinct extract algorithm found and produce the field orders (i.e. the report containing the field orders). <p>Refer to Downloading Field Orders for more information.</p>	Yes	MAX-ERRORS	NA
DWLDBILC	CIPXBCB	The billable charge download extraction	Yes	FILE-PATH=	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>process creates the flat file that contains billable charges to be interfaced to your service providers. This process extracts all FT / Process records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all FT / Process records marked with the current run number.</p> <p>The program that is delivered contains skeletal logic that should be used as the basis for your specific processing. The skeletal logic does NOT extract information to a flat file; you must introduce the logic to support your specific flat file format. In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Note: records are written to the FT / Process table when a bill is completed that contains master service agreements with service providers who provide billing service for your organization.</p> <p>Refer to Routing Billable Charges To Service Providers for more information.</p>		<p>directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	
DWLDCONS	CIPMXCSB	<p>The consumption download extraction process creates the flat file that contains consumption to be interfaced to your service providers. This process extracts all FT / Process records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all FT / Process records marked with the current run number.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>The program that is delivered contains skeletal logic that should be used as the basis for your specific processing. The skeletal logic does NOT extract information to a flat file; you must introduce the logic to support your specific flat file format.</p> <p>In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Note: records are written to the FT / Process table when a master service agreement's bill segment is frozen if the master service agreement is linked to service providers who need consumption.</p> <p>Refer to Routing Consumption To Service Providers for more information.</p>			
DWLDCOLL	CIPLXCRB	<p>The collection agency referral download extraction process creates the flat file that contains referrals to be interfaced to your collection agencies. This process extracts all collection agency referral history records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all referral history records marked with the current run number.</p> <p>The program that is delivered contains skeletal logic that should be used as the basis for your specific processing. The skeletal logic does NOT extract information to a flat file; you must introduce the logic to support your specific flat file format.</p> <p>In order to adapt the base package program to</p>	No	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Note: records are written to the referral history table when a collection agency oriented write-off events are activated. Referral history records may also be added manually by an operator.</p> <p>Refer to How Do Collection Agency Referrals Work? for more information.</p>			
FODL	CIPOFOX	<p>The field order download process creates the flat file that is interfaced to your field order print / dispatching software.</p> <p>This process uses all field order extract records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process uses all field order extract records marked with the current run number.</p> <p>The information that is extracted and placed on the flat file for each field order is controlled by each order's dispatch group's extract algorithm. Refer to Dispatch Groups Control The Information Merged Onto Field Orders for information about how a field order's flat file records are constructed.</p> <p>Refer to Downloading Field Orders for more information.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA
GLDL	CIPFXGLB	<p>The general ledger download process creates the flat file that is interfaced to your general ledger software.</p> <p>This process uses all FT / Batch Process records associated with its batch control that</p>	No	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>are marked with a supplied run number. If a run number is not supplied, the process uses all FT / Process records marked with the current run number.</p> <p>In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Refer to The GL Interface for more information.</p>		<p>of file into which output should be placed</p> <p>MAX-ERRORS</p>	
LTRPRT	CIPCLTPB	<p>The customer contact letter download process creates the flat file(s) that are interfaced to your letter print software to print letters associated with letter-oriented customer contacts.</p> <p>This process extracts all customer contact records associated with its batch control ID that are marked with a supplied run number. If a run number is not supplied, the process uses all customer contact records associated with its batch control ID that are marked with the current run number.</p> <p>Each downloaded letter's output is written to a filename that is a concatenation of the letter's Letter Template Code and the process's Thread Number. This means that this process can write to multiple files as multiple Letter Template Codes may be downloaded by this process.</p> <p>The information that is extracted and placed on the flat file for each letter is controlled by each customer contact's letter template's extract algorithm. Refer to Letter Templates Control The Information Merged Onto Letters for information about how a letter's flat file records</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FIELD-DELIM-SW=Y or N</p> <p>CNTL-REC-SW=Y or N</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>are constructed.</p> <p>The FILE-PATH parameter controls where the output files are placed.</p> <p>The format of the information on the flat file can be either tilde delimited or in a fixed position (based on the FIELD-DELIM-SW parameter). Tilde delimited output is used if you merge the information into a Word template. Fixed position output is used if you merge the information into a Doc 1 template.</p> <p>You can use the CNTL-REC-SW parameter to cause the extract to produce a control record that contains batch code, run number, number of letters to print, etc.</p> <p>Refer to Printing Letters for more information.</p>			
MDL	CIPMMDLB	<p>The meter read download extraction process creates the flat file that is interfaced to your meter reading software. This process uses all meter read download staging records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process uses all meter read extract records marked with the current run number.</p> <p>The program that is delivered contains skeletal logic that should be used as the basis for your specific processing. The skeletal logic does NOT extract information to a flat file; you must introduce the logic to support your specific flat file format. In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Refer to Downloading Meter Reads for more</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		information.			
NDSXTR	CIPWXNDB	<p>The notification download extraction process creates the flat file that is interfaced to your notification routing. This process uses all notification download staging records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process uses all records marked with the current run number.</p> <p>The program that is delivered should be used as a sample as your record formats will differ. In order to adapt the base package program to your specific needs, please following the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters <i>CM</i> (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Introduce logic to format the downloaded records into your specific format. <p>Refer to Downloading Notifications for more information.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA
POSTROUT	CIPBXBLB	<p>The bill print process creates the flat file that is interfaced to your bill print software. This process uses all bill routing extract records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all bill routing extract records marked with the current run number.</p> <p>The information that is extracted and placed on the flat file for each bill is controlled by each bill route type's extract algorithm. Refer to Bill Route Controls The Information Merged Onto Bills for information about how a bill's flat file records are constructed.</p> <p>The FILE-PATH parameter controls where the output files are placed.</p> <p>Refer to Printing Bills for more information.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
QUOTROUT	CIPCQTXB	<p>The quote-print process creates the flat file that is interfaced to your quote-print software. This process uses all quote-routing extract records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all quote routing extract records marked with the current run number.</p> <p>The information that is extracted and placed on the flat file for each letter is controlled by each quote route type's extract algorithm. Refer to Quote Route Type Controls The Information Merged Onto Quotes for information about how a quote's flat file records are constructed.</p> <p>The FILE-PATH parameter controls where the output files are placed.</p> <p>Refer to Printing Quotes for more information.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	NA
RTTYPOST	CIPBRTBB	<p>The bill route type bill print production process invokes the bill extract algorithm defined in the bill route type associated to the batch control. The extract algorithm will determine the bill print software to use and the report or bill template to use to produce all the bill routing records associated with the batch control and run number for a range of bills specified.</p> <p>This process is similar to the Bill Print Production Process (POSTROUT). The differences are:</p> <ul style="list-style-type: none"> • POSTROUT calls the extract algorithm defined in the Bill Route Type for each bill associated to the Bill Route Type; while RTTYPOST only calls the distinct extract algorithm associated with the batch control. • POSTROUT produces a flat file that is interfaced to a bill print software, while RTTYPOST calls the bill print software for every distinct extract algorithm found and produce the bills (i.e. the report containing the bills). <p>Refer to Printing Bills for more information</p>	Yes	MAX-ERRORS	NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
STMDWLD	CIPBSTXB	<p>The statement download extraction process creates the flat file that contains statement information. This file will be interfaced to your bill print software, or whatever mechanism you will use for sending statements to the appropriate persons.</p> <p>This process extracts all Statement records associated with its batch control that are marked with a supplied run number. If a run number is not supplied, the process extracts all Statement records marked with the current run number.</p> <p>The information that is extracted and placed on the flat file for each statement is controlled by each statement's construct's route type's extract algorithm. Refer to Statement Route Types Control The Information Merged Onto Statements for information about how a statement's flat file records are constructed.</p> <p>Refer to Printing Statements for more information.</p>	Yes	<p>FILE-PATH= directory path into which output should be placed</p> <p>FILE-NAME= name of file into which output should be placed</p> <p>MAX-ERRORS</p>	200/15

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Adhoc Processes

These are background processes that are run on an ad hoc basis (e.g., if you need to back out bills that were created by the billing process).

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
CPCRALOC	CIPCCCLB	<p>The capital credit allocation process creates allocation service credit events for active capital credit memberships.</p> <p>THIS BACKGROUND PROCESS IS ONLY APPLICABLE TO COMPANIES THAT ISSUE CAPITAL CREDITS TO THEIR</p>	Yes	<p>SC-EVT-TYPE-CD = service credit event type to use for new events</p> <p>SC-EVT-</p>	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>CUSTOMERS.</p> <p>Refer to Allocating Capital Credits for more information.</p> <p>The program that is delivered one possible method for calculating capital credit allocations. If the logic provided does not meet your business needs, you must adapt the base package program to your specific needs. To do this, please follow the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from overwriting your new logic. Modify the new program as per your business needs. Create a batch control record for your new background process, prefixed with CM. 		<p>STATUS-FLG = Indicate if the events should be created in pending (10) or active (20) status</p> <p>SCMTY-SUBCAT-NAME = indicate the subcategory to assign to the new event, if applicable</p> <p>ALLOCATION-FACTOR = the allocation factor to use for the calculation of the capital credit allocation</p> <p>BILL-LINE-CHAR-TYPE-SQ = char type identifying bill line containing SQ info</p> <p>BILL-LINE-CHAR-VAL-SQ = char value identifying bill line containing SQ info</p> <p>BILL-LINE-CHAR-TYPE-SALES = char type identifying bill line containing sales (revenue) info</p> <p>BILL-LINE-</p>		

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
				CHAR-VAL-SALES = char value identifying bill line containing sales (revenue) info FISCAL-YEAR = indicate the fiscal year to assign to the new event MIN-CALC-AMT = indicate the minimum calculated amount needed to create an event MAX-ERRORS		
CPCRRETR	CIPCCCRB	<p>The capital credit retirement process creates retirement service credit events for active capital credit memberships.</p> <p>THIS BACKGROUND PROCESS IS ONLY APPLICABLE TO COMPANIES THAT ISSUE CAPITAL CREDITS TO THEIR CUSTOMERS.</p> <p>Refer to Capital Credit Retirement for more information.</p> <p>The program that is delivered one possible method for calculating capital credit retirement amounts. If the logic provided does not meet your business needs, you must adapt the base package program to your specific needs. To do this, please follow the standard steps:</p> <ul style="list-style-type: none"> Copy the base package program to your own program. Your own program should be prefixed with the letters CM (which stands for "customer modification"). This is important as it prevents the upgrade process from 	Yes	SC-EVT-TYPE-CD = service credit event type to use for new events SC-EVT-STATUS-FLG = Indicate if the events should be created in pending (10) or active (20) status PCT-ALLOC-TO-RETIRE = Percentage of balance to retire SCMTY-SUBCAT-NAME = indicate the subcategory to	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		overwriting your new logic. <ul style="list-style-type: none"> Modify the new program as per your business needs. Create a batch control record for your new background process, prefixed with <i>CM</i>. 		limit retirement to a single subcategory (if blank, all subcategories are processed) CTRL-ID = stamped onto new events - used for restart control MAX-ERRORS		
F1-AVALG	Java	This process regenerates algorithm type and their related algorithm information to be displayed by the Application Viewer. It produces a series of XML files in a designated folder under the application viewer /data folder. Refer to Application Viewer Generation for more information on this background process is used.	No	MAX-ERRORS	No	NA
F1-AVMO	Java	This process regenerates maintenance object information to be displayed by the Application Viewer. It reads the meta-data maintenance object information and produces a series of XML files in a designated folder under the application viewer /data folder. Refer to Application Viewer Generation for more information on this background process is used.	No	MAX-ERRORS	No	NA
F1-AVTBL	Java	This process regenerates table and column information to be displayed by the Application Viewer. It reads the meta-data table and related entities and produces a series of XML files in a designated folder under the application viewer /data folder. Refer to Application Viewer Generation for more information on this background process is used.	No	MAX-ERRORS	No	NA
MASSCNCL	CIPBMCNB	The mass bill cancellation process removes	Yes	BILL-CYC-CD=	Yes	100/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>an entire batch of bills that were created by the BILLING process.</p> <p>Refer to Canceling A Batch Of Bills After They're Complete for more information.</p>		<p>the bill cycle associated with the bills</p> <p>WIN-START-DT=the bill cycle window start date associated with the bills. This should be entered in the format YYYY-MM-DD.</p> <p>BILL-DT= the date on which the bills were created. This should be entered in the format YYYY-MM-DD.</p> <p>MAX-ERRORS</p>		
MASSROBL	CIPBMROB	<p>The mass bill reopen process reopens an entire batch of bills that were completed by the BILLING process.</p> <p>Refer to Reopening A Batch Of Bills After They're Complete for more information.</p>	Yes	<p>BILL-CYC-CD= the bill cycle associated with the bills</p> <p>WIN-START-DT= the bill cycle window start date associated with the bills</p> <p>BILL-DT= the date on which the bills were created. This should be entered in the format YYYY-MM-DD.</p> <p>MAX-ERRORS</p>	Yes	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
NEWLANG	CIPZLNGB	<p>This Create New Language batch program duplicates all language specific rows from the source to the target language. Once this program has run you will need to translate the language specific columns.</p> <p>This program can be rerun at anytime. It will only duplicate entries where they do not already exist.</p> <p>Note if you run this program with the "DELETE" action then the source and target language must be the same.</p> <p>If you have any questions, please see your implementation team for more information.</p>	Yes	<p>SOURCE-LANGUAGE= source language code</p> <p>TARGET-LANGUAGE= target language code</p> <p>PROGRAM-ACTION= (DELETE or INSERT)</p> <p>MAX-ERRORS</p>	No	200/15
REGCNST	CIPMRCVB	<p>This process selects all service points (SP) that meet the following criteria:</p> <ul style="list-style-type: none"> - they reference the input MR_CYC_CD (Service Cycle) - their SP type references the input SVC-TYPE-CODE (service type, e.g., Electric) and SP-SUBTYPE-FLG (sub type, e.g., Metered) <p>For each SP, it extracts the current and potential transformers linked to the SP (these are defined as equipment linked to the SP and are identified by CT-ITEM-TYPE and PT-ITEM-TYPE).</p> <p>For each current transformer linked to the service point, it extracts the CT ratio (identified by CT-RATIO-CHAR). The CT ratio value is defined as a ratio in the format a:b, e.g., 400:5, 1500:5, 7200:120, etc. The first number in the ratio is divided by the second number to derive the actual CT ratio. If a characteristic is not found, a CT ratio of 1 is assumed.</p> <p>It does the same for each potential transformer linked to the service point (using PT-RATIO-CHAR).</p> <p>It then calculates the product of the various CT and PT ratios. If the product is not</p>	Yes	<p>MR-CYC-CD = Service Cycle</p> <p>CT-ITEM-TYPE = CT Item Type</p> <p>CT-RATIO-CHAR = CT Ratio Characteristic Type on Item type (for equipment)</p> <p>PT-ITEM-TYPE = PT Item Type</p> <p>PT-RATIO-CHAR = PT Ratio Characteristic Type on Item type (for equipment)</p> <p>DIAL-CONST = Dial Constant Characteristic Type on meter</p> <p>C-TO-DO-TYPE = Constant To Do</p>	Yes	100/10

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
		<p>divisible by 5, a To Do entry is created (To Do type = D-TO-DO-TYPE – this should be RGCMOD) for a user to verify the ratios, as ratios for standard transformers are divisible by 5.</p> <p>It then finds the meter currently installed at the service point and extracts its dial constant from a meter characteristic identified by DIAL-CONST.</p> <p>It then calculates a register constant by taking the product of all CT ratios and PT ratios and multiplying this value by the meter's dial constant.</p> <p>It then compares the calculated register constant with the actual register constant of every register linked to the meter. If they differ, a To Do entry is created (To Do type = C-TO-DO-TYPE – this should be RGCDIFF).</p> <p>Refer to Validating Register Constants for more information.</p>		<p>Type D-TO-DO-TYPE = Divisible5 To Do Type</p> <p>SVC-TYPE-CD = Service Type Code</p> <p>SP-SUBTYPE-FLG = SP Subtype (this will typically be "M" for metered service points)</p> <p>MAX-ERRORS</p>		
UPDERR	CIPZUESB	<p>The process updates <i>In Progress</i> threads in an abnormally terminated batch run to be <i>Error</i>. When at least one thread is in <i>Error</i>, this process also updates the status of the batch run to be <i>Error</i>.</p> <p>Refer to Dealing With Abnormally Terminated Background Processes for information about when and why this process would be executed.</p>	No	<p>BATCH-CD-IN-PROGRESS = the batch control ID of the abnormally terminated batch process</p> <p>BATCH-NBR-IN-PROGRESS = the run number of the abnormally terminated batch process</p> <p>UPD-ALL-THRDS-SW must be Y or N. A value of Y means that the status of all <i>In</i></p>	No	N/A

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Error Generates To Do	Records Between Commits / Minutes Between Cursor Re-Initiation
				<p><i>Progress</i> threads will be changed to <i>Error</i>. A value of N means that the next parameter must be supplied.</p> <p>BATCH-THRD-NBR-IN-PROGRESS = the thread number whose status should be changed from <i>In Progress</i> to <i>Error</i>. This parameter should only be specified if UPD-ALL-THRDS-SW = N.</p> <p>MAX-ERRORS</p>		

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

To Do Entry Processes

These are background processes whose main purpose is to generate To Do Entry records based on a certain condition. Refer to [To Do Entries Created By Background Processes](#) for the details. The section that appears below simply lists these processes.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
TD-BCUPL	CIPQBCEB	This background process creates a To Do entry for	No	MAX-ERRORS	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		every billable charge upload record that's in error.			
TD-BIERR	CIQOBIEB	This background process creates a To Do entry for every bill that's in error.	Yes	MAX-ERRORS	200/15
TD-BSERR	CIQOBSEB	This background process creates a To Do entry for every bill segment that's in error.	Yes	MAX-ERRORS	200/15
TD-BTERR	CIQOBERB	This background process creates To Do Entry for any other batch processes that ended in error. A To Do Entry is only created if one does not already exist.	No	MAX-ERRORS	200/15
TD-CCCB	CIQCCCCB	This background process creates a To Do entry for customer contacts that have been flagged to generate a To Do entry on a future date. Note well, most To Do background processes create To Do entries in the <i>pending</i> state. If the customer contact indicates a specific user should be notified (as opposed to notifying a group of users – a role), the To Do entry will be created in the <i>being worked</i> state and it will be assigned to the designated user.	Yes	LEAD-DAYS = Number of days before the customer contact's reminder date that the To Do entry should be created. Valid values of 0 to 99 are acceptable. MAX-ERRORS	200/15
TD-CEVT	CIQCEVB	This background process creates a To Do entry for collection events that should generate a To Do entry.	Yes	MAX-ERRORS	200/15
TD-CLERR	CIQCLEB	This background process creates a To Do Entry for any batch process that has root objects that created an error. A To Do Entry is only created if one does not already exist.	Yes	MAX-ERRORS	200/15
TD-DTCST	CIQDTCB	This background process creates a To Do entry for deposit control staging / tender control staging records that are in error.	No	MAX-ERRORS	200/15
TD-ECBK	CIQENHB	This background process creates To Do entries for held orders.	Yes	DAYS-BEF-CALLBK = Number of days before the order's reminder date that the To Do entry should be	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				created MAX-ERRORS	
TD-EPND	CIPQENPB	This background process creates To Do entries for pending orders.	Yes	MAX-ERRORS	200/15
TD-FAUPL	CIPQFAEB	This background process creates a To Do entry for every field activity upload staging record that's in error.	Yes	MAX-ERRORS	200/15
TD-FACT	CIPQFARB	This background process creates a To Do entry for every field activity remark record that's in error.	Yes	MAX-ERRORS	200/15
TD-HILO	CIPQMHLB	<p>This background process creates a To Do entry for every meter read that has at least one register read that failed high/low review.</p> <p>Refer to TD-HILO - To Do for Meter Read High/Low Errors for more information about this process, including its parameter values. This section also provides more information about the To Do entry created.</p>	Yes	NO-OF-DAYS = System will auto-complete 'Open' To Do entries of this To Do Type whose creation date is before the business date minus this parameter. (Optional). VERY-LOW-PERCENTAGE-CUTOFF USE-ON-BILL-VERY-LOW-READS USE-ON-BILL-LOW-READS VERY-HIGH-PERCENTAGE-CUTOFF USE-ON-BILL-VERY-HIGH-READS USE-ON-BILL-HIGH-READS MULTIPLE-REGISTERS-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				PRIORITY MAX-ERRORS	
TD-MODTL	CIPQODMB	This background process creates a To Do entry for every disputed match event	Yes	NO-OF-DAYS = Number of days old the match event must be before a To Do entry is created (this prevents young entries from appearing on To Do lists) MAX-ERRORS	200/15
TD-MONTL	CIPQONMB	This background process creates a To Do entry for every open, non-disputed match event	Yes	NO-OF-DAYS = Number of days old the match event must be before a To Do entry is created (this prevents young entries from appearing on To Do lists) MAX-ERRORS	200/15
TD-MRRER	CIPQMRRB	This background process creates a To Do entry for every meter read remark record that's in error.	Yes	MAX-ERRORS	200/15
TD-MRUPL	CIPQMRUB	This background process creates a To Do entry for every uploaded meter read that's in error.	Yes	MAX-ERRORS	200/15
TD-NCDEX	CIPQNCDB	This background process creates a To Do entry for every non-cash deposit that is due to expire within the next X days (X is a parameter). Note. The process checks for completed To Do entries for this To Do Type and Account Id and will not create a new To Do if a completed one exists.	Yes	DAYS-TO-EXPIRY = Number of days prior to NCD expiration that the entry should be created MAX-ERRORS	200/15
TD-NOBC	CIPQNBCB	This background process creates a To Do entry for	Yes	MAX-ERRORS	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		every account that doesn't have a bill cycle but has active service agreements.			
TD-NOMR	CIPQNRCB	This background process creates a To Do entry for every metered service point that doesn't have a service cycle. Where the service point is in service and connected.	Yes	MAX-ERRORS	200/15
TD-NTDWN	CIPQNTDB	This background process creates a To Do entry for every notification download staging record that's in error.	Yes	MAX-ERRORS	200/15
TD-NTUPL	CIPQNTUB	This background process creates a To Do entry for every notification upload staging record that's in error.	No	MAX-ERRORS	200/15
TD-PYERR	CIPQPAYB	This background process creates a To Do entry for every payment that's in error or that is unfrozen.	Yes	MAX-ERRORS	200/15
TD-PYUPL	CIPQPYUB	This background process creates a To Do entry for every payment staging record that's in error.	No	MAX-ERRORS	200/15
TD-SEVT	CIPQSEVB	This background process creates a To Do entry for severance events that should generate a To Do entry.	Yes	MAX-ERRORS	200/15
TD-SPRO	CIPQSVPB	This background process creates a To Do entry for cancelled severance processes that have <i>dispatched</i> field activities that should be cancelled.	Yes	MAX-ERRORS	200/15
TD-SSFTL	CIPQSSLB	This background process creates a To Do entry for pending start/stops that are older than the number of days specified. This catches start/stop requests that have gone unfulfilled.	Yes	NO-OF-DAYS = the number of days old a pending start / stop SA must be to be considered unfulfilled. MAX-ERRORS	200/15
TD-UNBAL	CIPQPYEB	This background process creates a To Do entry for every payment event that's unbalanced.	Yes	MAX-ERRORS	200/15
TD-WEXTL	CIPQWEXB	This background process creates a To Do entry for every workflow event that's in error.	Yes	MAX-ERRORS	200/15
TD-WOEV	CIPQWEVB	This background process creates a To Do entry for write-off events that should generate a To Do	Yes	MAX-ERRORS	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		entry.			
TD-XAIDN	CIPOXADB	This background process creates a To Do entry for every XAI Download Staging exception.	No	MAX-ERRORS	200/15
TD-XAIUP	CIPOXAUB	This background process creates a To Do entry for every XAI Upload Staging in error.	Yes	MAX-ERRORS	200/15

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Object Validation Processes

These background processes are run to validate the master data objects. These programs are typically only run as part of the conversion and upgrade processes.

Another use for these programs. In addition to validating your objects after conversion or an upgrade, the validation programs listed below have another use. Say for example, you want to experiment with changing the validation of a person and you want to determine the impact of this new validation on your existing persons. You could change the validation and then run the person validation object – it will produce errors for each person that fails the new validation.

Refer to [Validate Information In The Staging Tables](#) for more information about these processes and where their errors appear.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
VAL-ACCT	CIPVACCB	Validate the account object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-BCHG	CIPVBCGB	Validate the billable charge object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status MAX-ERRORS	200/15
VAL-CEVT	CIPVCEVB	Validate the contract option event object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-CFG	CIPVMTGB	Validate the meter configuration object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-COLL	CIPVCLPB	Validate the collection process object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				row. STATUS1=validate rows with this status STATUS2=validate rows with this status MAX-ERRORS	
VAL-COP	CIPVCOPB	Validate the contract option object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-DCL	CIPVDCRB	Validate the declaration object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th	

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				row. MAX-ERRORS	
VAL-DTST	CIPVDTTB	Validate the device test object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status STATUS3=validate rows with this status STATUS4=validate rows with this status STATUS5=validate rows with this status MAX-ERRORS	
VAL-FA	CIPVFACB	Validate the field activity object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status STATUS3=validate rows with this status STATUS4=validate rows with this status MAX-ERRORS	
VAL-FO	CIPVFORB	Validate the field order object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row.	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				STATUS1=validate rows with this status STATUS2=validate rows with this status STATUS3=validate rows with this status STATUS4=validate rows with this status STATUS5=validate rows with this status MAX-ERRORS	
VAL-IDS	CIPVIDSB	Validate the Interval Data Set object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-INPF	CIPVINPB	Validate the interval profile object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-IRDS	CIPVIRSB	Validate the interval register data set object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-ITEM	CIPVITMB	Validate the item object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				example 10 to process every 10 th row. MAX-ERRORS	
VAL-IVS	CIPVIVSB	Validate the interval value set object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-LL	CIPVLNDB	Validate the landlord agreement object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-MTR	CIPVMTRB	Validate the meter object	Yes	OVRD-LOW-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-PER	CIPVPERB	Validate the person object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-PREM	CIPVPRMB	Validate the premise object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-SA	CIPVSVAB	Validate the service agreement object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-SARL	CIPVSREB	Validate the SA Relationship object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				example 10 to process every 10 th row. MAX-ERRORS	
VAL-SEVP	CIPVSEPB	Validate the severance process object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status MAX-ERRORS	200/15
VAL-SCM	CIPVSCMB	Validate the service credit membership object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				example 10 to process every 10 th row. MAX-ERRORS	
VAL-SP	CIPVSPPB	Validate the service point object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-TDS	CIPVTDSB	Validate the TOU data set object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	200/15
VAL-TMAP	CIPVTMBB	Validate the TOU map object	Yes	OVRD-LOW-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. MAX-ERRORS	
VAL-WFP	CIPVWPRB	Validate the workflow process object	Yes	OVRD-LOW-ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status MAX-ERRORS	200/15
VAL-WOP	CIPVWOPB	Validate the write off process object	Yes	OVRD-LOW-	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				ID=key value to override the calculated start-key value OVRD-HIGH-ID=key value to override the calculated end-key value SKIP-ROWS= nth row to be processed, for example 10 to process every 10 th row. STATUS1=validate rows with this status STATUS2=validate rows with this status MAX-ERRORS	

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Referential Integrity Validation Processes

The following table lists every background process that validates transaction data using the same program **CIPVRNVB**. These programs are typically run as part of the conversion and upgrade processes.

In all cases, the processes are not multi-threaded and do not include extra parameters. In addition, Records Between Commits and Minutes Between Cursor Re-Initiation are not applicable.

Refer to [Validate Information In The Staging Tables](#) for more information about these processes and where their errors appear.

Batch Control ID	Description
CIPVAAPV	Foreign Key validation for Account Automatic Payment

Batch Control ID	Description
CIPVACHV	Foreign Key validation for Account Characteristics
CIPVACPV	Foreign Key validation for Account Person Relationship
CIPVADJV	Foreign Key validation for Adjustment
CIPVAPAV	Foreign Key validation for Premise Alternate Address
CIPVAPRV	Foreign Key validation for A/P Check Request
CIPVARHV	Foreign Key validation for Collection Agency Referral History
CIPVARSV	Foreign Key validation for Credit Review Schedule
CIPVBCHV	Foreign Key validation for Bill Characteristic
CIPVBCGV	Foreign Key validation for Billable Charge
CIPVBCLV	Foreign Key validation for Billable Charge Line
CIPVBFVV	Foreign Key validation for Bill Factor Value
CIPVBLLV	Foreign Key validation for Bill Header
CIPVBLMV	Foreign Key validation for Bill Messages
CIPVBLRV	Foreign Key validation for Bill Routing
CIPVBSAV	Foreign Key validation for Bill - SA Balance Snapshot
CIPVBSCV	Foreign Key validation for Bill Segment Calc Header
CIPVBSIV	Foreign Key validation for Bill Segment Item
CIPVBSLV	Foreign Key validation for Bill Segment Calc Line
CIPVCARV	Foreign Key validation for Collection Agency Referral
CIPVCCFV	Foreign Key validation for Contract Option Characteristic
CIPVCECV	Foreign Key validation for Collection Event/Customer Contact
CIPVCLPV	Foreign Key validation for Collection Process
CIPVCLSV	Foreign Key validation for Collection Process Service Agreement
CIPVCOLV	Foreign Key validation for Contract Option Language
CIPVCRTV	Foreign Key validation for Credit Rating History
CIPVCSCV	Foreign Key validation for Customer Contact
CIPVCVCV	Foreign Key validation for Cont Opt Event Characteristic
CIPVCVNV	Foreign Key validation for Collection Event
CIPVDTCV	Foreign Key validation for Device Test Characteristic
CIPVDTMV	Foreign Key validation for Device Test Component
CIPVDTRV	Foreign Key validation for Device Test Result
CIPVDTTV	Foreign Key validation for Device Test
CIPVFAFV	Foreign Key validation for Field Activity
CIPVFAHV	Foreign Key validation for Field Activity Characteristics
CIPVFALV	Foreign Key validation for Field Activity Log

Batch Control ID	Description
CIPVFARV	Foreign Key validation for Field Activity / Remark
CIPVFORV	Foreign Key validation for Field Order
CIPVFSTV	Foreign Key validation for Field Activity Step
CIPVFTFV	Foreign Key validation for Financial Transaction
CIPVFTGV	Foreign Key validation for Fin'l Transaction Gen Ledger
CIPVFTPV	Foreign Key validation for Fin'l Transaction Process
CIPVIEQV	Foreign Key validation for Item / Equipment
CIPVILHV	Foreign Key validation for Item Location History
CIPVINLV	Foreign Key validation for Interval Profile Lang
CIPVITCV	Foreign Key validation for Item Characteristic
CIPVITFV	Foreign Key validation for Interval Data
CIPVITVV	Foreign Key validation for Interval Value
CIPVLLDV	Foreign Key validation for Landlord Agreement Detail
CIPVMEQV	Foreign Key validation for Meter / Equipment
CIPVMIDV	Foreign Key validation for Meter Ids
CIPVMLHV	Foreign Key validation for Meter Location History
CIPVMRDV	Foreign Key validation for Meter Read
CIPVMRCV	Foreign Key validation for Meter Read Characteristics
CIPVMRMV	Foreign Key validation for Meter Read Remark
CIPVMSGV	Foreign Key validation for Account Bill Messages
CIPVMTCV	Foreign Key validation for Meter Characteristic
CIPVNBSV	Foreign Key validation for Non-billed Budget SA / SA
CIPVNCDV	Foreign Key validation for Non Cash Deposit
CIPVNPMV	Foreign Key validation for NBB SA Payment Schedule Parameter Values
CIPVNSPV	Foreign Key validation for NBB SA Scheduled Payments
CIPVPAOV	Foreign Key validation for Person Address Override
CIPVPAYV	Foreign Key validation for Payment Header
CIPVPCHV	Foreign Key validation for Premise Characteristic
CIPVPGOV	Foreign Key validation for Premise Geographic location
CIPVPIDV	Foreign Key validation for Person Identifier
CIPVPNMV	Foreign Key validation for Person Name
CIPVPPEV	Foreign Key validation for Person to Person
CIPVPPHV	Foreign Key validation for Person Phone
CIPVPRCV	Foreign Key validation for Person Characteristics
CIPVPSAV	Foreign Key validation for Person Seasonal Address

Batch Control ID	Description
CIPVPSGV	Foreign Key validation for Payment Segment
CIPVPYCV	Foreign Key validation for Payment Characteristics
CIPVREFV	Foreign Key validation for Interval Register Data
CIPVREGV	Foreign Key validation for Register
CIPVRGCV	Foreign Key validation for Register Characteristics
CIPVRGRV	Foreign Key validation for Register Read
CIPVSACV	Foreign Key validation for SA Characteristics
CIPVSAHV	Foreign Key validation for SA Rate Schedule History
CIPVSAOV	Foreign Key validation for SA Contract Terms
CIPVSAPV	Foreign Key validation for SA SP
CIPVSAQV	Foreign Key validation for SA Contract Quantity
CIPVSARV	Foreign Key validation for SA Recurring Charge
CIPVSCAV	Foreign Key validation for SCM Account
CIPVSCCV	Foreign Key validation for SCM Characteristic
CIPVSCFV	Foreign Key validation for Service Credit Event FT
CIPVSCOV	Foreign Key validation for SA Override Contract Option
CIPVSCPv	Foreign Key validation for SA Contract Option
CIPVSCVv	Foreign Key validation for Service Credit Event
CIPVSECV	Foreign Key validation for Severance Event/Customer Contact
CIPVSEdV	Foreign Key validation for Severance Event Dependency
CIPVSEFV	Foreign Key validation for Severance Event/Field Activity
CIPVSEGV	Foreign Key validation for Bill Segment
CIPVSEPV	Foreign Key validation for Severance Process
CIPVSEQV	Foreign Key validation for SP / Equipment
CIPVSEVv	Foreign Key validation for Severance Event
CIPVSIEV	Foreign Key validation for SP Item Event
CIPVSIFV	Foreign Key validation for SA Interval Profile
CIPVSIHV	Foreign Key validation for History of Item at SP
CIPVSMEV	Foreign Key validation for SP Meter Event
CIPVSMGV	Foreign Key validation for SA Message
CIPVSMHV	Foreign Key validation for History of Meter at SP
CIPVSMIV	Foreign Key validation for Multi Item
CIPVSPCV	Foreign Key validation for Service Point Characteristics
CIPVSPGV	Foreign Key validation for Service Point Geo Location
CIPVSPMV	Foreign Key validation for Unbadged Items at SP

Batch Control ID	Description
CIPVSPDV	Foreign Key validation for Service Point/Operation Area
CIPVSQTV	Foreign Key validation for Bill Segment Service Quantity
CIPVSRLV	Foreign Key validation for SA Relationship
CIPVSRRV	Foreign Key validation for Bill Segment Register Read
CIPVSSCV	Foreign Key validation for SA SP Characteristics
CIPVSSFV	Foreign Key validation for SA SP Field Activity
CIPVSTMV	Foreign Key validation for SA TOU Map
CIPVTBVV	Foreign Key validation for TOU Bill Factor Value
CIPVTCVV	Foreign Key validation for TOU Contract Value
CIPVTMLV	Foreign Key validation for TOU Map Lang
CIPVTNCV	Foreign Key validation for Payment Tender Characteristics
CIPVTNDV	Foreign Key validation for Payment Tender
CIPVTOFV	Foreign Key validation for TOU Data
CIPVTRNV	Foreign Key validation for Trend
CIPVWECV	Foreign Key validation for Workflow Event Context
CIPVWEDV	Foreign Key validation for Workflow Event Dependency
CIPVWEVV	Foreign Key validation for Workflow Event
CIPVWOCV	Foreign Key validation for Write Off Event/Customer Contact
CIPVWOPV	Foreign Key validation for Write Off Process
CIPVWOSV	Foreign Key validation for Write Off Process/Service Agreement
CIPVWOVV	Foreign Key validation for Write Off Event
CIPVWPCV	Foreign Key validation for Workflow Process Context
CIPVWPRV	Foreign Key validation for Workflow Process

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Conversion Processes

These background processes are run only when converting or migrating data from external applications into the system. Your company may never use them depending upon your data migration strategy.

Refer to [The Conversion Tool](#) for more information about these processes and where their errors appear.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
CNV-ADM	CIPVADMB	Creates ADM triggers for converted accounts.	Yes	MAX-ERRORS	200/15
CNV-BAL	CIPVVSAB	Sets the correct balance for all SAs created during conversion. It creates adjustments that cause each SA's current and payoff balances to equal their balance in the prior system. Note, this process is referred to as the "tidy balances" process in the conversion tool document.	No	SET-PYOF-ZERO= Adjustment Type to set payoff balance to zero SET-CURR-ZERO = Adjustment Type to make current balance zero SET-PYOF-BAL= Adjustment Type to set payoff balance to the desired balance SET-CURR-BAL = Adjustment Type to set current balance to the desired balance and age BASE-DATE= Base Date of Aged Debt. This should be entered in the format YYYY-MM-DD. FILE-PATH = Path and filename of input data file MAX-ERRORS	200/15
CNV-BCG	CIPVCBCB	This process resets the Balance Control column on all FT's so that the FT's can be included in a balance control (see the last step below) after they have been transferred to production.	Yes	MAX-ERRORS	200/15

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

Conversion Processes Executed In The Staging Database

There are many other background processes that are only executed if you use the conversion tool to load historical data into your production database. These programs perform the following tasks:

- **Key Assignment Programs.** Background processes of this type assign random, clustered keys to the rows in the staging database.

A separate background process exists for every table with a system-assigned key that is supported by the conversion tool. The program names of these processes are documented in [The Conversion Tool](#) (scan for all references to “Key Assignment Program” for a matrix containing these program names).

- **Insertion Programs.** Background processes of this type insert converted rows into production from the staging database.

A separate background process exists for every table that is supported by the conversion tool. The program names of these processes are documented in [The Conversion Tool](#) (scan for all references to “Insertion Program” for a matrix containing these program names).

Purge Processes

These background processes are used to purge historical records from certain objects that generate a large number of entries and may become unwieldy over time.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
BCUP-PRG	CIPCDBSB	Purges <i>completed</i> billable charge upload objects.	Yes	NO-OF-DAYS = number of days after the creation date that a <i>completed</i> billable charge upload object should be purged MAX-ERRORS	200/15
FAUP-PRG	CIPODFSB	Purges <i>completed</i> field activity upload objects.	Yes	NO-OF-DAYS = number of days after the related field activity's schedule date that a <i>completed</i> field activity upload object should be purged	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				MAX-ERRORS	
MRUP-PRG	CIPMDUSB	Purges <i>completed</i> meter read upload objects.	Yes	NO-OF-DAYS = number of days after the meter read date that a <i>completed</i> meter read upload object should be purged MAX-ERRORS..AF1\07 BkgndProcess es.doc - SPLINKOverride Maximum Errors in Batch	200/15
PYUP-PRG	CIPPDUSB	Purges <i>completed</i> tender upload objects.	Yes	NO-OF-DAYS = number of days after the related tender's payment event's creation date that a <i>completed</i> tender upload object should be purged MAX-ERRORS	200/15
TD-PURGE	CIPQDTDB	Purges <i>completed</i> To Do entries.	Yes	NO-OF-DAYS = number of days after the completion date that a <i>completed</i> To Do entry should be purged DEL-ALL-TD-SW = Y or N. If this switch is Y, all <i>completed</i> To Do entries that are old enough will be deleted. If N, the next parameter	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				defines the specific type of To Do entry that will be deleted. DEL-TD-TYPE-CD. This parameter is only used if DEL-ALL-TD-SW is N. It contains the To Do type code whose <i>completed</i> entries will be deleted. MAX-ERRORS	
XMLUP-PR	CIPXDXUB	Purges <i>completed</i> XML upload objects.	Yes	NO-OF-DAYS = number of days after the completion date that a <i>completed</i> XML upload object should be purged MAX-ERRORS	200/15

Please refer to [Column Descriptions](#) for more information on the columns used in the table above.

ConfigLab Processes

The following table lists system background processes used in conjunction with the [Configuration Lab](#). The delivered system background processes are designed to copy sample DB processes from the demonstration database.

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
CL-APPCH	CIPYUPDB	The apply changes based on compare is used to update the current environment based on the results of a compare from a <i>ConfigLab</i> or <i>Compare Source</i> environment.	No	ENV-CODE = environment reference specified as a batch	200/15

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
		Refer to Configuration Lab for more information.		parameter on the batch run used to compare data. DB-PROC-CODE = The <i>Compare</i> DB process used to pull data from ENV-CODE. MAX-ERRORS	
CL-COPDB	CIPYSYCB	The copy "CL_" DB processes is used to compare DB processes from the demonstration database. It is assumed that the demonstration database has been registered as a <i>Compare Source</i> environment reference. Refer to Configuration Lab for more information.	No	ENV-CODE = environment reference representing a registered <i>Compare Source</i> environment. STATUS-ON-ADD = default root object status when root object action is <i>Add</i> . STATUS-ON-CHANGE = default root object status when root object action is <i>Change</i> . STATUS-ON-DELETE = default root object status when root object action is <i>Delete</i> . MAX-ERRORS	200/15

Archive and Purge Processes

The following table lists system background processes used in conjunction with the [Archive Engine](#). Note that the DB process that represents an archive or purge procedure defines the batch control used for the first step of a purge or archive. Although, the *CIPYCPRB* program (used universally for the first step of any archive or purge procedure) is delivered with the system, there are no system batch controls that reference it. It is included in this list for clarity.

For information on how to copy sample **Archive** and **Purge** DB processes from the demonstration database, refer to [How To Copy Samples From The Demonstration Database](#).

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
Special - Each archive or purge procedure defines its first step's batch control.	CIPYCPRB	The first step of any archive or purge process is to create primary archive roots objects. Note that for Purge DB processes, the environment reference batch parameter is optional. Refer to Archive Engine for more information.	Yes	ENV-CODE = environment reference of the target archive environment. TEST-MODE (Y/N) = If Y is specified, the program will write out information about primary roots to a log file.	200/15
AR-CRCHR	CIPYCRCB	The second step of any archive or purge process is to create child archive roots objects. Note that for Purge DB processes, the environment reference batch parameter is optional. Refer to Archive Engine for more information.	Yes	ENV-CODE = environment reference of the target archive environment. DB-PROC-CODE = The archive or purge DB process that specifies the batch control used to create primary archive root objects (step 1 of any archive or purge). MAX-ERRORS	200/15
AR-PRFK	CIPYRFKB	The third step of any archive or purge process is to check recursive foreign key relationships. Note that for Purge DB processes, the environment reference batch parameter is optional. Refer to Archive Engine for more information.	Yes	ENV-CODE = environment reference of the target archive environment. DB-PROC-CODE = The archive or purge DB process that specifies the batch control used	

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				to create primary archive root objects (step 1 of any archive or purge). MAX-ERRORS	
AR-DCDT	CIPYPARB	The fourth step of any archive or purge process is to move data to the target archive environment (in the case of an archive), or simply delete it (in the case of purge). Note that for <i>Purge</i> DB processes, the environment reference batch parameter is optional. Refer to Archive Engine for more information.	No	ENV-CODE = environment reference of the target archive environment. DB-PROC-CODE = The archive or purge DB process that specifies the batch control used to create primary archive root objects (step 1 of any archive or purge). MAX-ERRORS	200/15
AR-DCDTF	CIPYCARB	If desired, this background process may be run instead of AR-DCDT (above). This process calls the <i>Archive Copy Processing</i> algorithm specified on the DB Process Instruction to copy the data to a flat file instead of an archive environment. Note that for <i>Purge</i> DB processes, the environment reference batch parameter is optional. Refer to Archive Engine for more information.	Yes	ENV-CODE = environment reference of the target archive environment. DB-PROC-CODE = The archive or purge DB process that specifies the batch control used to create primary archive root objects (step 1 of any archive or purge). MODE = C (Copy data only), D (Delete records only), or B (Both	200/NA

Batch Control ID	Program Name	Description	Multiple Threads	Extra Parameters	Records Between Commits / Minutes Between Cursor Re-Initiation
				copy and delete). The default is B (Both). MAX-ERRORS	

Column Descriptions

The following descriptions explain the parameters used in the above tables:

- **Batch Control ID.** As described earlier, every background process has an associated batch control record. This column contains the unique identifier of each process' batch control record.
- **Program Name.** This is the name of the program.
- **Description.** This column describes each background process.
- **Multiple Threads.** This column indicates if the background process uses the thread number and thread count to control parallel processing. Refer to [Parameters Supplied to Background Processes](#) for more information.
- **Extra Parameters.** This column indicates if the background process uses additional parameters (in addition to those described under [Parameters Supplied to Background Processes](#)).
- **Error Generates To Do.** This column indicates if the background process generates a To Do entry for object-specific errors as described in [Processing Errors](#).
- **Records Between Commits / Minutes Between Cursor Re-Initiation.** These values represent the maximum number of records between commits to the database and the number of minutes between cursor re-initiations. The process will issue a commit whenever the maximum records threshold has been exceeded. And, whenever a commit is issued, the process checks if the number of minutes between cursor initiation has been exceeded and if so, it will re-initiate the cursor. These values may be overridden when a specific background process is submitted. Refer to [Parameters Supplied to Background Processes](#) for more information.

Batch Process Dependencies

The contents of this section illustrate the periodicity and dependencies between the various background processes described above.

Contents

[Batch Schedulers and Return Codes](#)
[The Nightly Processes](#)
[The Hourly Processes](#)

[The Workflow and XAI Processes](#)
[The Letter Processes](#)
[The Periodic Processes](#)
[The ConfigLab Processes](#)
[The Archive and Purge Processes](#)

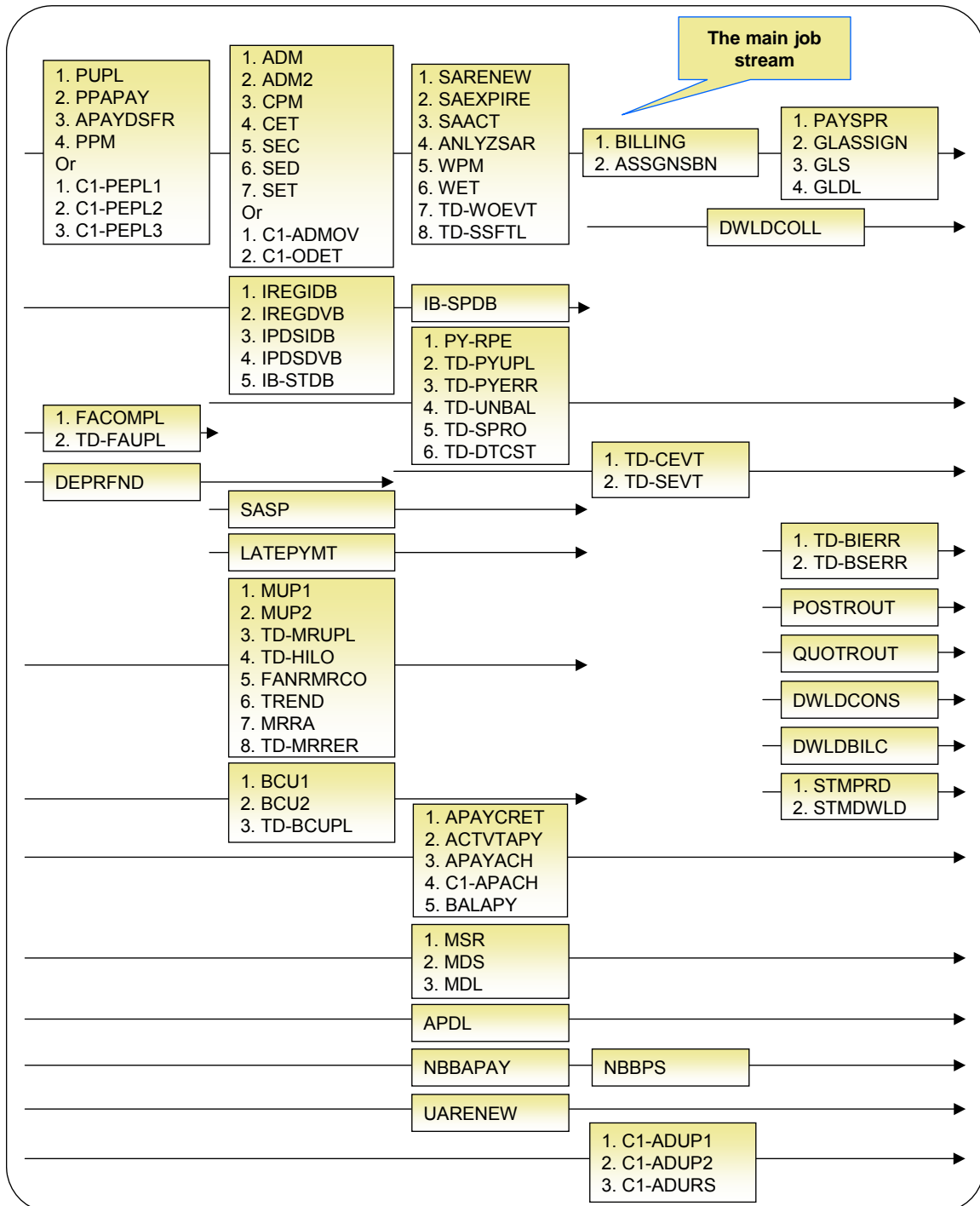
Batch Schedulers and Return Codes

If you use a batch scheduler (e.g., Control-M, Tivoli) to control the execution of your batch processes, it will be interested in the possible values of each process's return code. The return code is a number that indicates if the process ended successfully. All product processes will return one of the following return code values:

- **0** (zero). A value of zero means the batch process ended normally.
- **2**. A value of 2 means the batch process detected a fatal error and aborted.

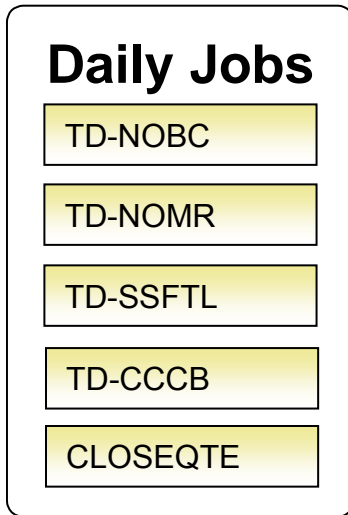
The Nightly Processes

The following diagram illustrates the dependencies between the batch processes.



The mnemonics in the boxes refer to the individual batch processes described above. When a box contains multiple processes, these processes must be run sequentially. When multiple boxes exist on a timeline, all processes in an earlier box must execute before the subsequent box is executed. Those timelines that appear beneath the Main Job Stream's timeline indicate when the timeline's respective processes can be executed in respect of the Main Job Stream.

The following diagram illustrates the daily batch processes for which there are no dependencies.

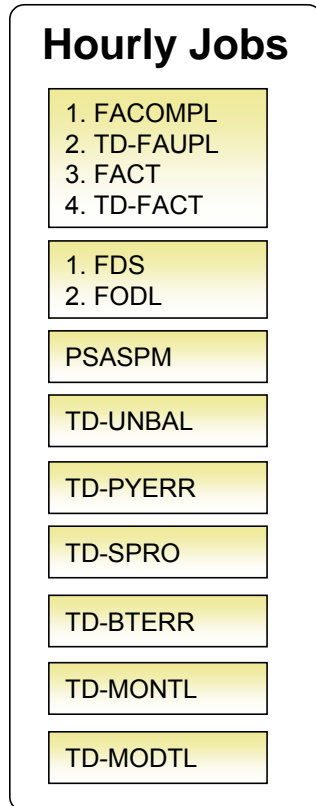


The mnemonics in the boxes refer to the individual batch processes described above.

No dependencies exist. As you can see, there are no dependencies between the boxes (meaning they may be run in parallel).

The Hourly Processes

The following diagram illustrates the dependencies between the hourly batch processes.

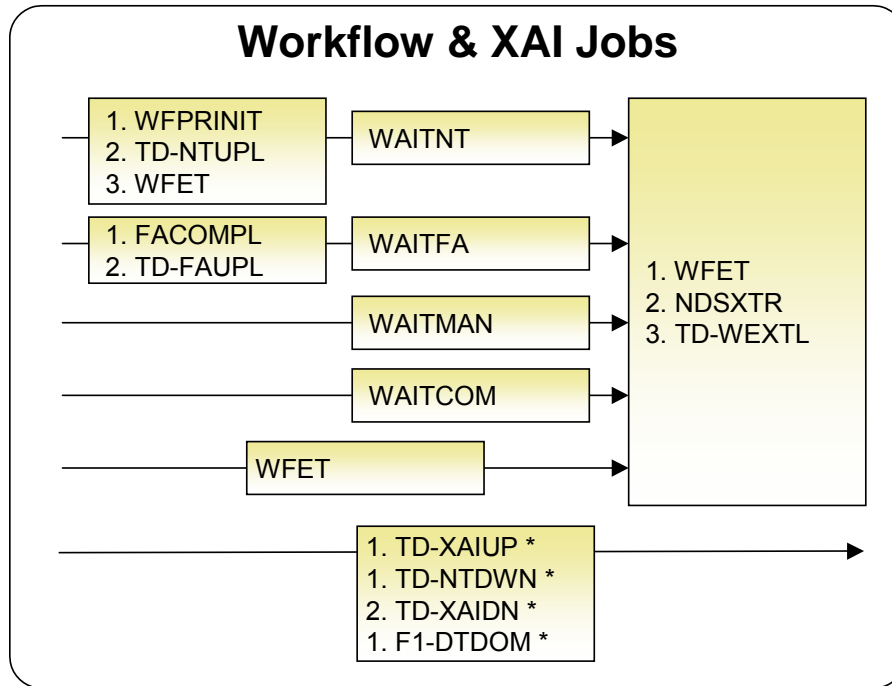


The mnemonics in the boxes refer to the individual batch processes described above. When a box contains multiple processes, these processes must be run sequentially.

No dependencies exist. As you can see, there are no dependencies between the boxes (meaning they may be run in parallel).

The Workflow and XAI Processes

The following diagram illustrates the dependencies between the workflow and XAI background processes. While these processes should be run at least once a day, you may want to consider running them more frequently (depending on how frequently you interface notifications and field activities into the system).



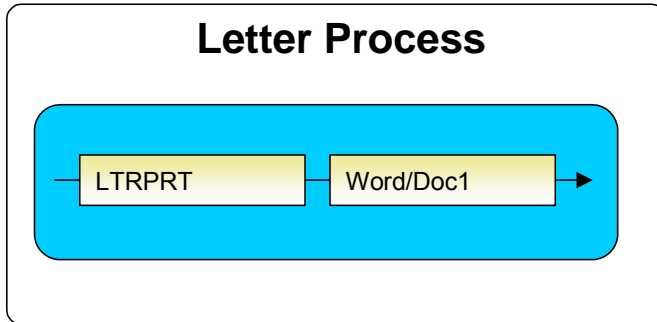
The mnemonics in the boxes refer to the individual batch processes described above. When a box contains multiple processes, these processes must be run sequentially. When multiple boxes exist on a timeline, all processes in an earlier box must execute before the subsequent box is executed.

* These processes create and/or clean up To Do entries for XAI upload staging, notification download staging, XAI download staging records or outbound messages in error. They are only applicable if your organization is using the XAI tool because only the XAI tool will mark one of these records in error.

The Letter Processes

To extract information for your various letters, only one background process, **LTRPRT**, is required regardless of the different types of letters you have. This process simply calls an algorithm plugged-in on the respective letter template to construct its flat-file content.

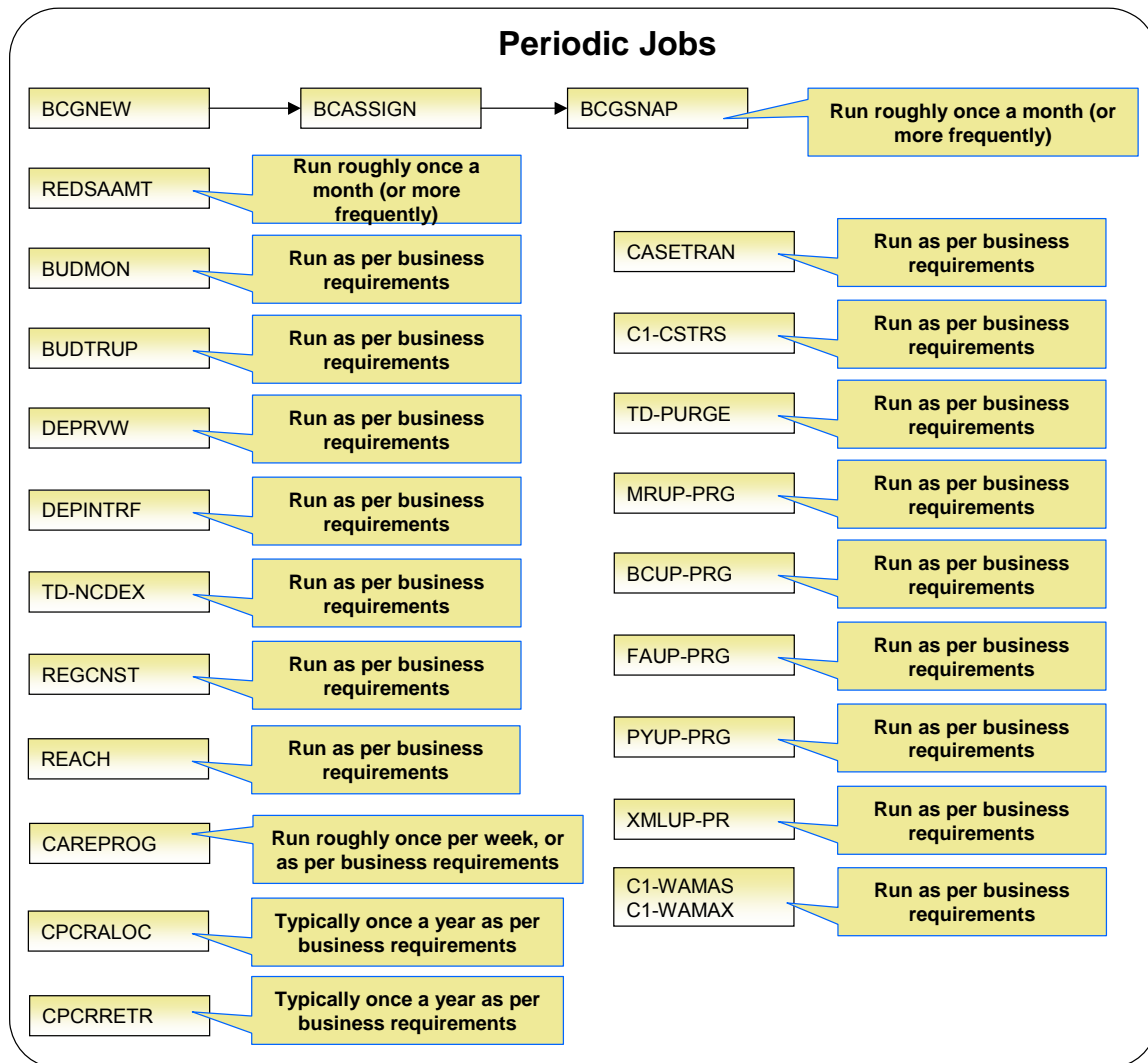
The following diagram illustrates the dependencies for the letter background process. While this process should be run at least on a daily basis, you may want to consider running it more frequently (depending on how frequently you produce letters).



The mnemonics in the boxes refer to the individual batch processes described above. When a box contains multiple processes, these processes must be run sequentially. When multiple boxes exist on a timeline, all processes in an earlier box must execute before the subsequent box is executed.

The Periodic Processes

The following diagram illustrates the dependencies between the periodic background processes. While many of these processes should be run at least on a monthly basis, you may want to consider running them more frequently (depending on business requirements).

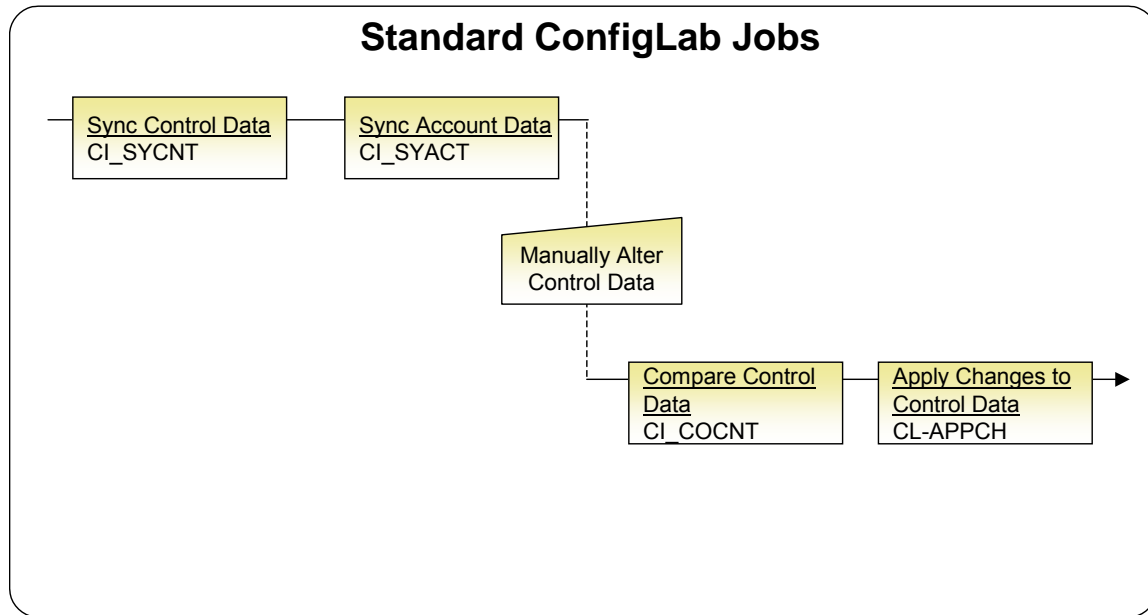


The mnemonics in the boxes refer to the individual batch processes described above.

Few dependencies exist. As you can see, there are few dependencies between the boxes (meaning they may be run in parallel).

The ConfigLab Processes

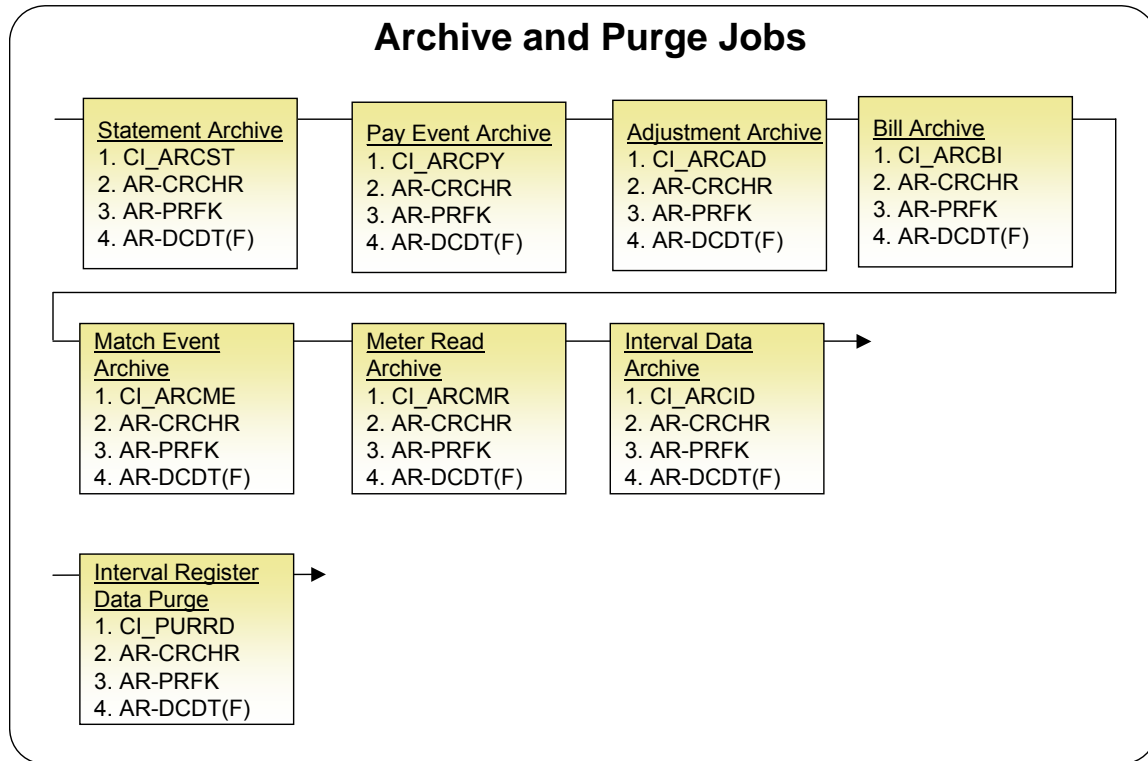
The following diagram illustrates the dependencies between the ConfigLab processes. The frequency of running ConfigLab processes is implementation specific. For more information on comparing data from an alternate environment, refer to [Configuration Lab](#).



Compare Sources and Targets. The [Configuration Lab](#) may be used with environments other than a **ConfigLab**. In cases where control and account data are pushed to a **Compare Target**, only the top two batch processes are executed. In cases where control data is pulled from a **Compare Source** environment, only the bottom two batch processes are executed.

The Archive and Purge Processes

The following diagram illustrates the dependencies between the sample archive and purge processes. The frequency of running archive and purge processes are implementation specific. For more information on archive and purge processes, refer to [Archiving and Purging](#) and [Archive Engine](#).



Steps 2, 3 and 4. Note that steps 2 and 3 are the same for all of the sample archive and purge jobs. For step 4, you can select from two background processes: [AR-DCDT](#) moves data to a target archive environment (or purges the data) and [AR-DCDTF](#) calls an algorithm that copies the data to a flat file. We have provided a sample algorithm for archiving meter read data to a file. If you want to use [AR-DCDTF](#) for other archive jobs, you must develop your own algorithms using [ARCD-MR](#) as an example.

How To Set Up A New Extract Processes

Several background processes delivered with the system are used to interface information out of the system. The topics in this section describe when and how to introduce an additional extract process.

Contents

- [Setting Up Meter Read Extracts](#)
- [Setting Up Automatic Payment Extracts](#)

Setting Up Meter Read Extracts

You will need a meter read extract for every mechanism your company uses to route meter read requests to the software that handles your meter reading requests. For example:

- You will need a meter read extract to interface records to your handheld device software. The **MDL** process delivered with the system is intended to be used to handle this function. This process will have to be populated to format the output records in keeping with the needs of your meter reading software.
- If you interface some meter read requests to automatic meter reading software, you will need a new meter read extract process.

If you need additional meter read extract processes, set up the following information:

- Add a new [batch control](#) record. Populate the fields as follows:
 - **ID.** Assign an easily recognizable unique ID for the meter read extract process.
 - **Description.** Enter a description of the meter read extract process.
 - **Accumulate All Instances.** Turn this switch on.
- Use [Route Type](#) to define the meter read extract process to be used for each route type.

Note. Route types are defined for each route linked to every service cycle. Refer to [Setting Up Service Cycles And Routes](#) for more information.

- It may be necessary to register the process with your scheduler software.

Setting Up Automatic Payment Extracts

You will need an automatic payment extract for every mechanism your company uses to route automatic payment requests to a financial institution / clearing house. For example:

- You will need an automatic payment extract to interface records to the Automated Clearing House (ACH) if you allow customer to pay via credit card or direct debit from a checking account. The **APAYACH** and **C1-APACH** processes delivered with the system are intended to be used to handle this function.

If you need additional automatic payment extract processes, set up the following information:

- Add a new [batch control](#) record. Populate the fields as follows:
 - **Batch Process.** Assign an easily recognizable unique ID for the automatic payment extract process.
 - **Description.** Enter a description of the automatic payment extract process.
 - **Accumulate All Instances.** Turn this switch on.
- Use [Auto Pay Route Type](#) to define the auto pay extract process to be used for each route type.

The Big Picture of Sample & Submit

Sample and Submit refers to the ability to create Activity Requests. This is functionality that enables an implementer to design an ad-hoc batch process using the configuration tools.

Some examples of such processes are:

- Send a letter to customers that use credit cards for auto pay and the credit card expiration date is within 30 days of the current date.
- Stop auto pay for customers that use credit cards as the form of payment if the credit card has already expired. Notify the customer that their auto pay agreement has been terminated and that they need to call to reinstate.
- Select auto pay accounts that have more than X non-sufficient fund penalties, stop the auto pay agreement and notify the customer.

Note that the terms activity request and sample & submit request may be used interchangeably.

Contents

- [Activity Type Defines Parameters](#)
- [Preview A Sample Prior To Submitting](#)
- [Credit Card Expiration Notice](#)
- [Exploring Activity Request Data Relationships](#)
- [Defining a New Activity Request](#)
- [Setting Up Activity Types](#)
- [Maintaining Sample & Submit Requests](#)

Activity Type Defines Parameters

For each type of process that your implementation wants to implement, you must configure an activity type to capture the appropriate parameters needed by the activity request.

Preview A Sample Prior To Submitting

To submit a new activity request, a user must select the appropriate activity type and enter the desired parameter values, if applicable.

After entering the parameters, the following actions are possible

- Click **Preview** to see a sample of records that satisfy the selection criteria for this request. This information is displayed in a separate map. In addition, the map displays the total number of records that will be processed when the request is submitted. From this map you can **Save** to submit the request, go **Back** to adjust the parameters or **Cancel** the request.
- Click **Cancel** to cancel the request.
- Click **Save** to skip the preview step and submit the request.

When an activity request is saved, the job is not immediately submitted for real time processing. The record is save in the status **Pending** and a monitor process for this record's business object is responsible for transitioning the record to **Complete**.

As long as the record is still **Pending**, it may be edited to adjust the parameters. The preview logic described above may be repeated when editing a record.

The actual work of the activity request, such as generating customer contact records to send letters to a set of customers, is performed when transitioning to **Complete** (using an enter processing algorithm for the business object).

Credit Card Expiration Notice

The base product supplies a sample process to find customers that use credit cards for auto pay and the credit card expiration date is within x days of the current date.

To this functionality the following configuration tasks are needed:

- Define an appropriate [customer contact class](#) and [type](#) to use.
- Define appropriate activity request Cancellation Reasons. Cancellation reasons are defined using a customizable [lookup](#). The lookup field name is **C1_AM_CANCEL_RSN_FLG**.
- Define an activity type for the business object **C1-NotifyExpiringCreditCardTyp**. You may define default parameter values for the number of days for expiration and customer contact class and type.

Exploring Activity Request Data Relationships

Use the following links to open the application viewer where you can explore the physical tables and data relationships behind the activity request functionality:

- Click [C1-ACM-ACTTY](#) to view the activity type maintenance object's tables.
- Click [C1-ACM-ACTRQ](#) to view the activity request maintenance object's tables.

Defining a New Activity Request

To design a new ad-hoc batch job that users can submit via Sample and Submit, first create a new Activity Type business object. The base product BO for activity type **C1-NotifyExpiringCreditCardTyp** may be used as a sample.

The business object for the activity request includes the functionality for selecting the records to process, display a preview map for the user to review and to perform the actual processing. The base product BO for activity request **C1-NotifyExpiringCreditCardReq** may be used as a sample. The following points highlight the important configuration for this business object:

- Special BO options are available for activity request BOs to support the [Preview Sample](#) functionality.
 - Activity Request Preview Service Script. This script is responsible for retrieving the information displayed when a user asks for a preview of a sample of records.
 - Activity Request Preview Map. This is the map that is invoked to display the preview sample results.
- The enter algorithm plugged into the **Complete** state is responsible for selecting all the records that satisfy the criteria and processing the records accordingly.

Setting Up Activity Types

Activity types define the parameters to capture when submitting an activity request via Sample and Submit. To set up an activity type, open **Admin Menu, Activity Type**.

The topics in this section describe the base-package zones that appear on the Activity Type portal.

Contents

[Activity Type List](#)

[Activity Type](#)

Activity Type List

The Activity Type [List zone](#) lists every activity type. The following functions are available:

- Click a [broadcast](#) button to open other zones that contain more information about the adjacent activity type.
- Click the **Add** link in the zone's title bar to add a new activity type.

Activity Type

The Activity Type zone contains display-only information about an activity type. This zone appears when an activity type has been broadcast from the Activity Type List zone or if this portal is opened via a drill down from another page. The following functions are available:

- Click the **Edit** button to start a business process that updates the activity type.
- Click the **Delete** button to start a business process that deletes the activity type.
- Click the **Duplicate** button to start a business process that duplicates the activity type.
- State transition buttons are available to transition the activity type to an appropriate next state.

Please see the zone's help text for information about this zone's fields.

Maintaining Sample & Submit Requests

Use the Sample and Submit transaction to view and maintain pending or historic activity requests. Navigate using **Main Menu, Batch, Sample & Submit Request**.

Contents

[Sample & Submit Request Query](#)

[Sample & Submit Request Portal](#)

Sample & Submit Request Query

Use the [query portal](#) to search for an existing sample & submit request. Once a request is selected, you are brought to the maintenance portal to view and maintain the selected record.

Sample & Submit Request Portal

This portal appears when a sample & submit request has been selected from the Sample & Submit Request Query portal.

The topics in this section describe the base-package zones that appear on this portal.

Contents

[Sample & Submit](#)

Sample & Submit Log

Sample & Submit

The Sample & Submit zone contains display-only information about an activity (sample & submit) request. The following functions are available:

- Click the **Edit** button to modify the parameters. Refer to [Preview A Sample Prior to Submitting](#) for more information.
- If the activity request is in a state that has valid next states, buttons to transition to each appropriate next state are displayed.

Please see the zone's help text for information about this zone's fields.

Sample & Submit Log

This is a standard [log zone](#).

Defining SA Relationship Options

We use the term “SA Relationship” to describe functionality that supports the following situations:

- When companies other than your own provide a service to your customers AND you have some type of interaction with these companies. For example, in a deregulated market, customers deal with both distribution and energy supply companies. These companies typically exchange a great deal of information about their joint customers.
- When multiple rates are associated with a service (where each rate corresponds with a sub-category of service). For example, most water companies charge for both water and wastewater service using separate rates for each. While it is possible to set up water and wastewater as separate service agreements, the SA relationship functionality allows you to set up a single “master” service agreement (for the water service) and associate with it a “sub” service agreement (for the wastewater service).
- When a party representing a group of customers negotiates a contract that is applied over and above those of the individual service agreements. For example, the head office of a national chain may negotiate for additional discounts that should be calculated together or individually. The SA relationship functionality may be used to track the covered service agreements and to calculate and transfer discounts to the head office’s service agreement. Refer to the special discounts section.

Warning! Setting up the SA relationship control tables is as challenging as your organization’s business rules. If you don’t have requirements similar to those described above, you don’t have to set up anything. If you have these types of requirements, your setup process will be taxing as you must design and set up control tables that manage the financial and consumption interactions that take place between you, your customers, and the various service providers.

The topics in this section describe tables that control your SA relationship functionality. Refer to [Defining Workflow and Notification Options](#) for a description of the tables that control how your organization communicates with the service providers who provide service to your customers.

Contents

[The Big Picture of SA Relationships and Service Providers](#)
[Designing Your SA Relationship Types and Service Providers](#)
[Setting Up SA Relationship Information](#)
[Negotiated Terms](#)

The Big Picture of SA Relationships and Service Providers

You must set up service providers if companies other than your own provide a service to your customers AND you have some type of interaction with these companies. You will have one service provider for each such company. For example, if you are a distribution company in a deregulated market, you will have a service provider for each company that provides any of the following services:

- Energy (commonly referred to as Energy Supply Companies, Energy Service Providers, Retailers, and Suppliers)
- Meter service (commonly referred to as Meter Service Providers and Meter Agents)

- Meter reading (commonly referred to as Meter Data Management Agencies and Meter Reading Service Providers)
- Billing (commonly referred to as Billing Agents)

The topics in this section provide background information about service providers.

Contents

[Persons and Service Providers](#)
[Service Providers Are Linked To Service Agreements](#)
[Service Providers May Change Over Time](#)
[How To Set Up SA Relationships On A Customer's Service Agreement](#)
[When Your Company Is A Service Provider](#)
[Service Providers Have To Communicate About Customers](#)
[Relationships Between Service Providers](#)
[A Service Agreement Can Have Many Types Of Relationships](#)
[Billing Relationships](#)
[Consumption Relationships](#)
[Deposits Issues](#)
[Credit and Collection Issues](#)
[An Object-Oriented Perspective Of Service Providers](#)
[How Do You Communicate With Service Providers?](#)

Persons and Service Providers

A great deal of information about your service providers is defined using a person. For example, a service provider's name, address, phone numbers, electronic ID's, etc. are all defined on the person object.

In addition, every service provider must have a service provider object created for it. The service provider object contains information about a provider's relationships with the customer and your organization, for example:

- Do you calculate bills for the service provider? If so, do you have their rates or do they interface their charges to you? Refer to [Billing Relationships](#) for more information.
- Do you send the customers' consumption to the service provider? Do they send it to you? Refer to [Consumption Relationships](#) for more information.
- How are financial settlements between your organization and the service provider implemented (do you pay them when you get paid?, do you purchase the receivable for them?, etc.). Refer to [Service Providers Have Service Agreements Too](#) for more information.
- Etc.

Refer to [Designing Your SA Relationship Types and Service Providers](#) for more information.

Note. In some situations, you will need to set up a service provider for your own company. Refer to [When Your Company Is A Service Provider](#) for the details.

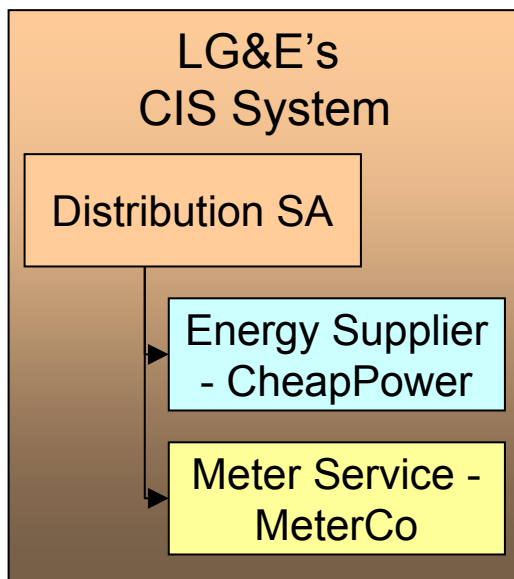
Service Providers Are Linked To Service Agreements

The following diagram illustrates a customer's bill for electric service in a deregulated market. Notice that there are separate sections for energy, distribution and meter service.

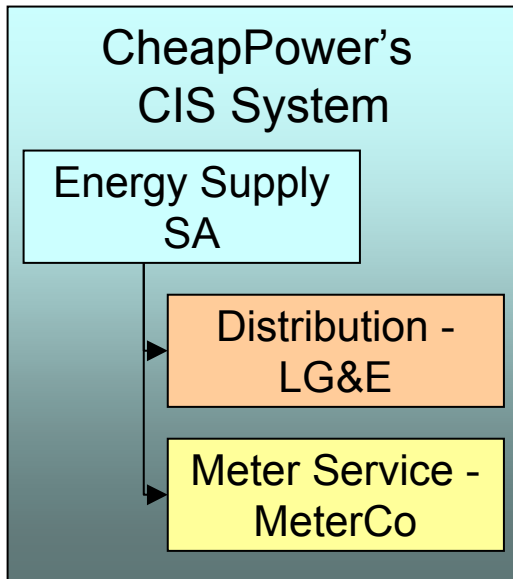
Consolidated billing. The following is an example of a bill that consolidates charges from many service providers. Rather than receive a consolidated bill, it is possible for the customer to receive 3 separate bills, one from each service provider (we refer to this as Dual billing).

Customer's Electric Bill	
Energy Supply - Cheap Power	\$12
Distribution - LG&E	\$9
Meter Service - MeterCo	\$2

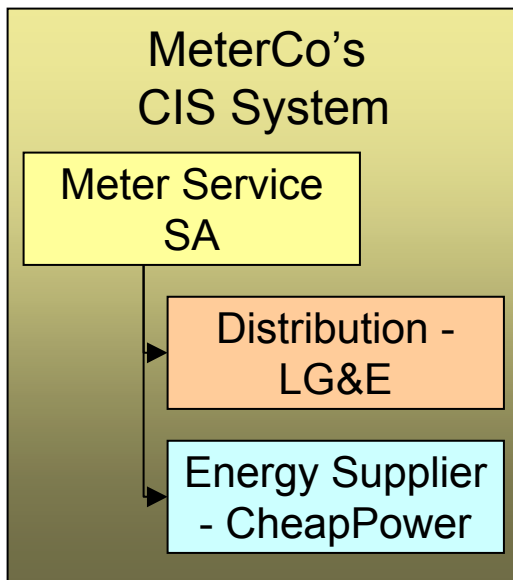
If we were to look at this customer's service agreement in the distribution company's CIS system, we'd find a service agreement for distribution charges, and linked to it would be information about the meter service and energy service providers:



If we look at this customer's service agreement in the energy supply company's CIS system, we'd find a service agreement for energy charges, and linked to it would be information about the distribution and meter service providers:



And finally, if we look at this customer's service agreement in the meter service company's CIS system, we'd find a service agreement for meter service charges, and linked to it would be information about the distribution and energy providers:



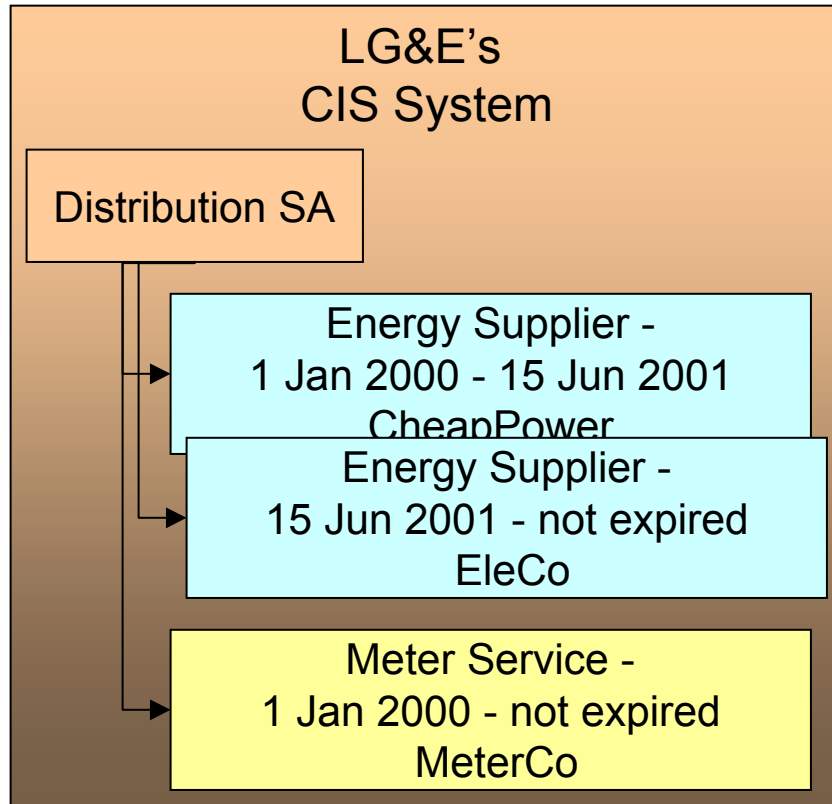
Bottom line. A customer's service providers keep track of the customer and each other in their respective CIS systems. A customer will have a service agreement (or the equivalent) in each service provider's CIS system. A customer's service agreement defines the service providers who supply each type of service.

Because information about customers and their services needs to be kept up-to-date in many different CIS systems, there is a need for automated communications between service providers. Refer to [Service Providers Have To Communicate About Customers](#) for more information.

Service Providers May Change Over Time

A customer typically has a choice of service providers. Over time, their choice may change. The system keeps track of a customer's service providers throughout time so that it can accurately cancel / rebill historical bills. This means that the service provider relationship is *effective-dated*.

The following diagram illustrates how a distribution company's system keeps track of a customer's service providers (notice the customer changed energy suppliers):



How To Set Up SA Relationships On A Customer's Service Agreement

There are three ways to set up a customer's service providers:

- **Manually.** An operator can manually change a service agreement's SA relationships. The manual method is NOT recommended as changing service providers typically involves many events (e.g., you have to notify the current service provider that they will be dropped). We strongly recommend having an operator kick off a workflow process and let the workflow process notify the service providers and make the desired changes. Refer to [Defining Workflow and Notification Options](#) for more information.
- **Use Default Service Providers.** The system will default a service provider on a required SA relationship when a service agreement is activated. Refer to [Defaulting Relationship Types And Defaulting Service Providers](#) for more information.
- **Workflow Processing.** A workflow process may contain workflow events that change a service agreement's SA relationships. Refer to [Defining Workflow and Notification Options](#) for more information.

When Your Company Is A Service Provider

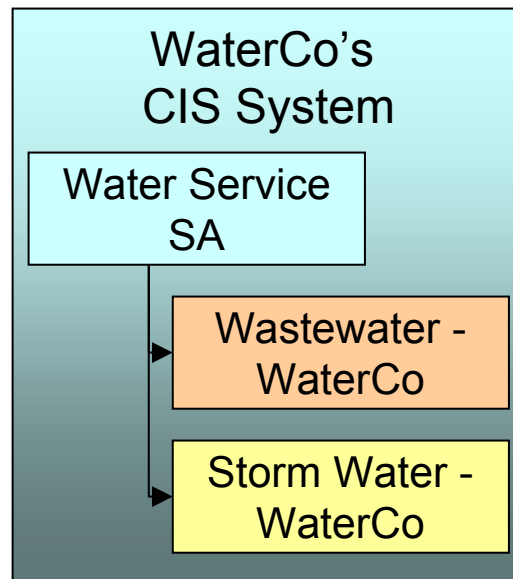
Besides setting up service providers for other companies that provide services to your customer, you may have to set up a service provider for your own company. You have to do this when:

- **Your organization can supply one of the services.** Refer to the illustration in [Service Providers Are Linked To Service Agreements](#). If you are LG&E and you supply energy in addition to distribution, you would need to set up a service provider for your own organization. Why? Because whenever you have a subcategory of service (e.g., energy supply), you must indicate the service provider who provides this service; even when it's you.
- **You decide to break up a service into subcategories** (and have a separate service agreement for each category). For example, a water company may choose to break up service charges into water, wastewater and storm water (they may do this because there are different rates for each category of service). The following is an example of the segregated charges associated with this water company's service charges.

Customer's Water Bill	
Water Service - WaterCo	\$15
Wastewater - WaterCo	\$6
Storm Water - WaterCo	\$5

This water company system may benefit by creating a single service (for water) and indicating there are subcategories of service (for wastewater and storm water). Whenever you have a subcategory of service, you must indicate the service provider who provides this service. And, in this example, the water company would be the sole service provider for each subcategory of service.

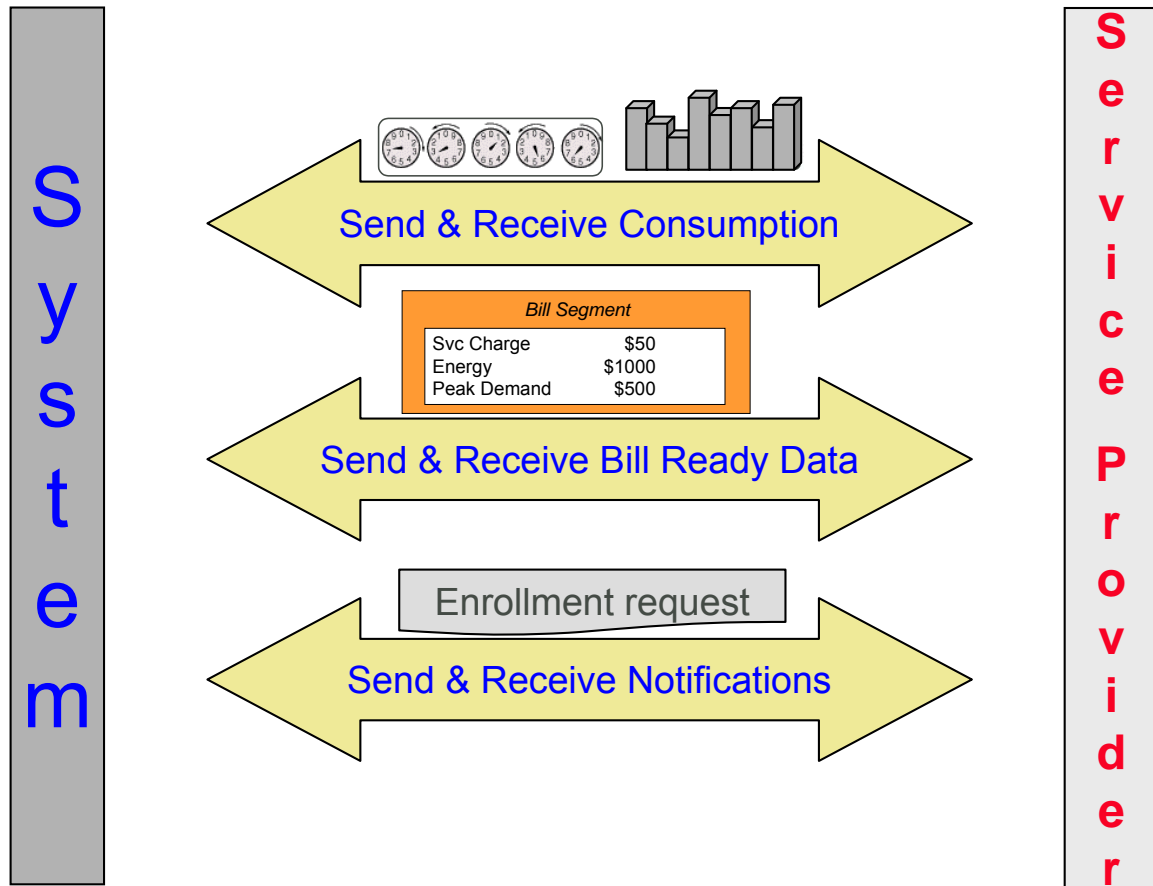
Customer's Water Bill	
Water Service - WaterCo	\$15
Wastewater - WaterCo	\$6
Storm Water - WaterCo	\$5



Note. Refer to [We Bill For Them – Rate Ready Bill Segments Are Special](#) for restrictions in respect of using subcategories of service.

Service Providers Have To Communicate About Customers

The providers of service typically have to communicate with each other in respect of the customer's service. The following diagram illustrates the major interfaces of information between your system and your service providers.



Depending on where your organization fits in the service provider hierarchy, you may:

- Bill for other service providers (or they may bill for you). Refer to [Billing Relationships](#) for more information about billing communications.
- Send the customer's consumption to service providers (or they may send it to you). Refer to [Consumption Relationships](#) for more information about consumption communications.
- Apprise service providers of the changes to the customers' service (or they may apprise you). Refer to [How Do You Communicate With Service Providers?](#) for more information about communications between service providers.

Relationships Between Service Providers

Service providers may arrange contractual relationships with other service providers to provide additional services, e.g., an energy service provider may work with a specific meter data management agency to gather and report interval meter read data.

The system does nothing special to enforce or record these inter-relationships. Why? Because most service provider switches are received by notification records. Notification records indicate all the associated service providers. There is no need for the system to maintain the inter-service provider associations.

A Service Agreement Can Have Many Types Of Relationships

As described in the previous sections, a given service can be subdivided into subcategories. Each type of service can have zero, one or more subcategories. We call each subcategory a **SA Relationship Type**. The topics in this section provide information about SA relationship types.

Contents

- [An Example](#)

- [Valid Relationship Types and Service Providers Are Defined On SA Types](#)

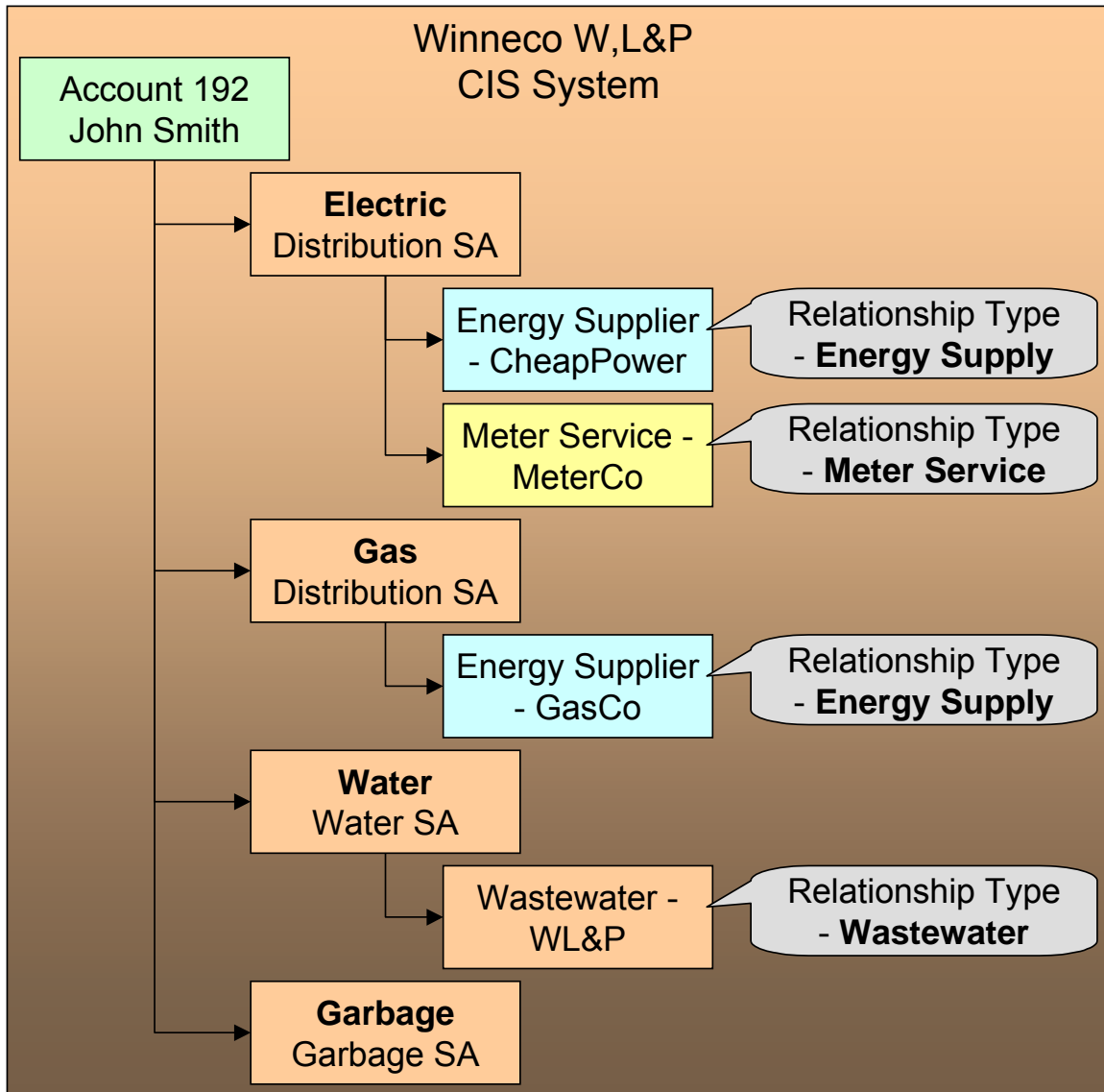
- [Defaulting Relationship Types And Defaulting Service Providers](#)

- [Required Relationship Types and Billing](#)

- [Relationship Types Do Not Impact Start / Stop](#)

An Example

We'll use the following example of a customer in a municipal utility's CIS system to explain SA Relationship types:



Note the following:

- Electric service has 2 SA relationship types: energy supply and meter service.
- Gas service has 1 SA relationship type: energy supply.
- Water service has 1 SA relationship type: waste water.
- Garbage service has no SA relationship types.

Bottom line. Service providers are related to the customer via “your” service agreement. Each service provider linked to a service agreement is defined in respect of a SA Relationship Type. This relationship is effective-dated because we care about how it changes over time.

Your relationship type is implied. The SA relationship type of “your” service agreement is implied, e.g., if you are a distribution company, “your” service agreement’s implied SA relationship type is “distribution”.

Valid Relationship Types and Service Providers Are Defined On SA Types

You control which services have SA relationships (and which don't) when you set up your SA types. Each SA type can have zero, one or more SA relationship types. Each relationship type, in turn, can have one or more valid service providers.

Refer to [Setting Up SA Relationships For SA Types](#) for the page used to define the service providers and SA relationship types for each SA type.

Defaulting Relationship Types And Defaulting Service Providers

Please be aware of the following:

- A SA relationship type for a given SA type can be marked as being **Required**.
- A service provider for a given SA relationship type / SA type can be marked as being the **Default**.

If, at activation time, the customer's master SA is missing a **Required** SA relationship that has a **Default** service provider, the activation process automatically creates the SA relationship type and links to it the **Default** service provider. If a master service agreement doesn't have all **Required** SA relationships, the service agreement cannot be activated. This is handy when your organization is the default service provider for a relationship type.

Refer to [Automatic Creation of Sub SAs](#) for information about how the system will automatically create sub service agreements for the defaulted SA relationship / service provider if your organization provides billing services for the service provider. Refer to [Setting Up SA Relationships For SA Types](#) for the page used to define the service providers and SA relationship types for each SA type.

Required Relationship Types and Billing

When the system attempts to create a bill segment for a service agreement whose SA type has **Required** relationship types, it checks if all such relationships are defined for the service agreement. If not, a bill segment error will be generated.

The reason this restriction exists is to handle the situation when your required relationship types change over time. For example, assume on your first day of production you only need energy suppliers defined on electric service agreements. After several months, gas deregulates. When this happens, you will need to change your control tables to indicate that your gas SA types require an energy supplier. If you don't write a default program to update your existing gas service agreements, billing will complain.

Relationship Types Do Not Impact Start / Stop

Customer service representatives (CSR's) are typically not involved with the customer's choice of service providers. Most organizations hear about a customer's service providers from the service providers or from a central body. This means that the start / stop dialog is not impacted by SA relationships. This also means that CSR's are not impacted by SA relationships (unless something goes wrong). If something goes wrong, the CSR's may need to manually correct SA relationships. Refer to [How To Set Up Service Providers On A Customer's Service Agreement](#) for more information.

Billing Relationships

When you set up a service provider, you must define your organization's billing relationship with the service provider. The following points provide examples of the billing relationships supported in the system,

- If you are an energy supply company, you may provide billing services for the distribution company. This means that your bill contains both your charges and the distribution company's charges. We refer to this as the **We Bill For Them** billing relationship.
- Alternatively, the distribution company may provide billing services for you. This means that the distribution company's bill contains your charges and their charges. We refer to this as the **They Bill For Us** billing relationship.
- Alternatively, you may both send bills to the customer. We refer to this as the **Dual Billing** relationship.
- Alternatively, if you subcategorize your services OR if your company provides one of the services that is provided by your service providers, then the system will create a separate bill segment for the subcategory of service. We refer to this as the **It's Us** billing relationship.

If you provide billing services for another service provider (i.e., you bill for them), there are two ways to determine the service provider's charges:

- You can load the service provider's rates in your system and calculate the charges for the service provider. We call this the **Rate Ready** calculation method.
- You can let the service provider calculate their own charges and interface them to you. We call this the **Bill Ready** calculation method.

If a service provider provides billing services for you (i.e., they bill for you):

- If the service provider has a suitable CIS system, they can load your rates in their system and calculate your charges for you. We call this the **Rate Ready** calculation method.
- You can calculate your charges and interface them to them. We call this the **Bill Ready** calculation method.

If you don't have a billing relationship with a service provider, you still need a service provider record to define such. Why? Because the system needs to know that it doesn't have to worry about a particular service provider in respect of billing. In addition, you may have other interactions with a service provider that have nothing to do with billing, e.g., you may send or receive consumption.

The topics in this section provide a wealth of information about the various billing relationships and the ramifications of each.

Contents

[Sub Service Agreements](#)
[We Bill For Them](#)
[Service Providers Have Service Agreements Too](#)
[They Bill For Us](#)
[Combinations Of Service Provider Billing Methods](#)
[Different Service Providers Can Bill Different Services](#)
[If You Deal With TBFU Service Providers, You Cannot Reopen Bills](#)

Sub Service Agreements

If you provide billing services for another service provider OR if you subcategorize your own charges, there will be a separate service agreement linked to the customer's account that holds these unique charges. We refer to this new service agreement as a **Sub SA**. We use the term "sub" because this service agreement is subservient to the "master" service agreement. By subservient we mean:

- A sub SA's start and stop dates are the same as the "master" service agreement. This statement may seem odd to you, but it's true – all sub SAs linked to a master service agreement have the same start and stop dates as does the master.
- A sub SAs status (i.e., **pending start**, **active**, **pending stop**, **stopped**, etc.) is controlled by its master service agreement.
- Refer to [The Rate Ready Calculation Method](#) for additional examples of subservience.

Note. You may find it helpful to keep in mind that sub SAs are only used for service providers with a billing relationship of **It's Us** and **We Bill For Them**. This is because these are the only relationships that have implicit billing responsibilities.

The topics in this section provide additional information about sub SAs.

Contents

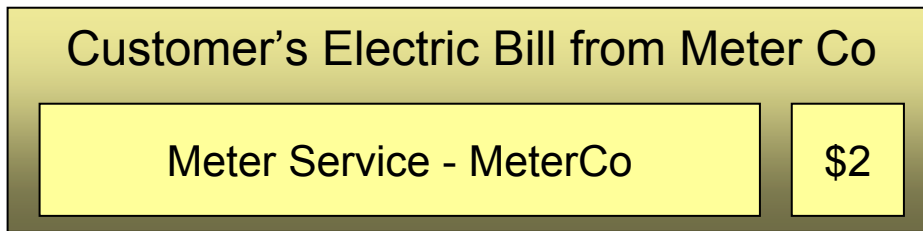
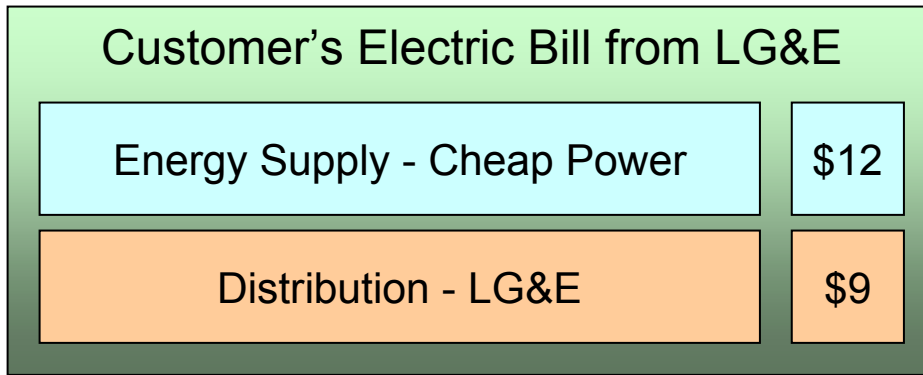
[Only Some Service Providers Have Sub SAs](#)
[Automatic Creation of Sub SAs](#)
[Sub SA State Transition](#)

Only Some Service Providers Have Sub SAs

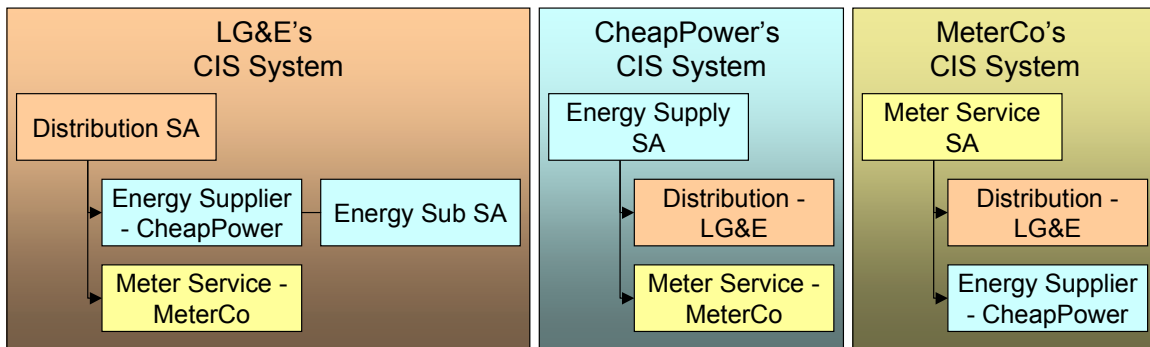
If you do not provide billing services for a service provider, there will be no sub SAs associated with the service provider's SA relationships. Let's use an example to make the point; assume:

- Service is divided into distribution, energy and meter service charges.
- You are the distribution company and you provide billing services for the energy supply company.
- The meter service company bills the customer independently.

In this situation, the customer will receive two bills: one from you (LG&E), the other from MeterCo. Notice that your bill (LG&E) contains your distribution charges AND CheapPower's energy charges:



If we were to look at the customer information in the three service providers' respective CIS systems, we'd find the following service agreements and SA relationships for the customer:



Notice the following:

- The customer has a service agreement in each supplier's CIS system.
- Because LG&E (the distribution company) bills for CheapPower, all customers who have their energy supplied by CheapPower will have a sub SA in LG&E's system. This sub SA maintains the charges (and receivable balance) associated with CheapPower's energy charges.
- Notice that neither CheapPower's nor MeterCo's CIS systems use sub SAs. This is because neither company bills for other service providers.

Sub SAs are needed if you subcategorize your charges. The above example shows sub SAs being used when a company provides billing service for another company. Sub SAs are also used when you subcategorize your charges – each sub SA contains the rate associated with each subcategory.

Automatic Creation of Sub SAs

The system creates sub SAs for customers choosing service providers where the billing option is **It's Us** or **We Bill For Them**. The system creates sub SAs via the following mechanisms:

- The [analyze SA relationships](#) background process (known by the batch control ID of **ANLYZSAR**) monitors newly activated SA relationships. If the respective service provider is **It's Us** or **We Bill For Them**, this process creates the sub SA(s) using the information defined on [SA Type SA Relationship Type - Sub SA Type](#).

Note. The [analyze SA relationships](#) background process also activates and stops sub SAs. Refer to [Sub SA State Transition](#) for the details.

- A button exists on the [SA Relationships](#) page. This button, when pressed, creates sub SAs real-time. This button would only be used if the operator couldn't wait for the background process to run.

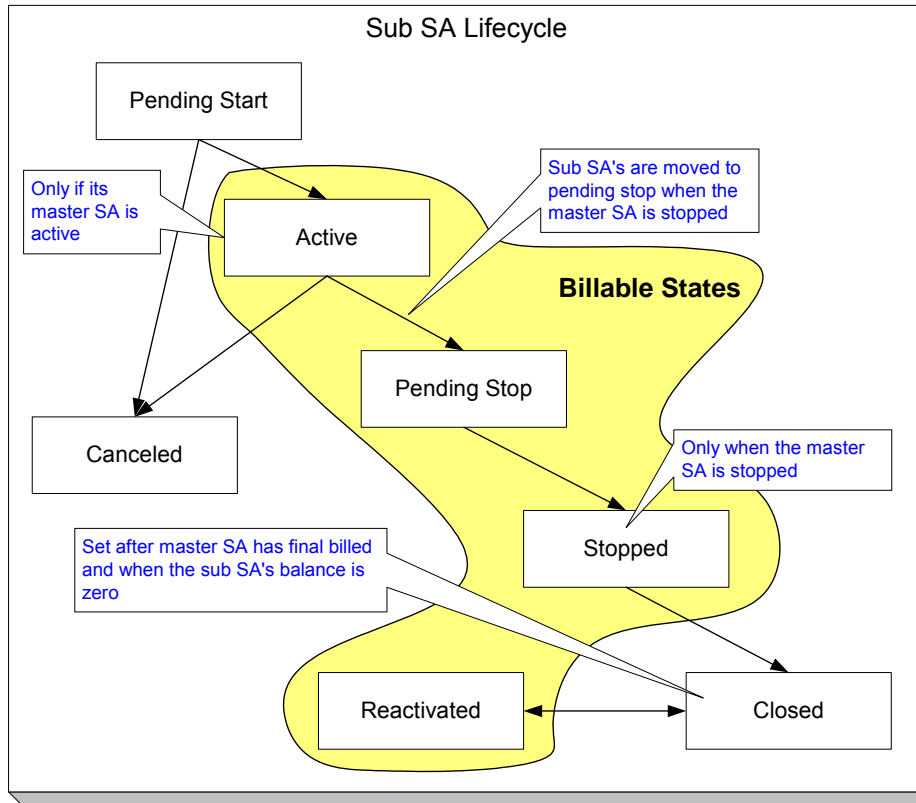
The SA type associated with the new sub SAs is defined when you set up each SA Type / SA Relationship Type. Refer to [Setting Up SA Relationships For SA Types](#) for more information.

If the SA type uses start options, the start option contains additional values (e.g., rates, contract riders) that are used to populate the newly created sub SAs. Note: start options are typically not used for **We Bill For Them – Bill Ready** service provider because we don't need to default rates, contract riders and/or contract values on billable charge sub SAs. Refer to [Setting Up SA Relationships For SA Types](#) for more information.

Manually created sub SAs. In very unusual situations, an operator may create a sub SA manually. An operator would do this using the [SA Relationship](#) transaction.

Sub SA State Transition

Sub SAs follow normal SA state transition rules, but have a few additional rules about when the transitions can take place.



The following points highlight the additional state transition rules:

- A sub SA can only become **Active** if its master SA is **Active**. Most sub SAs are activated by the [analyze SA relationships](#) background process. However, a user can manually activate a sub SA using the activate button on the [SA maintenance](#) page.
- Sub SAs become **Pending Stop** when their master becomes **Stopped**. This typically occurs when the [SA activation background process](#) stops the master SA. However, this can happen real time if a user manually stops a master SA using the stop button on the [SA maintenance](#) page. Note, contrary to normal service agreements, a sub SA cannot return to the **Active** state (because its master can never go back to **Active** or **Pending Stop**).
- A sub SA can only become **Stopped** if its master SA is **Stopped**. Most sub SAs are stopped by the [analyze SA relationships](#) background process. However, a user can manually activate a sub SA using the stop button on the [SA maintenance](#) page.
- A sub SA becomes **Closed** if its master has been final billed and it has a balance of zero. Most sub SAs are closed when their balance becomes zero after the master SA is final billed. However, a user can manually close a sub SA using the close button on the [SA maintenance](#) page. Note, if additional billable charges are interfaced after the master has been closed, the sub SA will be **Reactivated**.

We Bill For Them

If you provide billing services for another service provider, then you have a **We Bill For Them** billing relationship with the service provider. The topics in this section provide information about this type of billing relationship.

Contents

[Sub SAs Are Used When We Bill For Them](#)
[The Rate Ready Calculation Method](#)
[The Bill Ready Calculation Method](#)
[Pay At Bill Time vs. Pay At Pay Time](#)
[Paying The Service Provider - Technical Implementation](#)

Sub SAs Are Used When We Bill For Them

As described under [Sub Service Agreements](#), sub SAs are used for **We Bill For Them** service providers. The sub SAs hold the service providers charges. It's important to note that the system allows more than one sub SA to be created for a given service provider.

The Rate Ready Calculation Method

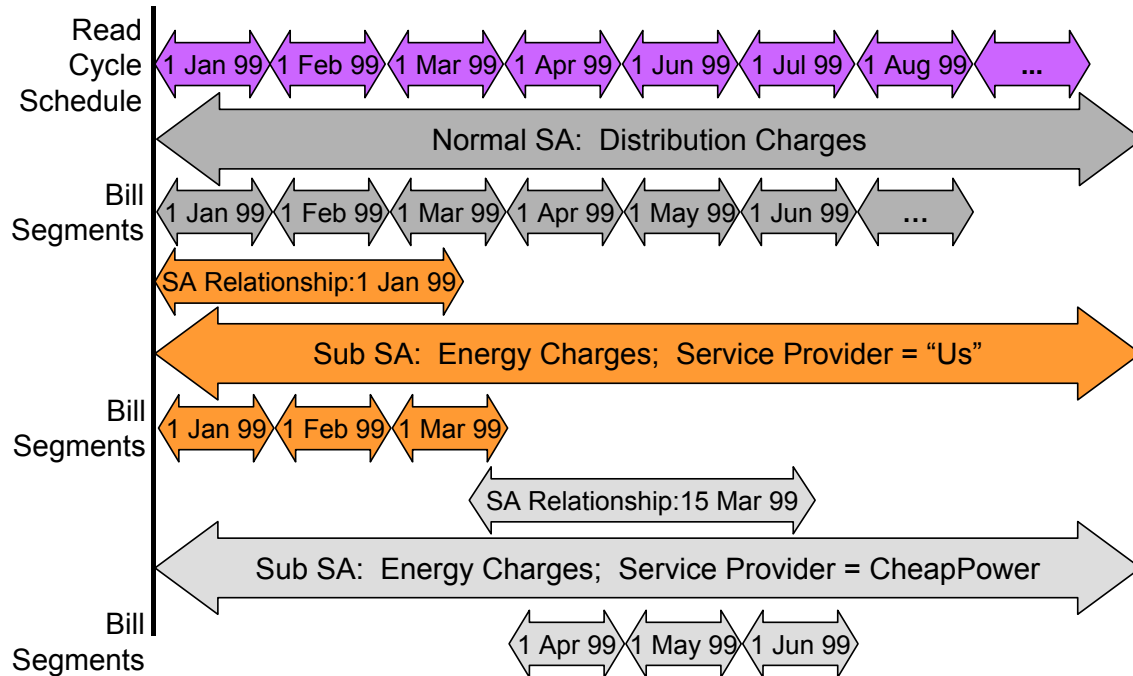
If you provide billing services for a service provider (i.e., the service provider has a billing relationship of **We Bill For Them**), you can load the service provider's rates in your system and calculate the charges for the service provider. We call this the **Rate Ready** calculation method.

It's Us can be **Rate Ready**. The prior paragraph indicated that the **Rate Ready** billing option was used for **We Bill For Them** service providers. If you subcategorize your services and use rates to calculate each subcategory's charges, the service provider(s) set up for your own organization will also be subject to the Rate Ready rules described below.

If a customer uses a **We Bill For Them – Rate Ready** service provider, the system will create a separate bill segment for the service provider's charges. Keep in mind the following in respect of these types of bill segments:

- The consumption is copied from the “master” service agreement (note: we copy item details and read details after register rule application). This means that the bill period on the sub SA's bill segment is identical to that on the normal SA. This also means that if the “master” SA is in error, bill segments for the sub SA will not be created.
- The rate defined on the sub SA is applied against this consumption. All register and SQ rules defined on the service provider's rate will be used to manipulate this consumption. If you have register rules on your master SA's rate to apply a line loss or convert a cubic foot to a therm, you don't have to have these register rules on the sub SA's rate (because we copy the consumption from the master to the sub after register rule application).

The following illustration should help:



Be aware of the following:

- This example is from a distribution company's perspective.
- From 1 Jan 99 through 15 Mar 99, the distribution company distributes power AND supplies energy (i.e., the distribution company is the service provider of energy).
- On 15 Mar 99, CheapPower becomes the energy supplier. However, the next bill period ends on 30 Mar 99. This means that CheapPower will only start supplying energy on 1 Apr 99. See the following note for why the effective date of the supplier switch is not the true effective date from billing's perspective.

Service Provider Changes Take Affect On The Next Bill. If the service provider is changed within a customer's billing period, the system assumes that the service provider in affect at the START of the period is effective the entire billing period. This means that a change of service providers will only take affect on the bill whose start date follows the change date.

Please be aware of the following characteristics of **Rate Ready** bill segments:

- You cannot cancel a **Rate Ready** bill segment independently from the master bill segment. If you need to cancel / rebill the sub SA's bill segment, you must cancel / rebill the master SA's bill segment.
- You cannot delete a **Rate Ready** bill segment independently from the master bill segment. If you need to delete the sub SA's bill segment, you must delete the master SA's bill segment.
- Similarly, if you cancel / rebill the normal SA's bill segment, all **Rate Ready** bill segments will be cancelled / rebilled.

Bottom line. Creating, freezing, deleting, and canceling a “master” bill segment does the same with all **Rate Ready** bill segments. You cannot create, freeze, delete or cancel a **Rate Ready** bill segment independent of its master bill segment.

The Bill Ready Calculation Method

If a customer uses a **We Bill For Them – Bill Ready** service provider, the service provider calculates their own charges and interfaces them to the system. We then present their charges on our bill; we don’t actually calculate anything (sometimes this is referred to as “pass through” billing).

The sub SA linked to a **We Bill For Them – Bill Ready** must be a billable charge SA. Why? Because billable charge SAs exist to hold bill lines until such time as a bill is created for the customer’s account. Refer to [Setting Up Billable Charge Templates](#) for more information.

Please be aware of the following characteristics of **Bill Ready** bill segments:

- Unlike **Rate Ready** bill segments, **Bill Ready** bill segments can span different time periods than the master SA. This is because you cannot predict when a service provider will interface their billable charges to you. In fact, a given bill could contain billable charges that span different periods and these charges could have been interfaced from historic and existing service providers.
- Unlike **Rate Ready** bill segments, **Bill Ready** bill segments can be created and deleted independently from the master bill segment.
- Unlike **Rate Ready** bill segments, **Bill Ready** bill segments can be cancelled independently from the master bill segment.

Contents

[Sending Consumption And Waiting For The Charges](#)
[Uploading Consumption \(Rather Than Uploading Calculated Charges\)](#)
[Calculating Taxes On Uploaded Charges](#)

[Sending Consumption And Waiting For The Charges](#)

If your organization supports **We Bill For Them – Bill Ready** service providers AND you are the source of consumption used by these service provider to calculate their charges, you need to be aware of the following :

- We do NOT recommend sending raw meter reads to service providers. Rather, we recommend sending these service providers the same consumption that you use on your bill segments. Remember, the system maintains a snapshot of billed consumption on bill segments associated with service agreements that are linked to service points.
- But to implement our recommendation (of only interfacing billed consumption to service providers), we need to create a bill segment for the master SA and then wait until the service provider returns the billable charge before sending out the bill. The following points describe how this works:
 - Early in the bill cycle, the system creates a bill segment for the master SA (remember, the system maintains a snapshot of billed consumption on bill segments linked to service points). When the master bill segment is frozen, the system interfaces the snapshot consumption to all service providers associated with the master service agreement who need consumption (this is defined on the service provider). Refer to [We Can Send Billed Consumption To Any Service Provider](#) for more information about interfacing consumption to service providers.

- The bill associated with the bill segment will not be completed. Why? Because the bill segments associated with the **We Bill For Them – Bill Ready** service providers will be in **Error**. This only happens if you use the **Billable Charge** bill segment creation algorithm (this algorithm is plugged in on the sub SA's SA type). On the algorithm, make sure to specify a value of **Y** for the parameter **Wait For The Last Day Of The Bill Cycle**. By doing this, a bill segment in the **Error** state will exist for the sub SA until the last night of the bill cycle's window.
- On the last night of the bill cycle window, when the system attempts to create a bill segment for the billable charge SA associated with the **We Bill For Them – Bill Ready**, it will either find recently interfaced billable charges or it won't. If it finds unbilled, billable charges, a bill segment will be created for the sub SA and the billable charges will be swept onto it. If it doesn't find unbilled billable charges, the bill will be completed (i.e., sent out) without the service provider's charges.

Bottom line. If a customer's account uses **We Bill For Them – Bill Ready** service providers, the bill will not be completed (i.e., sent out) until the last night of the bill cycle (if you use the appropriate algorithm). Why? Because we wait until the last night of the bill cycle before trying to sweep on recently interfaced billable charges. If no billable charges have been interfaced from the service provider by the last night of the bill cycle, the bill will be sent out without the service provider's charges.

Batch versus Online Bill Creation. If you create a bill online, the system will NOT create an **Error** bill segment for the **We Bill For Them – Bill Ready** service provider. Why? Because if you want to create an online bill, either the service provider has interfaced their charges or they haven't. If they have, they should be swept on the bill (via the creation of a bill segment). If they haven't, it shouldn't prevent you from completing the bill.

Uploading Consumption (Rather Than Uploading Calculated Charges)

We understand this is confusing, but it is quite possible to set up the system so that the **We Bill For Them – Bill Ready** service provider passes in CONSUMPTION rather than the calculated bill lines. They would only do this if they are not able to calculate the charges in their system and have therefore provided you with their rates. To do this, you would set up everything as described above. In addition, when you upload the billable charges, you must specify the consumption to be rated in the billable charge's service quantity (SQ) collection.

Calculating Taxes On Uploaded Charges

It is possible to set up the system to calculate taxes for billable charges. You would do this if the service providers are passing through the charges and want you to calculate the taxes.

If you want taxes calculated on top of billable charges:

- Create a service quantity (SQ) on the billable charge that contains the total monetary amount that taxes will be calculated on. Note, you would not have to do this if you have an SQ rule in your rate that calculates the total monetary amount to which taxes should be applied.
- Specify a rate on the billable charge service agreement. This rate will contain rate components that calculate taxes. Note, the rate components will be simple SQ rate components that apply a percentage to the value of the SQ that represents the monetary amount on which taxes should be calculated.

When the system creates the bill segment for the billable charge, it will call the rate and the rate will calculate the taxes and add additional lines (actually, bill calculation headers) to the bill segment.

Pay At Bill Time vs. Pay At Pay Time

If you provide billing services for another service provider (i.e., the service provider has a billing relationship of **We Bill For Them**), you will owe them money because you will be receiving money from their customers for their service. You have two options in respect of when the system increases the amount you owe the service provider:

- You can tell the system to increase how much you owe the service provider when you freeze the customer's bill segment. Some people refer to this method as "purchasing the receivable from the service provider". We call this the **Pay At Bill Time** method.
- You can tell the system to increase how much you owe the service provider only when you are paid for by the customer. We call this the **Pay At Pay Time** method.

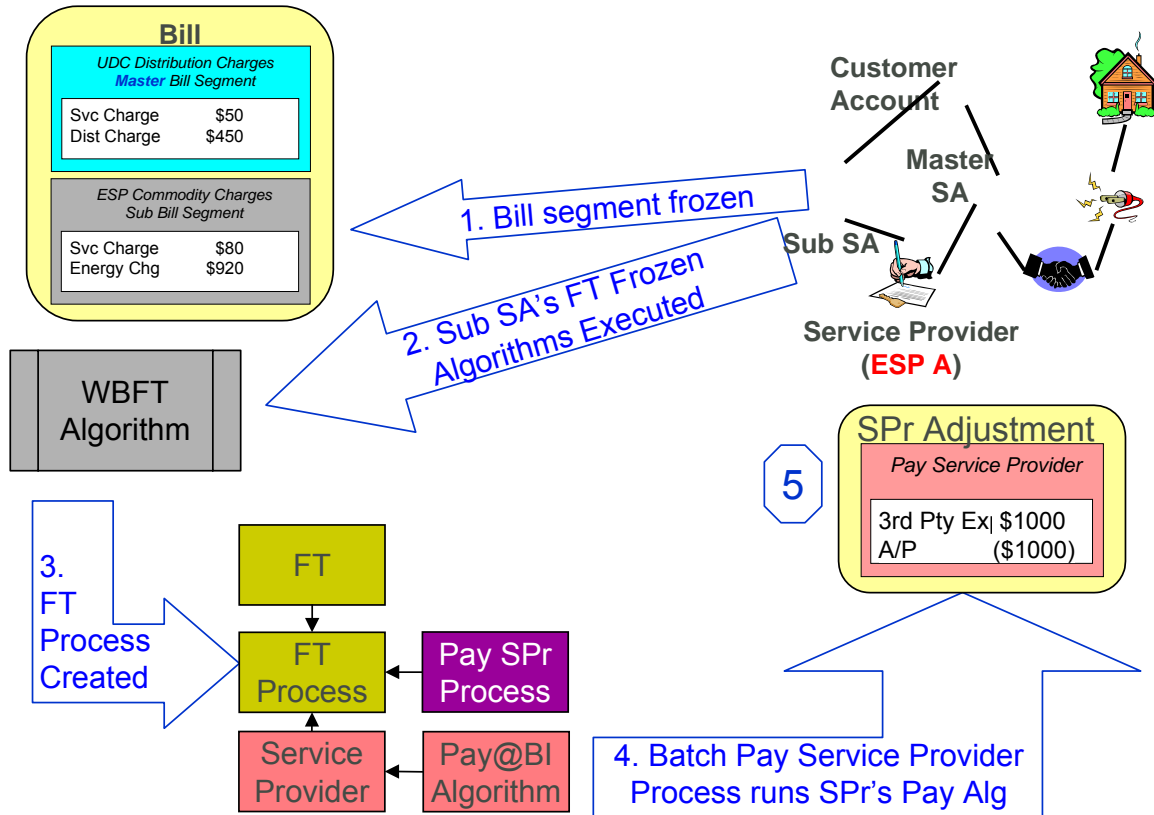
The method used for a service provider is defined on the service provider's record.

Refer to [When We Bill For Them, We Owe Them Money](#) for more information.

Paying The Service Provider - Technical Implementation

Technical section! This section describes, technically, how we increase the amount we owe a **We Bill For Them** service provider. If you aren't technically inclined, skip this section.

The following illustration shows the logical steps involved with increasing how much we owe a **We Bill For Them** service provider.



The following points explain the steps:

- When a financial transaction (FT) is frozen, the system executes the FT Freeze algorithms defined on the SA(s) SA type.
- If you've set up the system properly (i.e., you've put the appropriate FT Freeze algorithm on the sub SA's SA type), one of these algorithms will determine if there is a WBFT service provider associated with the sub SA. If so, it will insert a row on the FT Process table.
- Rows on the FT process table are used as "triggers" for batch processes. In this case, the batch process that is triggered is the one that looks at new FT's and determines if a related "payable" adjustment should be created for the **We Bill For Them** service provider. This batch process uses the service provider's Payment Relationship and Pay Service Provider algorithm to determine when and how to create these "payable" adjustments.

Service Providers Have Service Agreements Too

Most service providers need service agreements as explained in the following topics.

Contents

When We Bill For Them, We Owe Them Money
Adjustments and We Bill For Them Service Providers
When They Bill For Us, They Owe Us Money

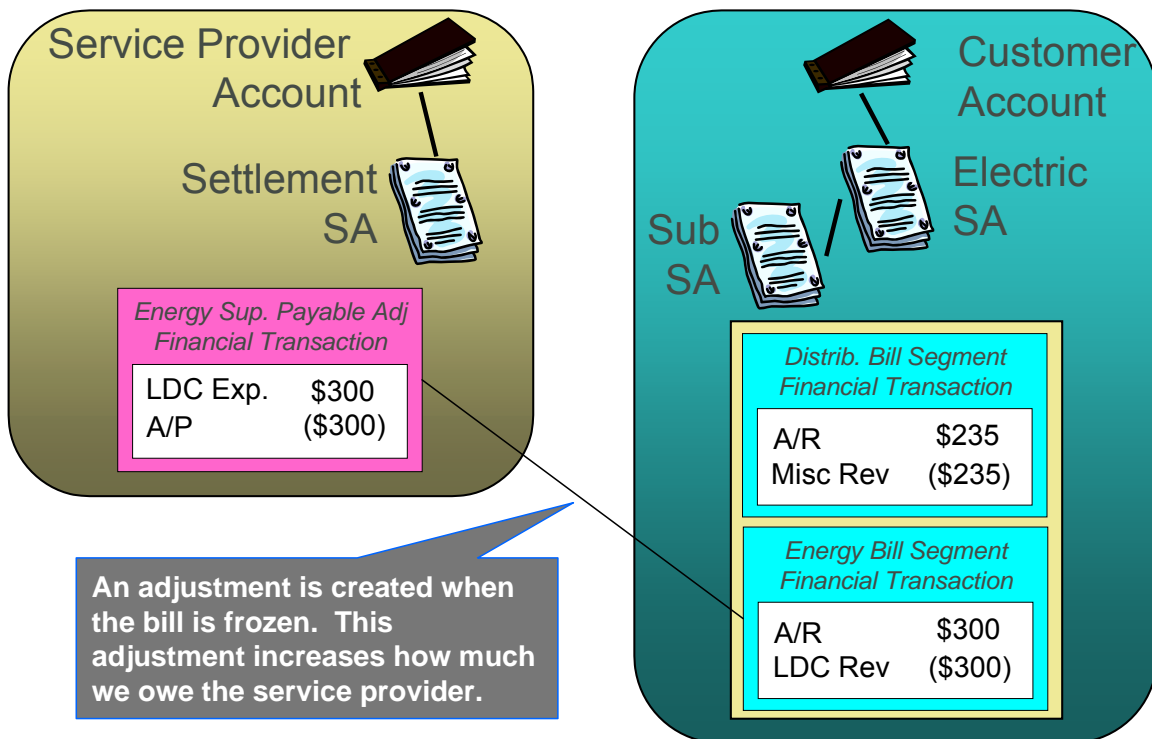
When We Bill For Them, We Owe Them Money

When you bill on behalf of a service provider (i.e., the service provider has a billing relationship of **We Bill For Them**), you will eventually owe them money (because the customer pays you for the service provider's service). You have two options in respect of when the system increases the amount you owe the service provider:

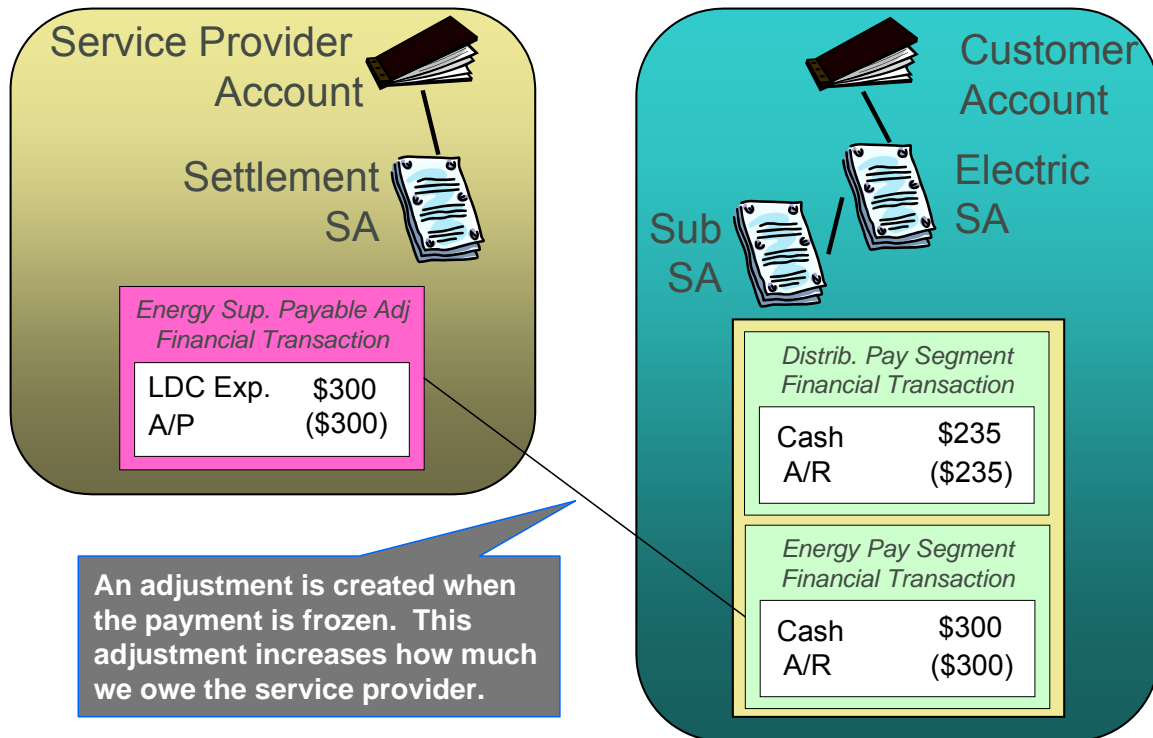
- You can tell the system to increase how much you owe the service provider when you create the customer's bill. Some people refer to this method as "purchasing the receivable from the service provider".
- You can tell the system to increase how much you owe the service provider only when you are paid for by the customer.

The system keeps track of how much you owe a service provider on a service agreement linked to the service provider's account. The system creates adjustments against this service agreement to increase how much you owe them.

If you "purchase the receivable" (i.e., you owe them when you bill the customer), an adjustment is created when the customer is billed. If you owe them only when you are paid by the customer, an adjustment is created when the customer pays. The following example illustrates an adjustment being created when the bill is frozen (illustrating the "purchase the receivable" scenario):



If you only pay the service provider when you are paid, the example would look as follows:



Adjustments and We Bill For Them Service Providers

It's important that you are comfortable with the information described under [When We Bill For Them, We Owe Them Money](#) before reading this section.

Adjustments associated with **We Bill For Them** sub service agreements are tricky. The following points describe how the system “pays” the related service provider when adjustments are issued against the customer’s sub service agreement:

- For **Pay At Bill Time** service providers, most adjustments are treated just like bill segments, i.e., when the adjustment’s FT is frozen, a payable adjustment is created for the respective service provider. The reason “most” is underlined in the previous sentence is because A/P adjustments (i.e., adjustments used to interface check requests to your A/P system) are excluded. Why? Because A/P adjustments are used to refund overpayments to the customer. Overpayments are purely between the customer and your company (you never transferred the overpayment to the service provider because it’s associated with a **Pay At Bill Time** service provider).
- For **Pay At Pay Time** service providers, A/P adjustments are treated just like payment segments, i.e., when the adjustment’s FT is frozen, a payable adjustment is created for the respective service provider. All other types of adjustments are ignored. Why? Because A/P adjustments are used to refund overpayments to the customer. Think of it like this - when the customer originally overpaid, you transferred this overpayment to the **Pay At Pay Time** service provider; therefore, when you refund the overpayment, you get to take the money back from the service provider.

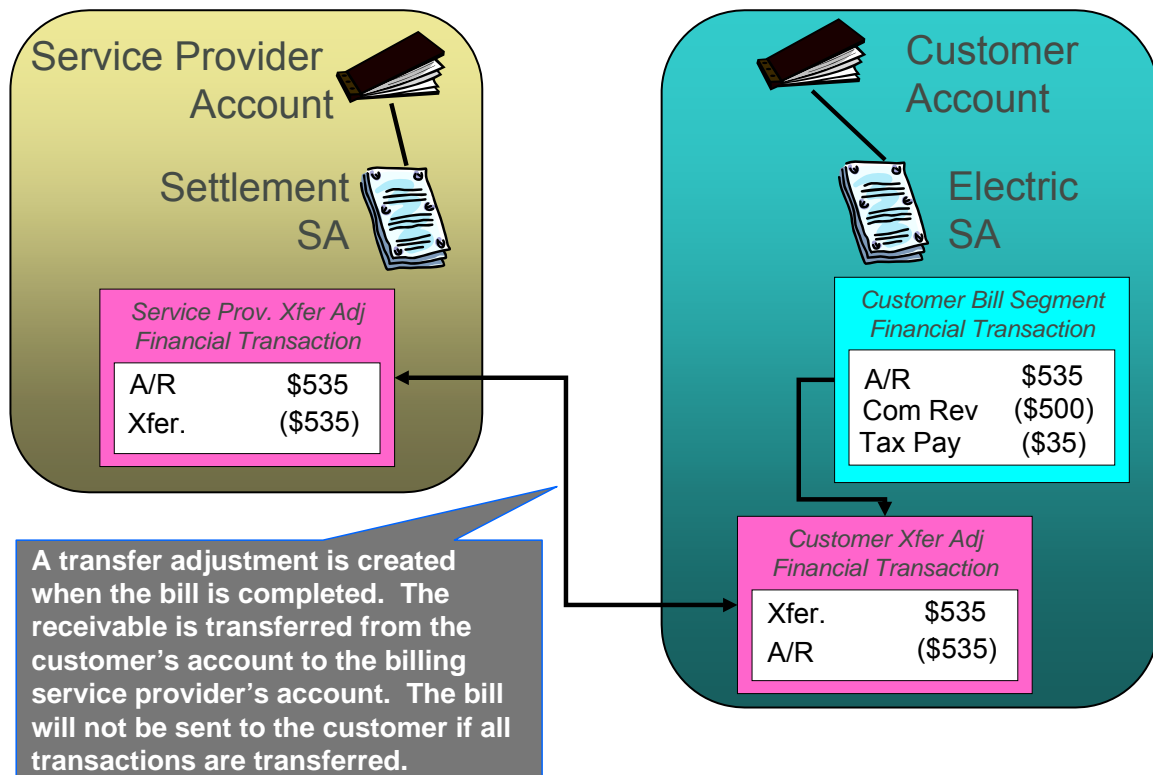
When They Bill For Us, They Owe Us Money

When a service provider bills on behalf of your organization, they will eventually owe you money (because the customer pays them for your service).

The system keeps track of how much a service provider owes you on a service agreement linked to the service provider's account. The question is, How does the system determine how much you are owed when you don't produce a bill? Well, you do produce a bill, it just doesn't get sent to the customer. We understand this is a little confusing, but think about it like this:

- The service provider is presenting your charges on their bill.
- You still have to calculate how much the customer owes your organization otherwise you'll never know how much you are owed by the service provider.

As illustrated below, when a bill is completed, the system determines if there are bill segments and/or adjustments associated with service agreements with a service provider who "bills for us". If it finds these, it transfers the receivable from the customer's service agreement to the service provider's service agreement. If all financial transactions have been transferred to the service provider, no bill is produced for the customer.



Bottom line. We always generate a bill for "us", even though we don't send it to the customer.

They Bill For Us

If a service provider provides billing services for you, then you have a **They Bill For Us** billing relationship with the service provider. The topics in this section provide information about this type of billing relationship.

Contents

[The Customer Still Needs A Service Agreement](#)
[They Bill For Us - Bill Ready](#)
[They Bill For Us - Rate Ready](#)
[A/R Transfer - Technical Implementation](#)
[Routing Billable Charges To Service Providers - Technical Implementation](#)
[Bill Routings Are Changed](#)

The Customer Still Needs A Service Agreement

If a service provider bills for you, you still need a service agreement for the customer. Why? Because:

- As explained under [Service Providers Are Linked To Service Agreements](#), service providers are defined in respect of a customer's service agreement (therefore the customer must have a service agreement).
- As explained under [When They Bill For Us, They Owe Us Money](#), you still have to calculate bills for the customer.

A customer's bill history still exists. Be aware that even when a service provider bills for us, you will still be able to see the customer's billing history. It's just that the customer won't owe you anything because the receivable balance will be transferred to the service provider's account.

They Bill For Us - Bill Ready

At bill completion time, the system determines if there are bill segments and/or adjustments associated with service agreements with a service provider who "bills for us". If it finds these,

- It transfers the receivable from the customer's service agreement to the service provider's service agreement. If all financial transactions have been transferred to the service provider, no bill is produced for the customer.

Note that payments and A/P adjustments are not transferred. Why? Because payments and A/P adjustments are purely between the customer and your company.

- Each bill segment and adjustment is marked to be interfaced to the service provider (via a separate background process). Refer to [The System Background Processes](#) for more information about this background process.

They Bill For Us - Rate Ready

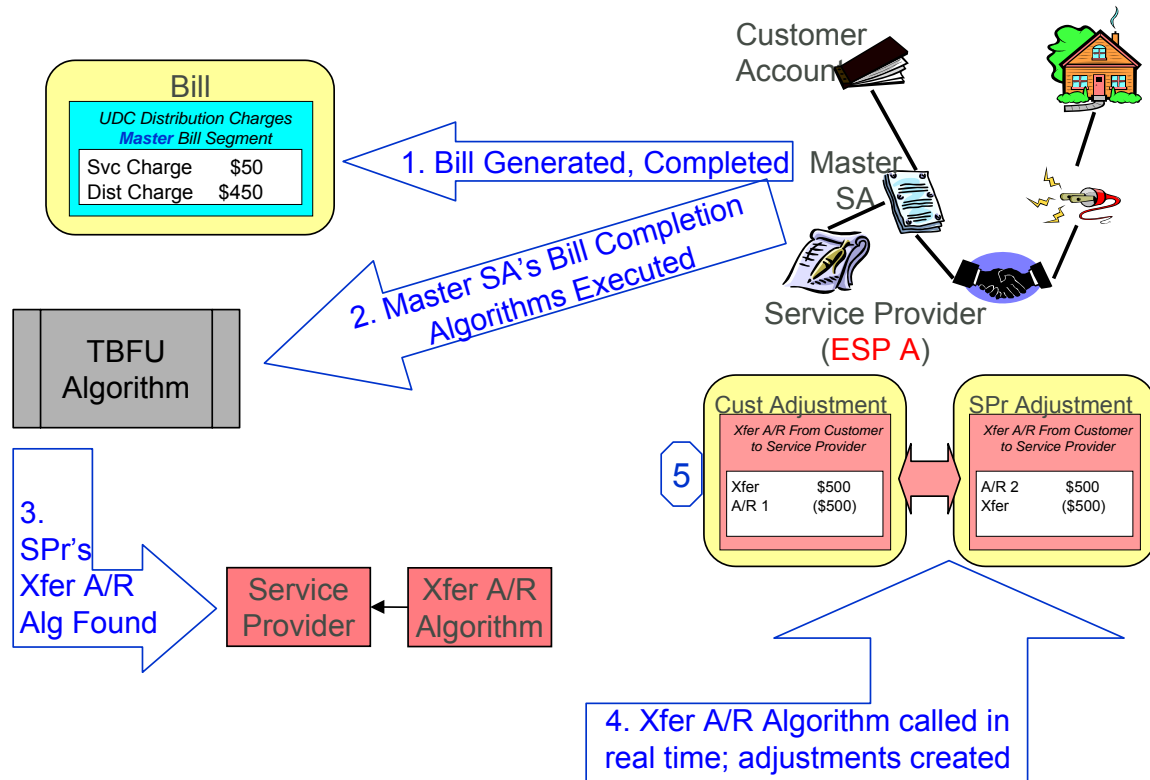
The ***They Bill For Us – Rate Ready*** option is not a recommended option. Why? Because you really have to compute how much the customer owes as explained above. If you go to the trouble of figuring out how much the customer owes, then it makes sense to interface this to the billing service provider.

The only difference between this option and ***They Bill For Us – Bill Ready*** is that the system will not interface the bill segments and adjustments to the billing service provider.

A/R Transfer - Technical Implementation

Technical section! This section describes, technically, how a customer's A/R balance is transferred to a ***They Bill For Us*** service provider. If you aren't technically inclined, skip this section.

The following illustration shows the logical steps involved with the transference of a customer's A/R balance to a **They Bill For Us** service provider.



The following points explain the steps:

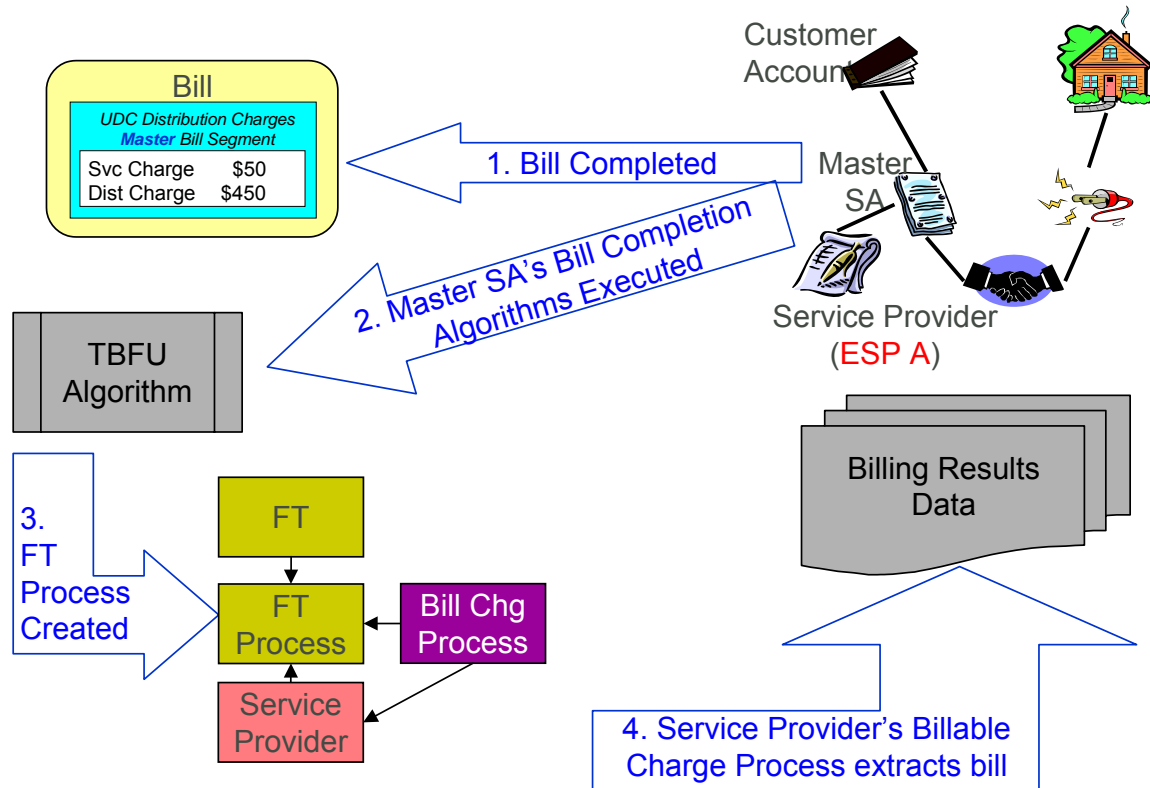
- When a bill is completed, the system executes the bill completion algorithms defined on the bill's master SA(s) SA types.
- If you've set up the system properly (i.e., you've put the appropriate Bill Completion algorithm on the master SA's SA type), one of these algorithms will determine if there is a **They Bill For Us** service provider associated with each master SA on the bill. If so, it will execute the Transfer A/R algorithm defined on the service provider's record. This algorithm causes a transfer adjustment to be created (transferring the financial transaction's affect on the customer's balance from the customer to the service provider).

Note. If there are multiple master SAs on a bill, the financial details associated with each respective master SA could be transferred to different service providers (e.g., one service provider could receive the financial details for gas and another for electricity). Refer to [Different Service Providers Can Bill Different Services](#) for more information.

Routing Billable Charges To Service Providers - Technical Implementation

Technical section! This section describes, technically, how we send billable charges to service providers. If you aren't technically inclined, skip this section.

The following illustration shows the logical steps involved with sending billable charges to service providers.



The following points explain the steps:

- When a bill is completed, the system executes the bill completion algorithms defined on the bill's master SA(s) SA types.
- If you've set up the system properly (i.e., you've put the appropriate Bill Completion algorithm on the master SA's SA type), one of these algorithms will determine if there is a **They Bill For Us** service provider associated with each master SA on the bill. If so, it will insert a row on the FT Process table.
- Rows on the FT process table are used as "triggers" for batch processes. In this case, the batch process that is triggered is the one that downloads billable charges to the service provider. The ID of the batch process that is referenced on the trigger comes from the Service Provider's Billable Charge Download Process.

Note. If there are multiple master SAs on a bill, the financial details associated with each respective master SA could be routed to different service providers (e.g., one service provider could receive the financial details for gas and another for electricity). Refer to [Different Service Providers Can Bill Different Services](#) for more information.

Bill Routings Are Changed

If all of an account's "master" SAs have a SA relationship with a service provider who bills for us (i.e., the service provider's billing relationship is **They Bill For Us**), then we have nothing to send to the customer. The system still creates bill routings, but with a couple of differences:

- The Batch Process Id and Run Number are reset.

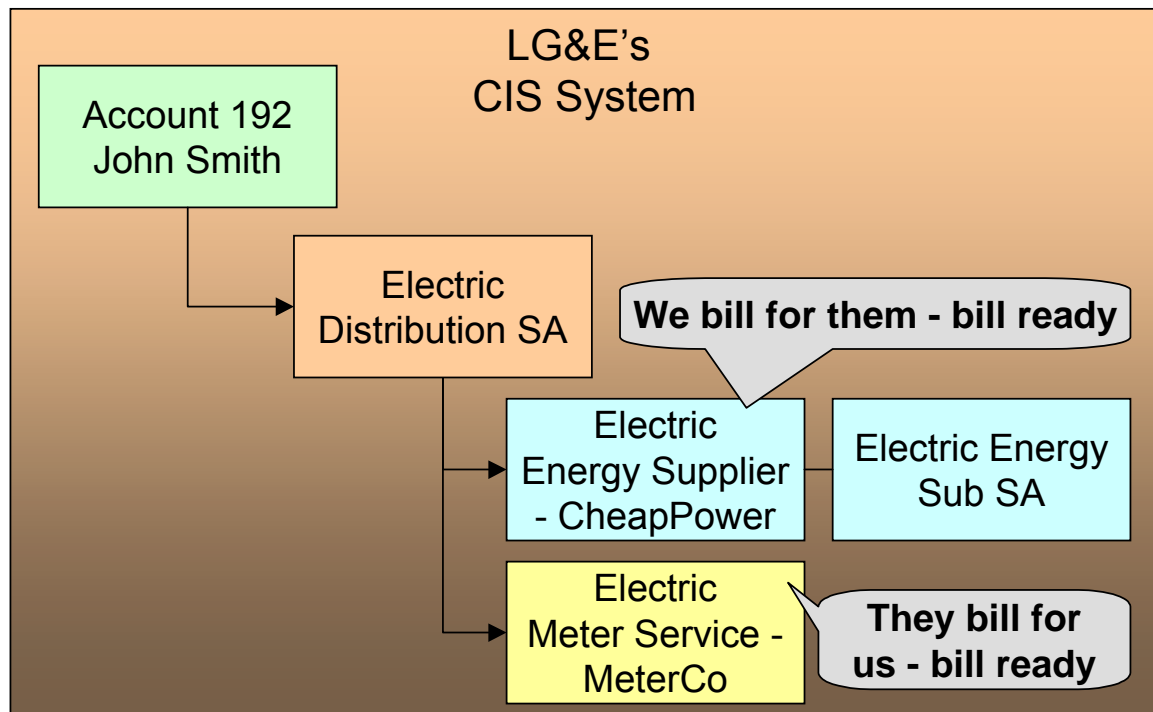
- The Customer's Name is prefixed with the text from a bill message code 6, 10103. This message code's text is ***** Bill not sent.**

This way, the operators can easily see that the bill was not routed and why.

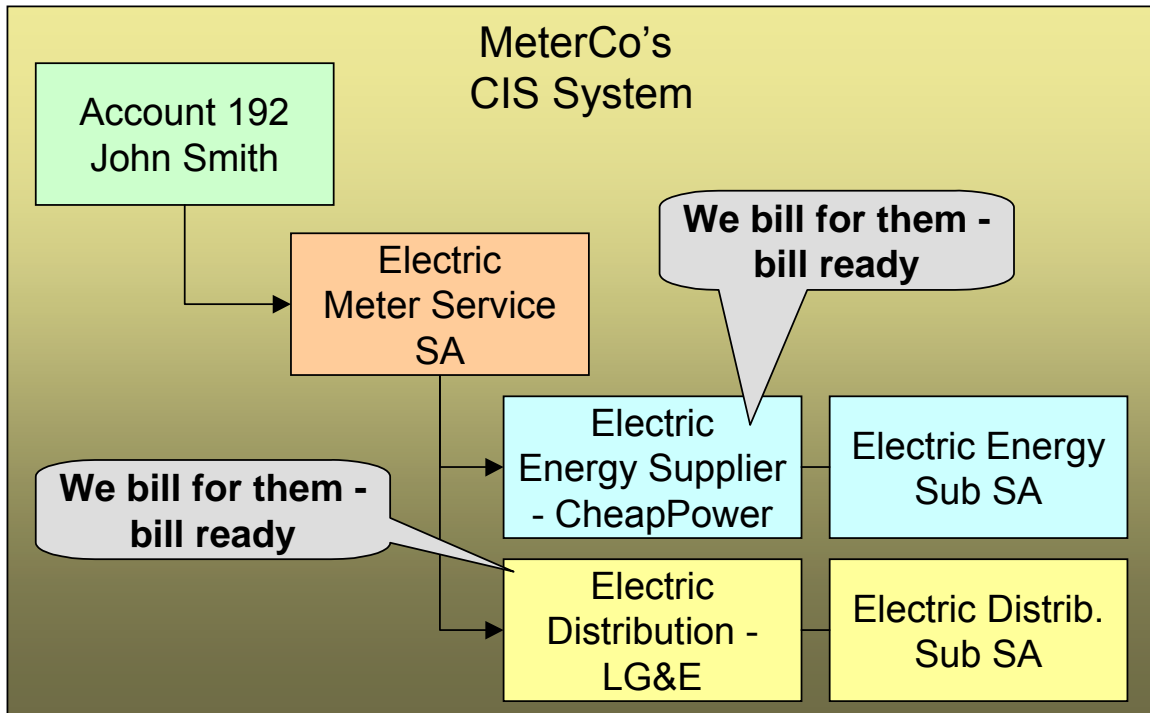
Refer to [Different Service Providers Can Bill Different Services](#) for information about how the various services under an account could be billed by different service providers.

Combinations Of Service Provider Billing Methods

Consider the following situation:



In the above example, we are billing for the energy service provider and the meter service provider is billing for us. This means that we will calculate the charges for ourselves, interface charges from CheapPower, and then interface our charges and CheapPower's charges to MeterCo. MeterCo will then produce a bill for the customer that contains our distribution charges, CheapPower's energy charges, and MeterCo's service charges. To help solidify this point, let's look at how this customer would look in MeterCo's CIS system.

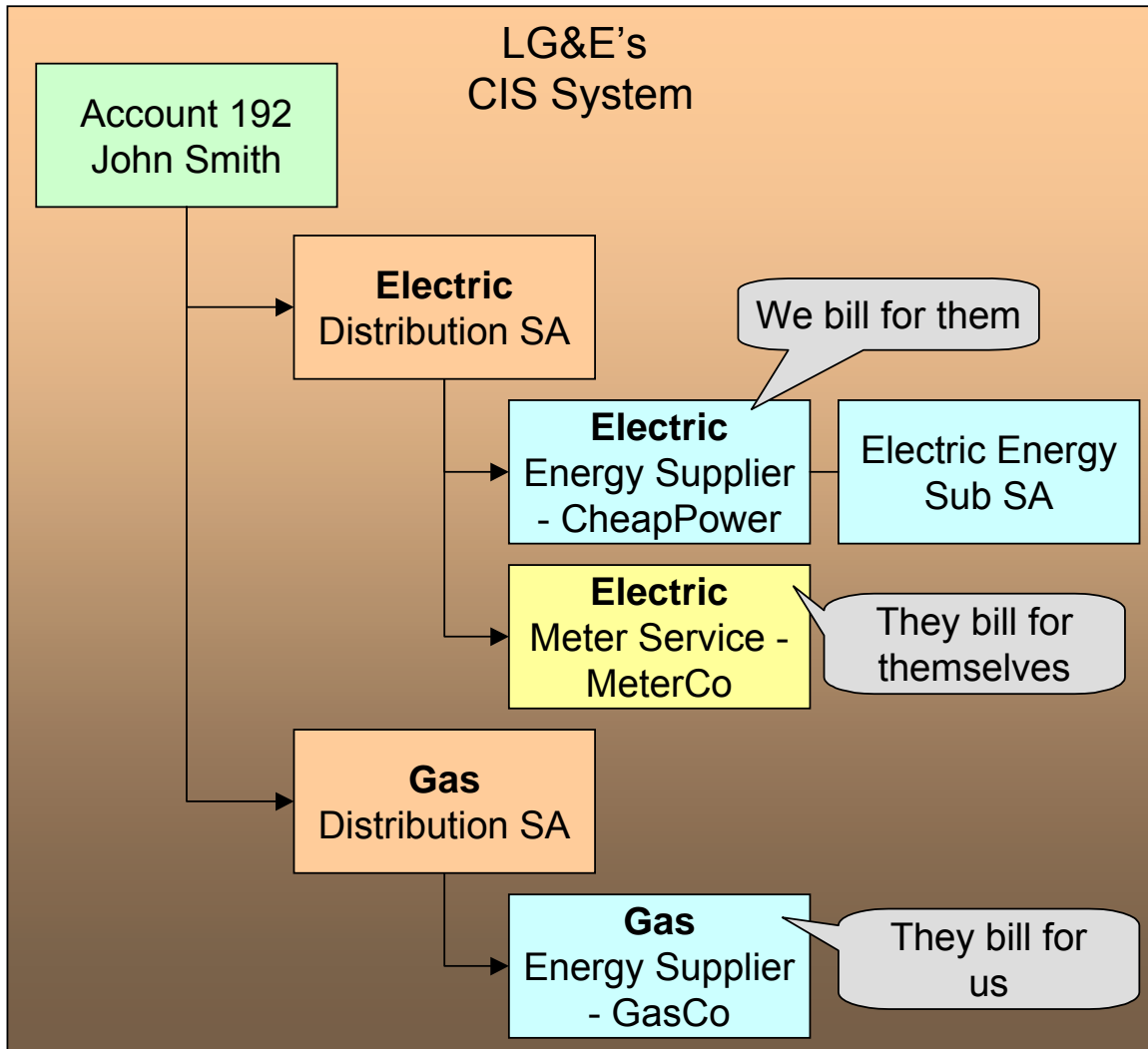


There are some restrictions in respect of permissible combinations of service providers that may supply service to a customer as described in the following points:

- If the system encounters a customer with a **We Bill For Them** (WBFT) – Bill Ready service provider and another service provider that is **They Bill For Us** (TBFU) – Rate Ready, a billing error will be produced. Why? Because TBFU – Rate Ready means they have everything they need to calculate our bills and therefore we do not interface bill lines to them. If we don't interface our bill lines to them, then we can't interface the charges that were interfaced from the WBFT – Bill Ready service provider. You may wonder why we don't prohibit WBFT – Rate Ready and TBFU – Rate Ready, because it's conceivable for the TBFU service provider to have our rate and the WBFT service provider's rate.
- If the system encounters a WBFT – Pay At Pay Time service provider and another service provider that is TBFU, a billing error will be produced. Why? Because when the system detects a TBFU service provider, it transfers the receivable from the customer to the service provider (and therefore the customer's account will never be paid).

Different Service Providers Can Bill Different Services

Be aware that the system determines billing relationships at the service agreement level, NOT at the account level. To make the point, check out the following customer in a distribution company's system:



Be aware of the following in respect of the above illustration:

- The distribution company (LG&E) distributes both electricity and gas.
- The customer has a choice of energy service providers for both gas and electricity.
- This customer – John Smith – purchases his electricity from CheapPower and his gas from GasCo.
- LG&E provides billing services for CheapPower.
- GasCo provides billing service for LG&E.

In this situation, LG&E will send bills to the customer that contain both electric distribution and energy charges (but no gas distribution charges). GasCo will also send bills to the customer; these will contain both LG&E's gas distribution charges as well as their own energy charges.

If You Deal With TBFU Service Providers, You Cannot Reopen Bills

Refer to [Bill Lifecycle](#) for information about reopening previously completed bills.

If your organization deals with **They Bill For Us** (TBFU) service providers, a great deal happens when a bill is completed (e.g., the receivable is transferred from the customer to the service provider, we may mark the bill segments and adjustments for routing to the service provider, etc.). These things cannot be undone and therefore the system will not let you reopen bills when these things have occurred.

Technical rule. The specific rule that prevents the reopening of bills is as follows: if a bill contains a service agreement whose SA type has one or more bill completion algorithms, the system will not allow the bill to be reopened. Refer to [SA Type - Algorithms](#) for more information about bill completion algorithms.

Consumption Relationships

When you set up a service provider, you must define your organization's consumption relationship with the service provider – a service provider may send the customers' consumption to you, you may send consumption to them, or you may have no consumption relationship with a given service provider.

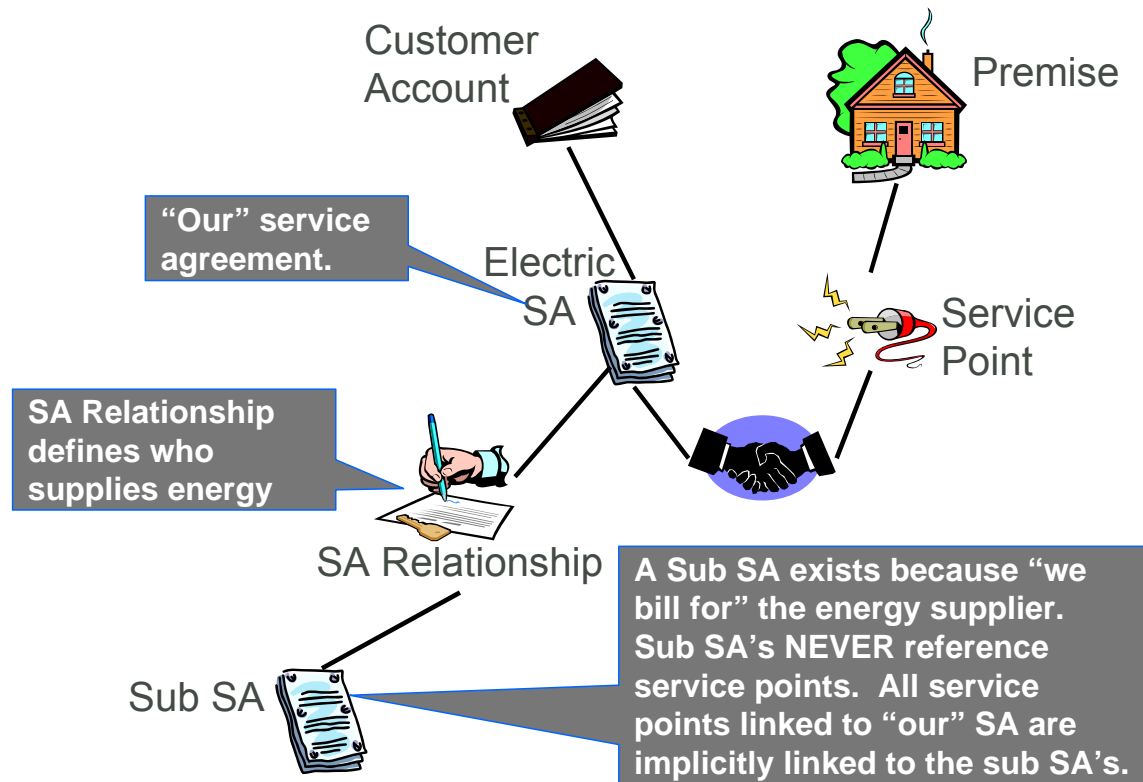
The topics in this section provide a wealth of information about the various consumption relationships and the ramifications of each.

Contents

- [Only The Master SA Is Linked To Service Points](#)
- [We Can Send Billed Consumption To Any Service Provider](#)
- [Routing Consumption To Service Providers - Technical Implementation](#)
- [We Can Receive Consumption From Service Providers](#)
- [MDMAs And Service Cycles](#)

Only The Master SA Is Linked To Service Points

The following diagram makes the point that service points cannot be linked to sub SAs (i.e., service agreements that exist to hold charges associated with **We Bill For Them** service providers). This is because all service points associated with our “master” SA are implicitly linked to all sub SAs.



It's important to understand why the system does not allow sub SAs to reference service points:

- The consumption associated with all service providers should be the same, otherwise the customer will receive inconsistent bills from different service providers.
- The easiest way to ensure consumption is the same for all service providers is to make sure that they all have the same service points (which they must if sub SAs "inherit" their service points from their master).

We Can Send Billed Consumption To Any Service Provider

You can send consumption to any service provider. It doesn't matter what their billing relationship is. Information on the service provider object tells the system if AND how to send consumption to a service provider.

Rather than send raw reads to service providers, we download consumption that has been calculated and snapshot onto the "master" bill segment. We send this billed consumption because:

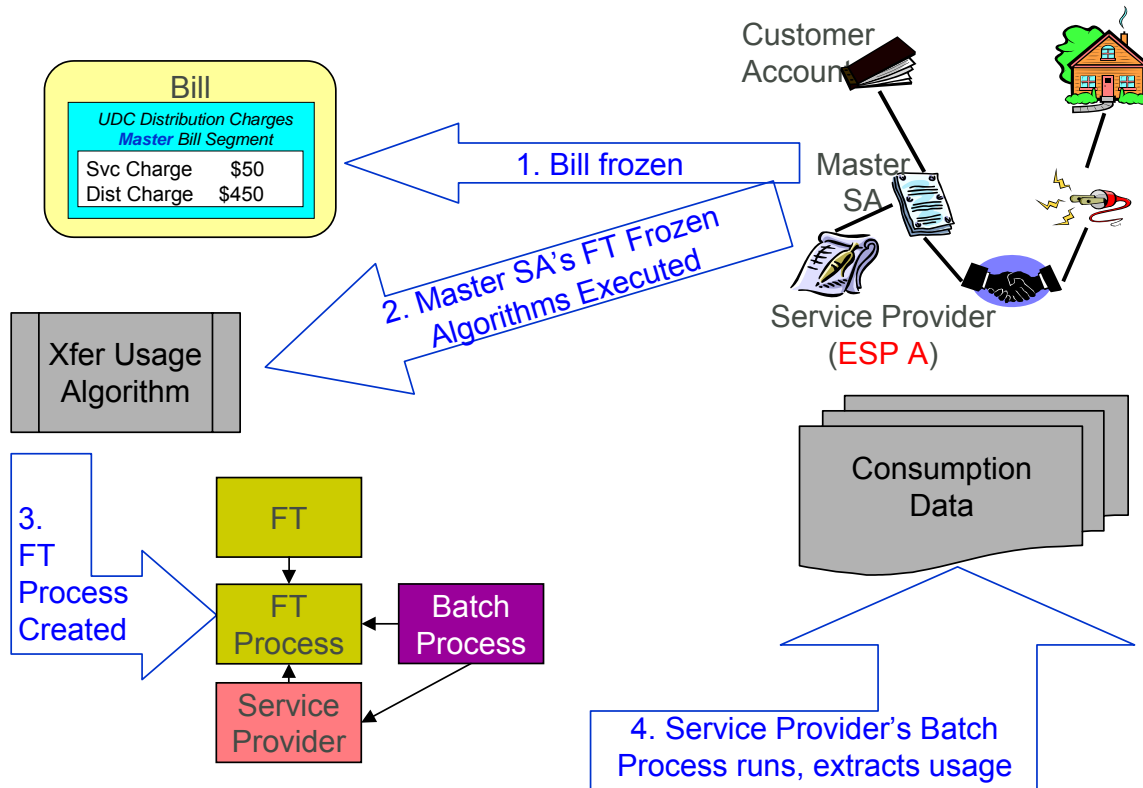
- It is clean and validated
- Register indexes have been subtracted
- Multiple registers have been summed

Note. We only send consumption after a bill segment is **Frozen**. If your organization supports **We Bill For Them – Bill Ready** service providers AND you are the source of consumption used by these service providers to calculate their charges, please refer to [The Bill Ready Calculation Method](#) for an explanation of how the system waits a given amount of time for the recipient of the consumption to interface their charges back to the system before sending the bill to the customer. In other words, the bill sent to the customer should contain the bill segment that triggered the consumption download as well as bill segments containing uploaded billable charges.

Routing Consumption To Service Providers - Technical Implementation

Technical section! This section describes, technically, how we send consumption to service providers. If you aren't technically inclined, skip this section.

The following illustration shows the logical steps involved with sending consumption to service providers.



The following points explain the steps:

- When a financial transaction (FT) is frozen, the system executes the FT Freeze algorithms defined on the SA(s) SA type.
- If you've set up the system properly (i.e., you've put the appropriate FT Freeze algorithm on the master SA's SA type), one of these algorithms will determine if there are service providers associated with the master SA who need consumption. If so, it will insert a row on the FT Process table.

- Rows on the FT process table are used as “triggers” for batch processes. In this case, the batch process that is triggered is the one that downloads billable consumption to the service provider. If multiple service providers need consumption, multiple rows will be inserted. The ID of the batch process that is referenced on the trigger comes from the Service Provider’s Consumption Download Process.

We Can Receive Consumption From Service Providers

We can receive consumption from any source. We use the standard meter read upload to interface consumption from service providers. Because we use the standard meter read upload, each customer must have:

- A premise
- A service point
- A meter with registers sufficient to hold the interfaced consumption

You don’t have to perfectly model the service points and meters. If you are not the distribution company, you may be worried about how to keep meter and service point information up-to-date. Be aware that you don’t have to model the SP’s and meters perfectly. Why? Because you just need to set up enough information so that consumption can be uploaded accurately. Let’s use an example of a customer with multiple meters and service points at a premise. Rather than model this perfectly, you could set up a single SP and link to it a single meter. All that matters is that the meter has the appropriate registers to hold the interfaced consumption.

When a customer’s consumption (i.e., meter reads) is uploaded from a service provider, the service provider must be defined as the “upload source” on the meter read upload staging records. The system uses this information to validate that the service provider is linked to the customer on the effective date of the read AND that they are defined as ***Sending Consumption To Us***. Refer to [Uploading Meter Reads](#) for more information.

MDMAs And Service Cycles

Warning! This section uses terminology and concepts described in The Cyclical Meter Read Process.

A Meter Data Management Agency (MDMA) is a service provider who reads meters. In some locales, a meter can be read by a variety of MDMAs, in other locales there is no MDMA as the meter is read by the distribution company. If your organization has MDMAs, then you need to be aware of the following:

- MDMAs must be linked to service agreements as a service provider. Like all service providers, they can change over time.
- When an MDMA is reading a service agreement’s meters, the MDMA may override the customer’s service cycle (and schedule) with their own cycle. If they do this, the customer’s service cycle in the MDMA’s system is defined on the service agreement’s SA relationship information for the MDMA. Note: whether or not a service provider can override a service cycle is controlled by a switch on the service provider’s record. Refer to [Service Provider – Main](#) for more information.

- A service provider's service cycle schedules are maintained using the system's normal service cycle schedule. The ID of the service provider associated with each schedule is defined on the service cycle; in other words, if a service provider reads meters and they can override the customer's service cycle, the service provider's service cycles must be defined in the system. Refer to [Designing Service Cycles, Routes, And Schedules](#) for more information.
- At billing time, the system determines if a service agreement is covered by an MDMA. If so, it uses the service cycle defined on the service agreement's MDMA SA relationship record.
- If your organization ever reads the customers' meters, your regular read cycle should be maintained on your service points. You can think of the service cycle that is defined on a service agreement's MDMA SA relationship record as an override of your service points normal service cycle.
- If the meter read download process detects that a service point is linked to a service agreement with an active MDMA, it still creates a meter read download staging record; however, it marks it as **Do Not Need To Read**. This means that when a service point is no longer read by an MDMA, the meter read will be downloaded normally.
- Refer to [We Can Receive Consumption From Service Providers](#) for a description of how a service provider interfaces consumption into the system.

Deposits Issues

Warning! This section uses terminology and concepts described in [The Big Picture Of Deposits](#).

Deposits should be held using normal deposit service agreements (SAs). You should NOT use the [Sub Service Agreements](#) (sub SA) functionality to hold or bill for deposits because deposit service agreements do not have the same state transition as do master SAs (e.g., you can activate or stop a deposit independent from its master).

However sub SAs can be covered by a deposit. If so, their SA type must reference a deposit class. To make the point, let's examine a few scenarios:

- Assume you have a sub SA for your own charges (this can happen when we use sub SAs to unbundle charges from the Master SA). In this case, it is likely that the sub SA and master SA will be in the same deposit class. This means that a single deposit SA would cover both the master and the sub SA.
- Assume **They Bill For Us** (Bill or Rate Ready). In this situation, we still have a master SA for our charges and we transfer the charges to the service provider who does the billing. In this case, it is likely that we would be holding the deposit for the service provider, not on the end-use customer. If we are in a situation where 1) we cannot hold a deposit against the service provider, and/or 2) we are not assured of the service provider paying us when the customer doesn't pay them, then we might want to put the master SA in a deposit class and hold a deposit against the customer's account using a normal deposit SA. We would not expect the service provider to bill the customer for the deposit, so we don't need a sub SA. We bill the customer directly for the deposit using our normal deposit SA.
- Assume **We Bill For Them** (Bill or Rate Ready). In this situation, we could hold a normal deposit SA for the customer's master SA. For sub SAs, we have two scenarios:
 - **We pay at billing time.** Since we purchase the receivable, we would want to increase our normal deposit to cover the Sub SA. To do this, the sub SA's SA type's deposit class should be the same as our master SA's deposit class.

- **We pay at payment time.** It seems unlikely that we would want to hold a deposit on behalf of a service provider when we don't purchase the receivable. However, it is possible to do so by putting the sub SA into its own deposit class. If you did this, the system will require a separate deposit SA for the service provider's deposit. The system would calculate and refund such deposits using the algorithms defined on the new deposit SA's SA type's deposit class. It's important to be aware that the deposit is not held with respect to the specific service provider. Rather, it is just held in the system as separate deposit that could be used for any service agreement that belongs to its deposit class.

Credit and Collection Issues

Warning! This section uses terminology and concepts described in [The Big Picture Of Credit & Collections \(C&C\)](#).

C&C is only tricky if you deal with **We Bill For Them** service providers. We'll run through the service provider billing relationships to explain why:

- If we have a **Dual** relationship with a service provider, we don't have their debt, so we only have a responsibility to tell them when we cut a customer (via a Notification). We don't have to worry about collecting their debt.
- If we have a **They Bill For Us** relationship with a service provider, there is no debt on the customer's SA because it gets transferred to the service provider (and the service provider's SA will fall into arrears if they don't pay us).
- If we have a **We Bill For Them** or **It's Us** relationship with a service provider, the customer's debt associated with the service provider's service is maintained on a sub SA (i.e., it is segregated from our debt). This segregation of debt is both a powerful feature and a cause of administrative difficulties. The topics in this section provide more information about this issues.

Contents

[Debt Class Recommendations](#)
[Severing Service](#)

Debt Class Recommendations

A service agreement's debt class is an important element in determining how a customer's debt is collected. In general, we recommend the following:

- If the service provider has a billing relationship of **It's Us**, we recommend the sub SAs belong to the same debt class as the "master". Why? Because both SAs' overdue debt should probably be grouped together under a single collection process.
- If you buy the receivable from the service provider (i.e., the service provider has a payment relationship of **Pay at Billing Time**), we recommend the sub SAs belong to the same debt class as the "master". Why? Because both SAs' overdue debt should probably be grouped together under a single collection process.
- If you don't buy the receivable from the service provider (i.e., the service provider has a payment relationship of **Pay at Pay Time**), you may want to use a different debt class on the sub SA. Why? Because you may collect the service provider's debt differently.

Bottom line. If both the “master” and the sub SAs fall into arrears, you will have 1 or 2 collection processes, it all depends on the debt class assigned to each SA type.

Refer to [Automating Your C&C Activities](#) for information describing how debt class plays a part in this processing.

Severing Service

Sub SAs and severance is tricky. Why?

- Because it's possible for the master SA to be in arrears when the sub SA isn't (for all the standard reasons - directed payments, cancel / rebills, etc.).
- Because it's possible for the sub SA to be on one collection process and the master to be on another (due to different debt classes or different time lines).

Both of these situations could result in severance starting for only one of the service agreements in the master / sub relationship. However, YOU CAN'T CUT SERVICE FOR ONE WITHOUT CUTTING THE OTHER because there is only one service point.

Before we describe how to deal with this conundrum, we'd like to remind you that the system starts a unique severance process for each SA (sub or normal) to be severed. It only creates a severance process for those service agreements linked to a collection process when the collection process' **Start Severance** event is activated. The type of severance process that is created is controlled by each service agreement's SA Type's severance criteria. Please keep in mind the following when designing these severance processes:

- [Only The “Master” Service Agreement Is Linked To Service Points](#). This means only master SAs should have a “cut for non payment” severance event. Note: typically, such a severance process will expire the “master SA” several days after the cut event if funds are NOT received.
- As described under [Sub SA State Transition](#), a sub SA becomes **Pending Stop** (and eventually **Stopped**) when its “master” is stopped. This means the sub SAs will be finaled when the master is finaled.
- If you start severance on a sub SA when the master isn't being severed, you have a problem because you can't cut the sub SA independent from the master SA.

We'll use an example to illustrate how you should design your severance processes to deal with the above challenge. Assume you have a master and a sub SA where both are being managed under the same collection process. Also assume that the **Start Severance** event kicks off on 18-Dec-1999. In this situation, we'd recommend the following severance processes to be kicked off.

Master SA Severance Process		
Trigger Dt.	Event Type	Status
18-Dec-99	Set door hanger	Pending
Pending	Cut for non pay	Pending
Pending	Expire SA	Pending

Sub SA Severance Process		
Trigger Dt.	Event Type	Status
3-Jan-00	Create To Do Entry	Pending

Notice that the sub SA's severance process contains a single event that generates a To Do Entry on a date in the future of the **Expire SA** event on the Master SA. This entry should be something like "sub being severed independent of its master". This event will only be triggered if the master SA is paid off and the sub SA isn't. Why? Because if the master SA's **Expire SA** event is executed, the Sub SA will be **Stopped** and stopping a SA cancels outstanding severance processes. If the sub gets paid, the system will cancel the sub's severance process.

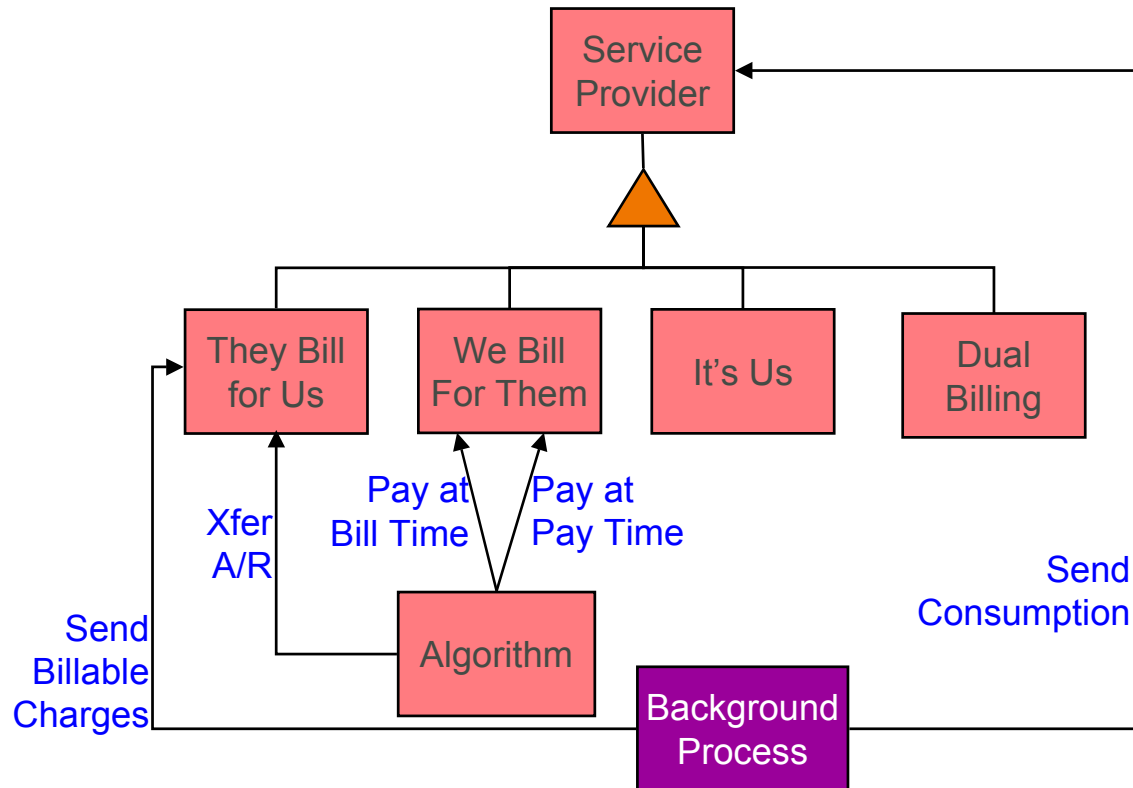
Let's change the example and assume that the master starts severance and the sub doesn't. In this situation, the master SA will eventually hit the **Expire SA** event and the sub SA will also stop. There's no alternative.

And let's change the example again and assume the sub starts severance and the master doesn't. In this situation, the To Do Entry will only be created X days after the start of severance. If you can't stand this date being X days in the future of the creation of the severance process, create an "Severance Criteria Algorithm" that checks if the master is not being severed or collected and generates a different severance process (with a different start date). Refer to [Designing Your Severance Procedures](#) for more information about Severance Criteria Algorithms.

An Object-Oriented Perspective Of Service Providers

Warning! Skip this section if you are not technically inclined.

The following [object diagram](#) illustrates in a concise format the various types of service providers and the plug in algorithms and processes available for each.



How Do You Communicate With Service Providers?

You communicate with service providers (and they communicate with you) using the following mechanisms:

- You can use traditional interface techniques. For example, if you send consumption to service providers every month (so they can compute their charges), you could use the Consumption Download interface.
- You can use “notifications” to communicate with service providers. Notifications are electronic transactions that service providers exchange to communicate information about a customer. For example, you could use a notification to send a message to an energy supplier when a customer stops service. Note: the term “Direct Access Service Request” (DASR) is synonymous with Notification.

The difference between notifications and traditional interfaces is subtle. Think of notifications as a generic interface that can be used to communicate many different things (e.g., you can use notifications to advise stops, meter exchanges, supplier switches). Traditional interfaces communicate only one thing (and you need one traditional interface for each “thing”). For example, one interface is devoted to downloading consumption, another is responsible for uploading pass through charges, etc.

Refer to [The Big Picture Of Notification Processing](#) for more information about notification. Refer to [The System Background Processes](#) for more information about the traditional interfaces that are supported in the system.

Designing Your SA Relationship Types and Service Providers

As explain in [A Service Agreement Can Have Many Types Of Relationships](#), SA relationship types and service providers are only required when you subcategorize your service agreements.

The topics in this section describe how to design your SA relationship types.

Contents

- [Designing SA Relationship Types](#)
- [Designing Service Providers](#)
- [Designing Your SA Types And Start Options For Sub SAs](#)
- [Reference Send Consumption Algorithm On Master SA Types](#)
- [Reference TBFU Algorithm On Master SA Types](#)
- [Designing SA Types For Service Provider Financial Settlements](#)

Designing SA Relationship Types

The easiest way to design SA relationship types is to start with the matrix of SA types designed in [SA Types And The Financial Design](#). For each SA type in the matrix, determine if either of the following questions is true:

- Can companies other than your own provide some subcategory of the service (and do you have some type of interaction with these companies)? For example, this would be true if you are an energy supply company because a different company is responsible for distributing the power to the customer (and you probably exchange consumption and financial transactions with this company).
- Does your organization use different rates for subcategories of the service? For example, this would be true if you use separate rates for water and wastewater service (even though both are based on the customer's water consumption).

If either of the above is true, you will need a SA relationship type for each subcategory of service.

We'll use an example to help make the point. Using the SA types we designed earlier (see [SA Types And The Financial Design](#)), we'll assume the following:

- We are designing the system for an electric, gas, water, waste water, and cable utility.
- Residential electric customers have a choice of energy supplier.
- Commercial and industrial electric customers can choose an energy supplier and a meter service provider.
- Commercial and industrial gas customers have a choice of energy supplier.

Note. We are not showing most of the SA types that we designed earlier because they do not have subcategories of service.

CIS BU/ SA Type	SA Relationship Type
G/COM	Energy supply
G/IND	Energy supply

E/RES	Energy supply
E/COM	Energy supply
	Meter service
E/IND	Energy supply
	Meter service

Note. Notice that we did not design a SA relationship type for our own distribution service. This is because our relationship type is implied (e.g., if you are a distribution company, you do not have to set up a SA relationship type for distribution service because the customer’s “master” service agreement is implicitly associated with distribution service).

Designing Service Providers

After you design your SA relationship types, you need to list every potential service provider for each SA relationship type.

CIS BU/ SA Type	SA Relationship Type	Service Provider
G/COM	Energy supply	AmeriGas
		TransGas
		Green Power
G/IND	Energy supply	AmeriGas
		TransGas
		Green Power
E/RES	Energy supply	Green Power
		Cheap Power
		Us
E/COM	Energy supply	ElectriCorp
		Cheap Power
		TeniCorp
	Meter service	MeterCorp
		Us
E/IND	Energy supply	ElectriCorp
		Cheap Power
		TeniCorp
	Meter service	MeterCorp
		Us

Next, list each unique service provider identified above:

Service Provider

AmeriGas
TransGas
Green Power
Cheap Power
Us
ElectriCorp
TeniCorp
MeterCorp

You will have at least one service provider for each entry in the above list. However, you may have to set up more than one service provider in the system for a given company. The topics below explain how this happens.

Contents

[Billing Relationship Segmentation](#)
[We Bill For Them - Payment Relationship Segmentation](#)
[Consumption Relationship Segmentation](#)
[Other Segmentations](#)
[Geographic Area Segmentation](#)

Billing Relationship Segmentation

As described under [Billing Relationships](#), a service provider may bill for you, you may bill for them, or you may each send a separate bill to the customer. In the table below, we have shown the assumed billing relationships for each service provider.

Service Provider	Billing Relationship
AmeriGas	<i>We Bill For Them – Bill Ready</i>
TransGas	<i>We Bill For Them – Bill Ready</i>
Green Power	<i>We Bill For Them – Rate Ready</i>
Cheap Power – Res	<i>We Bill For Them – Rate Ready</i>
Cheap Power – Com/Ind	<i>Dual Billing</i>
Us – Billable	<i>It's Us</i>
Us – Non billable	<i>None</i>
ElectriCorp	<i>They Bill For Us – Bill Ready</i>
TeniCorp	<i>We Bill For Them – Bill Ready</i>
MeterCorp	<i>Dual Billing</i>

Notice that we had to introduce additional service providers:

- Cheap Power has two service providers – one for residential customers, the other for commercial/industrial customers. This is necessary because we provide billing service for them for residential customers (***We Bill For Them – Rate Ready***), but for commercial and industrial customers they bill for themselves (***Dual Billing***).

- Our own service provider (the service provider known as “Us”) has two service providers – one for energy supply because we bill for the energy we supply (***It's Us***), and another for meter service because we don't create bills for meter service (***None***).

We Bill For Them - Payment Relationship Segmentation

As described under [Pay At Bill Time vs. Pay At Pay Time](#) when you provide billing service for a service provider you have to define if you pay the service provider when you bill the customer OR only later, when the customer pays you.

In the table below, we have shown the payment relationships for each service provider.

Service Provider	Billing Relationship	Payment Relationship
AmeriGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>
TransGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>
Green Power	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>
Cheap Power – Res	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>
Cheap Power – Com/Ind	<i>Dual Billing</i>	<i>N/A</i>
Us – Billable	<i>It's Us</i>	<i>N/A</i>
Us – Non billable	<i>None</i>	<i>N/A</i>
ElectriCorp	<i>They Bill For Us – Bill Ready</i>	<i>N/A</i>
TeniCorp – Com	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>
TeniCorp – Ind	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Pay Time</i>
MeterCorp	<i>Dual Billing</i>	<i>N/A</i>

Notice that we had to introduce an additional service provider for TeniCorp because for commercial customer we purchase the receivable (***Pay At Bill Time***), but for industrial customers we only pay them when we're paid by the customer (***Pay At Pay Time***).

Consumption Relationship Segmentation

As described under [Consumption Relationships](#) a service provider may send the customers' consumption to you, you may send consumption to them, or you may have no consumption relationship with a given service provider.

In the table below, we have shown the consumption relationships for each service provider.

Service Provider	Billing Relationship	Payment Relationship	Consumption Relationship
AmeriGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
TransGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
Green Power	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>	<i>N/A</i>
Cheap Power – Res	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>	<i>N/A</i>

Cheap Power – Com/Ind	<i>Dual Billing</i>	<i>N/A</i>	<i>N/A</i>
Us – Billable	<i>It's Us</i>	<i>N/A</i>	<i>N/A</i>
Us – Non billable	<i>None</i>	<i>N/A</i>	<i>N/A</i>
ElectriCorp	<i>They Bill For Us – Bill Ready</i>	<i>N/A</i>	<i>N/A</i>
TeniCorp – Com	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
TeniCorp – Ind	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Pay Time</i>	<i>We Send Consumption</i>
MeterCorp	<i>Dual Billing</i>	<i>N/A</i>	<i>N/A</i>

Notice that we didn't have to proliferate service providers due to consumption relationships.

Other Segmentations

The earlier parts of this discussion described the most common factors that cause the creation of service providers. However, many obscure factors could cause the introduction of more service providers. In this section, we explain these more obscure factors.

Contents

- [Payment Method](#)
- [Transfer Receivable Method](#)
- [Billable Charge Download Process](#)
- [Consumption Download Process](#)
- [Notification Upload Processing](#)
- [Notification Download Processing](#)
- [Financial Settlement Service Agreement](#)

Payment Method

As described under [Pay At Bill Time vs. Pay At Pay Time](#), when you provide billing service for a service provider you have to pay the service provider at some point in time. The algorithm that defines the amount to pay (and how the related adjustment is generated) is defined on the service provider record. If a service provider has different payment algorithms for different customer segments, you must split the service provider accordingly.

Transfer Receivable Method

As described under [When They Bill For Us, They Owe Us Money](#), a service provider will owe you money if they provide billing service for you. The algorithm that defines how to transfer the customer's receivable to the service provider is defined on the service provider record. If a service provider has different transfer A/R algorithms for different customer segments, you must split the service provider accordingly.

Billable Charge Download Process

As described under [They Bill For Us – Bill Ready](#), billable charges are interfaced to service providers who provide billing service for you. The background process that performs the interface of billable charges is defined on the service provider record. If a service provider has different billable charge interfaces for different customer segments, you must split the service provider accordingly.

Consumption Download Process

As described under [We Can Send Billed Consumption To Any Service Provider](#), consumption can be sent to any service provider. The background process that performs the interface of consumption is defined on the service provider record. If a service provider has different consumption interfaces, you must split the service provider accordingly.

Notification Upload Processing

As described under [Designing Notification Upload & Workflow Procedures](#), a service provider can send you notifications. Whenever a notification is uploaded, the system creates a workflow process to process each such notification. The type of workflow process that's created is controlled by the service provider's workflow process profile. If a service provider requires a different workflow process for a given type of notification (for whatever reason), you must split the service provider accordingly.

Notification Download Processing

As described under [Designing Notification Downloads](#), the system will send notifications to service providers when something noteworthy happens and when information is needed from a service provider. The type of notification that is sent to a service provider and the background process that interfaces the notification to the service provider is defined on the service provider's notification download profile. If a service provider requires a different type of notification to be sent or they have different interface protocols for a given type of notification, you must split the service provider accordingly.

Financial Settlement Service Agreement

As described under [Service Providers Have Service Agreements Too](#), service providers have service agreements. These service agreements contain how much you owe the service provider (if you bill for them) and how much they owe you (if they bill for you). If you want to have separate service agreements for financial settlements associated with distinct customer segments, you must split the service provider accordingly.

Geographic Area Segmentation

The following table reflects the new service providers that were added since we started [Designing Service Providers](#).

CIS BU/ SA Type	SA Relationship Type	Service Provider
G/COM	Energy supply	AmeriGas
		TransGas
		Green Power
G/IND	Energy supply	AmeriGas
		TransGas
		Green Power
E/RES	Energy supply	Green Power
		Cheap Power – Res
		Us – Billable
E/COM	Energy supply	ElectriCorp

E/IND		Cheap Power – Com/Ind
		TeniCorp – Com
	Meter service	MeterCorp
		Us – Non billable
	Energy supply	ElectriCorp
		Cheap Power – Com/Ind
		TeniCorp – Ind
	Meter service	MeterCorp
		Us – Non billable

Next, determine the postal code ranges in which a service provider is allowed to provide service.

CIS BU/ SA Type	SA Relationship Type	Service Provider	Postal Range
G/COM	Energy supply	AmeriGas	94000 - 95999
		TransGas	94000 - 95999
		Green Power	94000 - 95999
G/IND	Energy supply	AmeriGas	94000 - 95999
		TransGas	94000 - 95999
		Green Power	94000 - 95999
E/RES	Energy supply	Green Power	94000 - 95999
			93000 - 93999
		Cheap Power – Res	94000 - 95999
		Us – Billable	94000 - 95999
E/COM	Energy supply	ElectriCorp	94000 - 95999
		Cheap Power – Com/Ind	94000 - 95999
		TeniCorp – Com	94000 - 95999
	Meter service	MeterCorp	94000 - 95999
		Us – Non billable	94000 - 95999
E/IND	Energy supply	ElectriCorp	94000 - 95999
		Cheap Power – Com/Ind	94000 - 95999
		TeniCorp – Ind	94000 - 95999
	Meter service	MeterCorp	94000 - 95999
		Us – Non billable	94000 - 95999

Next, we need to strip off the SA types because the postal ranges are defined for combinations of service provider and SA relationship type. Notice the problem – we have a service provider - Green Power has different postal ranges for the same SA relationship type. You have two ways to fix this problem, you can split your service provider (have one for the gas and another for the electric), or you can split the SA relationship type (have one for the gas and another for the electric). We've chosen the former in our example.

SA Relationship Type	Service Provider	Postal Range
Energy supply	AmeriGas	94000 - 95999
	TransGas	94000 - 95999
	Green Power – Gas	94000 - 95999
Energy supply	AmeriGas	94000 - 95999
	TransGas	94000 - 95999
	Green Power – Gas	94000 - 95999
Energy supply	Green Power – Electric	94000 - 95999
		93000 - 93999
	Cheap Power – Res	94000 - 95999
	Us – Billable	94000 - 95999
Energy supply	ElectriCorp	94000 - 95999
	Cheap Power – Com/Ind	94000 - 95999
	TeniCorp – Com	94000 - 95999
Meter service	MeterCorp	94000 - 95999
	Us – Non billable	94000 - 95999
Energy supply	ElectriCorp	94000 - 95999
	Cheap Power – Com/Ind	94000 - 95999
	TeniCorp – Ind	94000 - 95999
Meter service	MeterCorp	94000 - 95999
	Us – Non billable	94000 - 95999

In the table below, we have shown the final list of service providers.

Service Provider	Billing Relationship	Payment Relationship	Consumption Relationship
AmeriGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
TransGas	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
Green Power - Electric	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>	<i>N/A</i>
Green Power - Gas	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>	<i>N/A</i>
Cheap Power – Res	<i>We Bill For Them – Rate Ready</i>	<i>Pay At Billing Time</i>	<i>N/A</i>
Cheap Power – Com/Ind	<i>Dual Billing</i>	<i>N/A</i>	<i>N/A</i>
Us – Billable	<i>It's Us</i>	<i>N/A</i>	<i>N/A</i>
Us – Non billable	<i>None</i>	<i>N/A</i>	<i>N/A</i>

ElectriCorp	<i>They Bill For Us – Bill Ready</i>	<i>N/A</i>	<i>N/A</i>
TeniCorp – Com	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Billing Time</i>	<i>We Send Consumption</i>
TeniCorp – Ind	<i>We Bill For Them – Bill Ready</i>	<i>Pay At Pay Time</i>	<i>We Send Consumption</i>
MeterCorp	<i>Dual Billing</i>	<i>N/A</i>	<i>N/A</i>

Designing Your SA Types And Start Options For Sub SAs

When you were [Designing Service Providers](#), you defined the service providers that were valid for every combination of SA type and SA relationship type. If you provide billing services for another service provider or if you subcategorize your own services, you another task – you have to design the SA types for your sub SAs.

As described earlier, there will be a separate [Sub Service Agreement](#) for every SA relationship for which we calculate a bill segment. Every sub SA must reference an SA type. The following table shows sample SA types (notice that they are only used for **We Bill For Them** and **It's Us** service providers).

CIS BU/ SA Type	SA Relationship Type	Service Provider	SA Type(s) for Sub SAs
G/COM	Energy supply	AmeriGas	AG1
		TransGas	TG1
		Green Power - Gas	GP-GC1
G/IND	Energy supply	AmeriGas	AG1
		TransGas	TG1
		Green Power - Gas	GP-GI1
E/RES	Energy supply	Green Power - Electric	GP-ER1
		Cheap Power – Res	CP-ER1
		Us – Billable	US-ER1
E/COM	Energy supply	ElectriCorp	Not applicable – they bill for us
		Cheap Power – Com/Ind	Not applicable – dual billing
		TeniCorp – Com	TC1
	Meter service	MeterCorp	Not applicable – dual billing
		Us – Non billable	Not applicable – no billing
E/IND	Energy supply	ElectriCorp	Not applicable – they bill for us
		Cheap Power – Com/Ind	Not applicable – dual billing
		TeniCorp – Ind	TC1
	Meter service	MeterCorp	Not applicable – dual billing
		Us – Non billable	Not applicable – no billing

The design steps required to set up these SA types are similar to those described under [Designing SA Types For Service Agreements With Service Points](#). The following points provide a few suggestions that will help you design your SA types for sub SAs:

- The business unit should be the same as defined for the master service agreement.
- Service type should be set up as per [Service Segmentation](#).
- Distribution code should be set up as per [Receivable Segmentation](#).
- Obviously, the sub SA switch should be turned on.
- Start options:
 - If the SA type is used for a **We Bill For Them – Rate Ready** service provider or for yourself, the start options should be Required because rates, contract riders and/or contract values will be populated on the sub SA from a start option. Refer to [Automatic Creation of Sub SAs](#) for more information.
 - If the SA type is used for a **We Bill For Them – Bill Ready** service provider, the start options should be not allowed because we don't need to default rates, contract riders and/or contract values on billable charge sub SAs.
- The payment distribution and late payment information should be set up as for any SA type. Refer to [Cash Distribution Segmentation](#) and [Late Payment Charge Segmentation](#) for more information.
- If the SA type is used for **We Bill For Them – Bill Ready** service provider, the special role should be **Billable Charge**, otherwise it should not be used. Remember, you should not use sub SAs for **Cash Deposits**.
- Deposit class should be used if the sub SA is covered by a deposit. Refer to [Deposits Issues](#) for more information.
- The one time charge switch should be off.
- Bill segment type:
 - If the SA type is used for a **We Bill For Them – Bill Ready** service provider, the bill segment type should reference a bill segment creation algorithm that creates bill segments from billable charges (and generates bill segment errors until the last night of the bill cycle). Refer to [The Bill Ready Calculation Method](#) for more information.
 - If the SA type is used for a **We Bill For Them – Rate Ready** service provider, the bill segment type should reference a get consumption algorithm that gets consumption from the master SA, and a bill segment creation algorithm that applies a rate.
- Specify a characteristic premise is required if the sub SA is associated with premise-oriented service.
- The calendar billing options should not be used.
- The recurring charge information should not be used.
- Sub SAs used for **We Bill For Them – Rate Ready** service providers should NOT be eligible for budgets and therefore the eligible for budget switch should be off. This admonition is given because budget billing causes current amount due to be out-of-sync with payoff amount due and we don't want this to happen for sub SAs. Why? Because we use the bill segments associated with these sub SAs to determine how much we owe the service provider.

- If the SA type is used for a ***We Bill For Them – Rate Ready*** service provider or for your own company, rates need to be set up; otherwise they should not be used.
- Sub SAs never reference service points. Refer to [Only The “Master” Service Agreement Is Linked To Service Points](#) for more information.
- Adjustment types profiles should be set up accordingly.
- Refer to [Credit and Collection Issues](#) for recommendations in respect of the debt class, write off debt class, and severance criteria associated with these SA types.
- Billable charge templates should not be used.
- Completion algorithms cannot be used for sub SAs.
- If the SA type is used for a ***We Bill For Them*** service provider, you should link to the SA type the FT freeze algorithm that controls how we pay the service provider. Refer to [Pay At Bill Time vs. Pay At Pay Time](#) for how this algorithm is used.
- The creation of bill segments for the sub SAs occurs after the bill segment for the related master SA is created. If you populate a [billing processing sequence](#) on an SA type for a sub SA, it is used to control the order in which the sub SAs for a given master SA are processed relative to each other.

Refer to [Setting Up SA Types](#) for how to set up these new SA types in the system.

Reference Send Consumption Algorithm On Master SA Types

As explained under [We Can Send Billed Consumption To Any Service Provider](#), when a master SA's bill segment is frozen, the system must check if there are any service providers who need the bill segment's consumption. If so, it sets up the data necessary to interface the master SA's consumption (snapshot on the bill segment) to the service provider(s). The system will only do this if you specify an appropriate FT Freeze Algorithm on the master SA types. Refer to [SA Type – Algorithm](#) (FT Freeze Algorithm) for more information.

Reference TBFU Algorithm On Master SA Types

As explained under [They Bill For Us](#), when a bill is completed, the system needs to check if there are any service providers who bill for us associated with the bills “master” SAs. If so, it sets up the data necessary to interface the master SA's charges to the service provider and to transfer the receivable balance from the customer to the service provider. The system will only do this if you specify an appropriate FT Completion Algorithm on the master SA types. Refer to [SA Type – Algorithm](#) (Bill Completion Algorithm) for more information.

Note. If there are multiple master SAs on a bill, the financial transactions associated with each respective master SA could be routed to different service providers (e.g., one service provider could bill for gas and another could bill for electricity). Refer to [Different Service Providers Can Bill Different Services](#) for more information.

Designing SA Types For Service Provider Financial Settlements

As explained in [Service Providers Have Service Agreements Too](#), **We Bill For Them** and **They Bill For Us** service providers require a service agreement. You must create SA types for these types of service agreements. The following points provide a few suggestions that will help you design these financial settlement SA types:

- Service type should probably be a non-service oriented service type.
- Distribution code for **We Bill For Them** settlement SAs should be a payable account (or treat it as a “contra” receivable. Refer to [Receivable Segmentation](#) for more information.
- The sub SA switch should be turned off.
- Start options are not allowed.
- The payment distribution and late payment information should be set up as for any SA type. Refer to [Cash Distribution Segmentation](#) and [Late Payment Charge Segmentation](#) for more information.
- Special role should not be used.
- Deposit class should be used if the settlement service agreement is covered by a deposit. This would probably only be used for **They Bill For Us** service providers (because they will owe us money).
- The one time charge switch should be off.
- These service agreements are not billable and therefore none of the billing information should be specified.
- The characteristic premise switch should be off.
- Rates should not be used.
- Service points should not be used.
- Adjustment types profiles should be set up accordingly.
- Debt class, write off debt class, and severance criteria should be set up accordingly.
- Billable charge templates should not be used.
- Completion algorithms should not be used.
- Freeze algorithms should not be used.

Refer to [Setting Up SA Types](#) for how to set up these new SA types in the system.

Setting Up SA Relationship Information

In the previous section, Designing Your SA Relationship Types and Service Providers, we presented a case study that illustrated a mythical organization’s SA relationship information. In this section, we explain how to set up this information.

Contents

[Setting Up SA Relationship Types](#)

[Setting Up Service Providers](#)
[Setting Up SA Types and Start Options For Sub SAs](#)
[Setting Up SA Types For Financial Settlements](#)
[Update Master SA Types With FT Freeze and Bill Completion Algorithms](#)
[Setting Up SA Relationships For SA Types](#)

Setting Up SA Relationship Types

Open **Admin Menu, SA Relationship Type** to define your SA relationship types. Refer to [Designing SA Relationship Types](#) for more information.

Description of Page

Enter an **SA Relationship Type** code and **Description** for every relationship type.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_SA_REL_TYPE](#).

Setting Up Service Providers

The topics in this section describe how to set up service providers.

Contents

[Service Provider - Main](#)
[Service Provider - Detail](#)
[Service Provider - SA Relationship Type](#)
[Service Provider - Bill Messages](#)

Service Provider - Main

Open **Admin Menu, Service Provider** to define core information about a service provider.

Description of Page

Enter a unique **Service Provider** code for the service provider.

End a brief **Description** of the service provider.

If you communicate with this service provider via notification messages or outbound messages, indicate the service provider's **External System**.

If you send notifications to this service provider, select a **Notification DL (download) Profile** that is used to define the configuration of the outgoing messages. Refer to [Designing Notification Download Profiles](#) for more information.

Select the **Person ID** that contains this service provider's phone numbers and demographic information.

If you bill for the service provider or if they bill for you, select the **Service Agreement** that holds how much you owe them or they owe you. Refer to [Service Providers Have Service Agreements Too](#) for more information.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_SPR](#).

Service Provider - Detail

Open **Admin Menu**, **Service Provider** and navigate to the **Detail** tab to define additional information about your service providers.

Description of Page

Define the **Billing Relationship** you have with the service provider. Refer to [Billing Relationships](#) for more information.

If you provide billing services for this service provider (i.e., the **Billing Relationship** is **We Bill For Them**), define the **Payment Relationship**. Refer to [Pay At Bill Time vs. Pay At Pay Time](#) for more information. You may not be paying some service providers as such. Rather, the customer's receivables are simply transferred to the service provider, e.g., when you calculate discounts for special [negotiated terms](#). For these service providers, choose a payment relationship of **Pay Spr Not Applicable**.

Define the **Consumption Relationship** you have with the service provider. Refer to [Consumption Relationships](#) for more information.

If a service provider reads meters and they can override the customer's service cycle, turn on **Overrides Service Cycle**. Refer to [MDMAs And Service Cycles](#) for more information.

As described under [Pay At Bill Time vs. Pay At Pay Time](#), when you provide billing service for a service provider you have to pay the service provider at some point in time. The **Pay Service Provider Algorithm** defines the amount to pay and how the related adjustment is generated. Refer to [Paying The Service Provider - Technical Implementation](#) for more information about how this algorithm is used. If you haven't done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that increases how much is owed the service provider. Click [here](#) to see the algorithm types available for this plug-in spot.

As described under [When They Bill For Us, They Owe Us Money](#), a service provider will owe you money if they provide billing service for you. The **Transfer A/R Algorithm** defines how to transfer the customer's receivable to the service provider. Refer to [A/R Transfer - Technical Implementation](#) for more information about how this algorithm is used. This algorithm is also used to transfer receivables when you calculate discounts for special [negotiated terms](#). If you haven't done so already, you must set up this algorithm in the system. To do this:

- Create a new algorithm (refer to [Setting Up Algorithms](#)).
- On this algorithm, reference an Algorithm Type that transfers financial transactions from the customer to the service provider. Click [here](#) to see the algorithm types available for this system event.

As described under [They Bill For Us – Bill Ready](#), billable charges are interfaced to service providers who provide billing service for you. The **Billable Charge Download Process** performs the interface of billable charges to the service provider. Refer to [Routing Billable Charges To Service Providers - Technical Implementation](#) for more information about how this process is used.

As described under [We Can Send Billed Consumption To Any Service Provider](#), consumption can be sent to any service provider. The **Consumption Download Process** performs the interface of consumption to the service provider. Refer to [Routing Consumption To Service Providers - Technical Implementation](#) for more information about how this process is used.

Service Provider - SA Relationship Type

Open **Admin Menu**, **Service Provider** and navigate to the **SA Relationship Type** page to define the types of relationships (e.g., energy supplier, energy distributor, meter data management agency) associated with a service provider and the postal code ranges in which the service provider operates.

Description of Page

Use the **SA Relationship Types** collection to define this service provider's **SA Relationship Types**. Use the collection that appears in the grid to define the **Postal Code** ranges in which this service provider is allowed to operate for each **SA Relationship Type**.

Important! After defining the SA relationship types that can be associated with a service provider, you must then define the SA types on which the service provider / SA relationship type combination can be used. This information is defined using [Setting Up SA Relationships For SA Types](#).

Service Provider - Bill Messages

Open **Admin Menu**, **Service Provider** and navigate to the **Bill Messages** page to define bill messages to appear on bills that contain charges associated with a service provider.

Description of Page

Use the **Bill Messages** collection to define **Bill Message** codes that should appear on bills that contain charges associated with a given service provider. For each message, also specify the **Start Date** and **End Date** when such a message should appear on the bill (leave **End Date** blank if the message should appear indefinitely).

Where Used

The system snaps bill messages on a bill during bill completion. Refer to [The Source Of Bill Messages](#) for more information.

Setting Up SA Types and Start Options For Sub SAs

The SA types and start options described under [Designing Your SA Types And Start Options For Sub SAs](#) must be set up. Refer to [Setting Up SA Types](#) for how to do this.

Setting Up SA Types For Financial Settlements

The SA types described under [Designing SA Types For Service Provider Financial Settlements](#) must be set up. Refer to [Setting Up SA Types](#) for how to do this.

Update Master SA Types With FT Freeze and Bill Completion Algorithms

Refer to [SA Type – Algorithm](#) for more information.

Setting Up SA Relationships For SA Types

Contents

[SA Type SA Relationship Type - Main](#)

[SA Type SA Relationship Type - Sub SA Type](#)

SA Type SA Relationship Type - Main

Open **Admin Menu, SA Type SA Rel. Type** to define the types of SA relationships and service providers that can be associated with a SA type.

Description of Page

Define the **SA Relationship Type** that can be associated with service agreements of this **SA Type**.

Note. You may only define SA Relationship Types for “master service agreements”.

Turn on the **Required** switch if this **SA Relationship Type** must be defined for service agreements of this type. Refer to [Defaulting Relationship Types And Defaulting Service Providers](#) and [Required Relationship Types and Billing](#) for more information.

Indicate if **Gaps in SA Relationships** of this type that are associated with this service agreement type are **Allowed** or **Not Allowed**. You should only select **Allowed** if relationships of this type can be expired without a relationship with another service provider to replace it. Deregulated relationships typically should not have gaps in the relationship. For example, a relationship with an energy service provider should not expire unless a relationship with another energy service provider replaces it. Refer to [negotiated terms](#) for an example of SA relationship types that allow gaps in SA relationships.

Use the collection to define the **Service Providers** who can be associated with this **SA Relationship Type** on service agreements of this **SA Type**.

Note. Only **Service Providers** previously defined as being valid for the **SA Relationship Type** can be specified (refer to [Service Provider – SA Relationship Type](#) for how to link a service provider to a SA relationship type).

Turn on **Default SPR** if the **Service Provider** should be defaulted on newly created SA relationships. Refer to [Defaulting Relationship Types And Defaulting Service Providers](#) for more information.

Use **Status** to control if the **Service Provider's** relationship to the SA Type / SA Relationship type is **Active** or **Inactive**. Only **Active** service providers can be linked to service agreements of this type. The system allows **Inactive** service providers in order to support historical service providers who are no longer active and to allow you to set up new service providers in advance of their start date.

Use the drill down button adjacent to a service provider to view the valid Sub SA Types. Alternatively, navigate the Sub SA Type tab and scroll until you find the desired service provider.

Where Used

When a new SA relationship is defined for a service agreement, the system uses this information to make sure the relationship is valid and that the associated service provider is valid.

SA Type SA Relationship Type - Sub SA Type

Open **Admin Menu, SA Type SA Rel Type** and navigate to the **Sub SA Type** tab to define valid sub SA types for service providers associated with a SA type.

If you provide billing services for the service provider (i.e., the service provider's billing relationship is **We Bill For Them**) or if you subcategorize your own charges (i.e., the service provider is your organization and it has a billing relationship of **It's Us**) a [Sub Service Agreement](#) will be created for the service provider.

Description of Page

The information in the collection defines the valid **Sub CIS Division** and **Sub SA Types** of these sub service agreements. Those entries marked as **Create Initially** are used by the process that creates sub SAs for new SA relationships. This background process uses this information as follows:

- If the sub SA's SA type doesn't use Start Options (as defined on SA Type – Main), the background process simply creates a sub SA with the given SA type. Note: these types of sub SAs are typically used for service providers who send their charges to you (i.e., they have a billing relationship of **We Bill For Them – Bill Ready**). This is because Billable Charge service agreements are used for these types of service agreements and billable charge service agreements contain very little information.
- If the sub SA's SA type uses Start Options (as defined on SA Type – Main), the **Start Option** defined in the collection is used to populate the sub SA with default values (e.g., rate, contract rider, etc.). Refer to [Setting Up Start Options](#) for more information.

Where Used

The process that creates sub SAs for new SA relationships uses this information to determine the number and type of sub SAs to create for each SA relationship.

Negotiated Terms

The topics in this section describe the use of SA relationship functionality for applying certain types of negotiated terms. It assumes that you are familiar with SA relationship functionality in general.

Negotiated terms are optional. The functionality described in this section is only relevant if your organization offers this functionality.

Umbrella agreements. If the **Contract Management** module is not [turned off](#), you may also choose to use umbrella agreements to manage the functionality described here.

Imagine that the head office for a multi-site organization negotiates special terms that cover a number of its sites, each of whom have their own account and service agreement. The negotiated terms typically involve discounts. These discounts may be realized

- Under a single service agreement, separate and distinct from the service agreements that are covered, or
- Individually, for each covered service agreement. In this case, two sub-scenarios exist:

- The discounts may be reflected on the bill segment of each service agreement, or
- The discounts may be transferred to another service agreement (the group's service agreement). The discount does NOT appear on the individual bill segment for the service agreements that are covered by the negotiated terms.

SA relationships track and manage complex business relationships between a customer and a service provider. You can define the above relationships using service agreement relationship functionality. This is a special type of SA relationship in which the head office is the service provider.

Contents

[Examples Of Special Discounts](#)

[Setting Up The System To Enable Negotiated Terms](#)

Examples Of Special Discounts

The following topics illustrate examples of how your organization may configure the system to handle this business functionality.

Contents

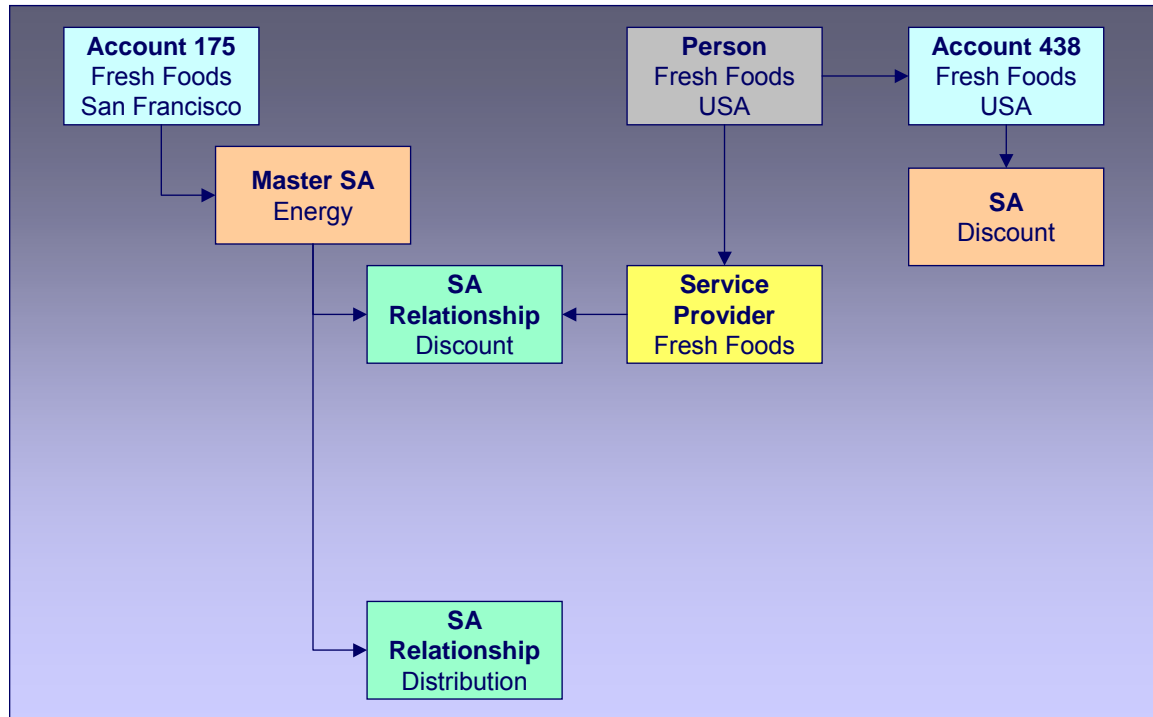
[Example Using Aggregated Consumption](#)

[Example Using Site-by-site Discount](#)

Example Using Aggregated Consumption

A customer, Fresh Foods has 10 stores each with its own account and electric service agreement, i.e., each store is billed separately. Fresh Food's head office decides to negotiate a group discount that applies to one or more of the individual stores.

In the following diagram, one store's account is shown.



Example Using Aggregated Consumption

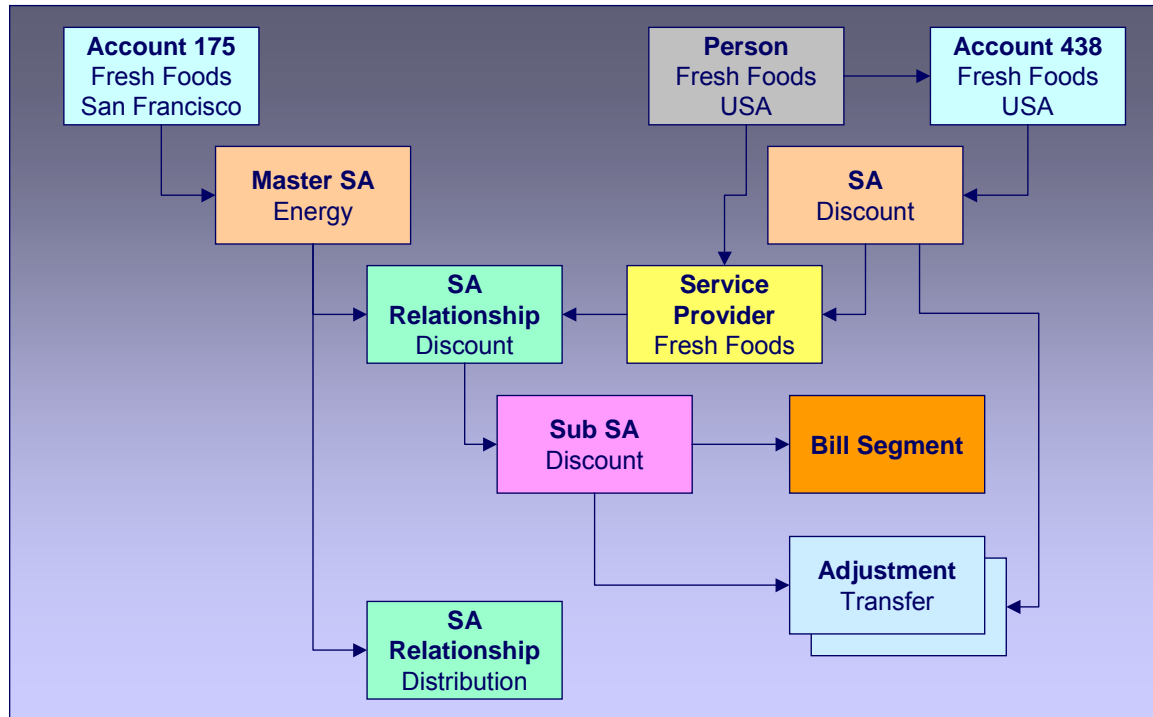
Note the following:

- An SA relationship is created for the energy SA covered by the negotiated terms.
- The service provider in the relationship is the head office.
- This relationship does not have a sub-SA because no additional billing services are provided for each individual covered service agreement.
- Discounts are calculated using the head office discount service agreement and affect only this service agreement and therefore only the head office's account.
- The master SA may have other deregulated relationships, such as the distribution relationship shown.

Aggregated consumption algorithm. In this scenario, the service provider's service agreement amalgamates the group's consumption and applies a rate to calculate the discount. This algorithm is not provided – you get to write this algorithm to meet your needs.

Example Using Site-by-site Discount

In this scenario, the discount is calculated on a site-by-site basis.



Example Using Site By Site Discount

Note the following differences between this example and the one where consumption is aggregated:

- Each service agreement participating in the discount has an SA relationship with a sub SA.
- Discounts are calculated on a site-by-site basis by the sub SA for each service agreement covered by the discount agreement.
- The discounts may be transferred to the head office service agreement using transfer adjustments. In this case, you may choose to not show the discount on the individual site's bills.

Setting Up The System To Enable Negotiated Terms

The above topics provided background information about how special negotiated terms could be supported in the system. The following discussion summarizes the various setup tasks alluded to above. These notes highlight the setup required in addition to that detailed in [Setting Up SA Relationships Information](#).

Contents

Algorithm
 SA Relationship Type
 Service Provider
 SA Type SA Relationship Type
 SA Type

Algorithm

If you use sub SAs to calculate charges / discounts for each covered service agreement individually and you want to transfer the charges / discounts to the service provider's service agreement at bill completion, you will need to set up a bill completion algorithm to do this.

Refer to the algorithm type [BCMP-TR](#) for more information.

SA Relationship Type

You will need an SA relationship type that will be used for each type of negotiated term or discount.

Service Provider

Set up a service provider using the principal party of the negotiated discount. For example, if a head office negotiates a special agreement for its sites, you should set up a new service provider representing the head office.

- **Billing Relationship** will depend on the type of negotiated terms you choose to set up. For terms where debits / credits are calculated for each covered service agreement individually, select **We Bill For Them, Rate Ready**. We are effectively billing on behalf of the principal party. For terms where the discount is calculated on an amalgamated SA there is no billing relationship because no additional billing services are provided for each covered service agreement.
- Select a **Payment Relationship** of **Pay SPR Not Applicable** if debits / credits to the sub-SA should not be transferred to the service provider. If you want to transfer charges at bill completion time using a bill completion algorithm, you should also select **Pay SPR Not Applicable** because you will not be creating adjustments to pay the principal party service at payment time or at billing time.
- Enter a **Transfer A/R Algorithm** to transfer the debit / credit from the sub-SA to the principal party's service agreement. If you do not want to transfer the debit / credit to the principal party's service agreement, you do not need to specify a transfer algorithm.

SA Type SA Relationship Type

Associate the SA relationship type setup for negotiated terms with the SA type(s) of the service agreements that are covered by the terms. The following are suggested values for the SA type SA relationship type:

- **Required**. Relationships of this type should not be required for the SA type because the negotiated terms will only cover selected service agreements with that SA type.
- **Gaps in SA Relationship** are **Allowed** if service agreements of this type do not have to have an SA relationship of this type with a service provider throughout the effective period of the SA.
- Add the service provider set up above to the list of valid service providers for this SA type and SA relationship type combination.

SA Type

Set up an SA type for the sub SAs that will be used to calculate discounts. This SA type should use the [BCMP-TR](#) bill completion algorithm if you want to transfer the discounts to the head office.

TBFU Deregulated Relationship with WBFT Negotiated Terms Relationship. If you have the unusual situation where a master service agreement has a relationship with a TBFU service provider as well as a negotiated term "relationship" with a WBFT service provider, and you transfer charges / discounts from the discount agreement sub SA to the WBFT service provider, you will not be able to use the algorithms provided to transfer receivables to the service providers. This is because bill completion algorithms associated with the SA type of a master service agreement are executed before any bill completion algorithms associated with the SA types of sub service agreements. Consequently the TBFU XFER algorithm associated with the master service agreement will transfer all receivables from related sub SAs to the TBFU service provider before the BCMP-TR algorithm can transfer the discount agreement receivables. You will need to modify the TBFU XFER bill completion algorithm that transfers receivables from the customer to the TBFU service provider to exclude the discount agreement receivables.

Defining Interval Billing Options

The system provides capability to bill complex customers who measure usage in small intervals, such as one hour, thirty minutes, fifteen minutes, etc. The “Interval Billing” subsystem covers a variety of functionality including:

- Capturing billable interval data quantities linked to a service agreement
- Capturing “raw” interval data quantities linked to service points, which may be later aggregated to create billable quantities for the service agreement
- Application of contract-based modifications to produce interval data quantities
- Apply interval prices to interval quantities. (Sometimes this is known as “direct billing”)
- Support time of use mapping, including override maps created for certain special periods, such as interruptions
- Support time of use pricing and time of use contract values
- Support special contract options under which certain calculations supporting a contract’s rate may be overridden or altered occasionally for specific periods of time

Note. The ability to capture and maintain interval prices and TOU prices is officially part of the Rates subsystem. Although we will refer to them here, the main documentation is in the Rates administration guide.

Warning! Setting up the interval billing control tables is as challenging as your organization’s interval billing rules. If you have simple rules then your setup process will be straightforward. If your interval billing rules are complicated (e.g., specific prices for each customer, complicated TOU Mapping rules etc.), then your setup process will be more challenging.

Separate modules. Please note that interval billing functionality is associated with separate modules, **Complex Billing** and **Meter Data Management**. If these modules are not applicable to your business you may turn them off. Refer to [Turn Off A Function Module](#) for more information.

Contents

[The Big Picture of Interval Billing](#)
[Designing Interval Billing Options](#)
[Setting Up Interval Billing Options](#)

The Big Picture of Interval Billing

This section provides an overview of important Interval Billing concepts with which you should be familiar before you set up your Interval Billing control tables.

Contents

[Interval Pricing](#)
[Raw Data Collection and Aggregation](#)

Time of Use Billing

Interval Pricing

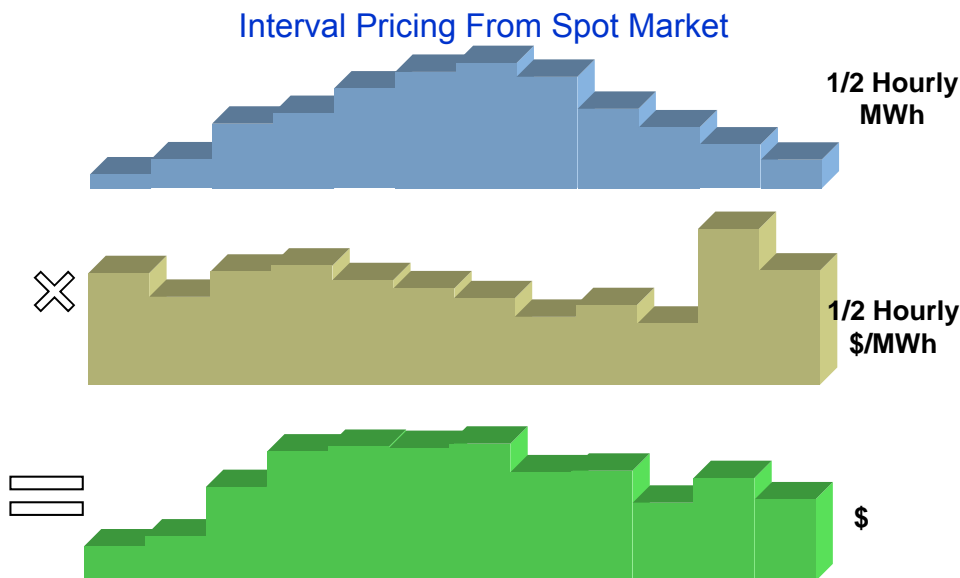
This section provides an overview of concepts related to setting up interval pricing options for your rates. Applying interval prices to interval quantities is sometimes referred to as 'direct billing'.

Contents

- [Interval Pricing Rate Application](#)
- [Physical Attributes of Interval Data vs. Its Role](#)

Interval Pricing Rate Application

Interval Pricing is the term used to describe applying interval prices to interval quantities to arrive at a bill calculation line item.



What data is needed in order to apply a rate component for an interval pricing scenario and how is this data defined?

- You need prices that vary at a given interval. Interval prices are stored for a **Bill Factor**/characteristic. Refer to [Bill Factor Interval Values](#) for more information.
- You need consumption values for each corresponding interval. The consumption values are stored for an Interval Profile linked to a service agreement. Refer to [Billable Interval Quantities for a Service Agreement](#) for more information. In order to find the correct interval profile for the service agreement, the rate component will reference a **Profile Relationship Type**. Refer to [Physical Attributes of Interval Data vs. Its Role](#) and [Interval Data Serves a Role for a Service Agreement](#) for more information.
- You need an algorithm in order to know how to apply the prices correctly. The rate component will reference a **Calculation Algorithm**, which will be executed to apply the prices to the quantities.

Refer to [Designing Your Interval Rate Components](#) for help in designing rate components of this type. Refer to [Setting Up Interval Pricing Rate Components](#) for more information about setting up this type of rate component.

Physical Attributes of Interval Data vs. Its Role

You will see, as you learn more about the design of interval billing, that there are two control tables that are important for defining billable interval data:

- Profile Type – this defines the physical attributes of the interval data
- Profile Relationship Type – this defines the role that the interval data is playing for a particular contract. You can also think of this as defining the business purpose of the data.

Contents

[Interval Data Physical Attributes](#)

[Business Role of Interval Data](#)

[The Business Role Defines Interval Data Physical Attributes](#)

Interval Data Physical Attributes

When defining a collection of billable data, there are basic attributes, which need to be defined:

- Unit of Measure
- Minutes per Interval
- Service Quantity Identifier
- Is this data owned by a service agreement or is it common data?

These are physical attributes of the data and have nothing to do with the business purpose of the data. This information is defined on the Profile Type.

The profile type may also include algorithms related to its data:

- Validation algorithms may be used to check and correct various conditions related to the interval data. Refer to [Validation of Profile Data](#) for more information.
- A creation algorithm may be used to derive data for a profile. Refer to [Creation of Profile Data through Data Derivation](#) for more information.

Refer to [Designing Interval Profile Types](#) for more information.

You will see later that TOU Maps also have a TOU Map Type. Refer to [Physical Attributes of a TOU Map](#) for more information.

Business Role of Interval Data

The Profile Relationship Type is used by the system to indicate the “role” that a collection of billable interval data is playing. The following are some examples of roles that interval data may play:

- Measured Demand
- Contract Demand
- Aggregated Heating Demand
- Hedge Cover
- Excess Demand
- Reactive Energy

The profile relationship type is used by the rate component to indicate the data being billed. For example, the rate is billing “excess demand”. When applying the rate, the system will determine which data is playing the role of “excess demand” for the service agreement. Refer to [Billable Interval Quantities for a Service Agreement](#) and [Interval Data Serves a Role for a Service Agreement](#) for more information.

You will see later that TOU Maps also have a “role”. Refer to [Business Role of a TOU Map](#) for more information.

The Business Role Defines Interval Data Physical Attributes

To enable proper setup, you will need to define the valid [profile types](#) for each profile relationship type.

Raw Data Collection and Aggregation

This section provides an overview of concepts related to setting up your control tables to support the capturing of raw interval data.

Refer to [The Big Picture of Raw Data Collection and Aggregation](#) for more information.

Note. If your company uses an external system for collecting, adjusting and aggregating raw interval data, then you may skip this section.

Physical Attributes of Raw Interval Data

Raw interval data is available for use by algorithms in the system to create billable data for the service agreement. This data will not have any ‘business’ role, as with the billable interval data, described above.

As a result only ‘physical’ attributes of the raw interval data need to be defined, such as:

- Unit of Measure
- Minutes per Interval
- Service Quantity Identifier

This information is defined on the Interval Register Type.

The [interval register type](#) may also include algorithms related to its data:

- Validation algorithms may be used to check and correct various conditions related to the interval data. Refer to [Validation of Register Data](#) for more information.

Time of Use Billing

This section provides an overview of concepts related to setting up your control tables to support time of use billing.

Contents

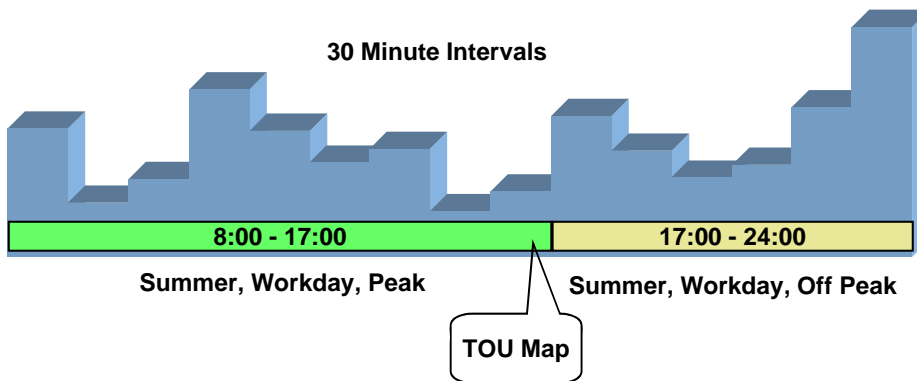
[Defining Time of Use Periods](#)

[Grouping of TOU Codes](#)
[Time of Use Mapping and Pricing](#)
[Physical Attributes of TOU Map vs. Its Role](#)

Defining Time of Use Periods

Many customers choose not to price their interval data using interval prices. Time of Use Mapping enables a customer to map out time of use periods for their usage. This option for interval data might be preferred because:

- Typically it involves fixed prices for the use periods
- It is more manageable than direct billing
- It is easier for a customer to forecast and budget



A TOU Map holds the collection of time period definitions. The TOU Map has a TOU Map Type, which defines the minutes per interval. This is similar to the interval profile and profile type.

The time period definitions for a TOU Map indicate the TOU code for a given date and time.

Map #123 (TOU Group 2)	
Effective 1 Jan 2000	
Interval Date/Time	
30/Apr/00 16:30	On Peak/Winter
30/Apr/00 16:45	On Peak/Winter
30/Apr/00 17:00	Off Peak/Winter
30/Apr/00 17:15	Off Peak/Winter
...	
01/May/00 7:45	Off Peak/Summer
02/May/00 8:00	On Peak/Summer
02/May/00 8:15	On Peak/Summer
...	

All the possible time of use codes for a given map are grouped together in a TOU group. Refer to [Grouping of TOU Codes](#) for more information.

Grouping of TOU Codes

Contents

[Overview](#)
[TOU Sequence](#)

Overview

The time of use map's purpose is to define time of use codes for given time periods. For example:

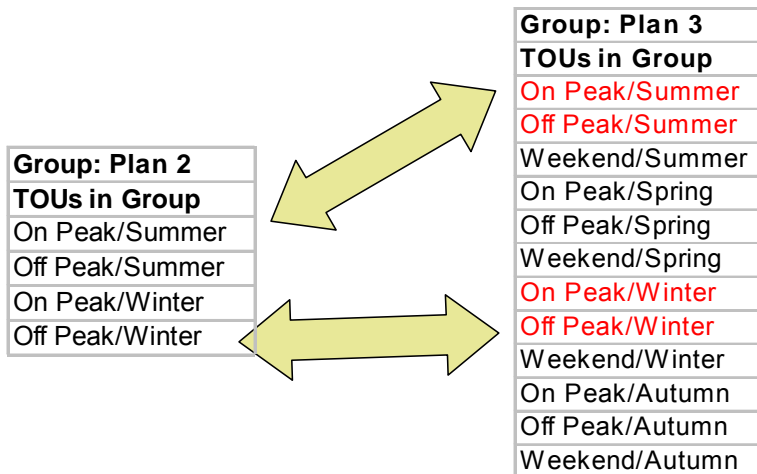
1/Jan/2001 05:00:00 is Off Peak, Winter

1/July/2001 13:00:00 is Weekend, Summer

5/July/2001 12:30:00 is On Peak, Summer

The time of use codes of "Off Peak, Winter" and "Weekend, Summer" are user-defined and use the same [TOU code](#) available for defining TOU for simple metering.

The set of time of use codes that make up a certain TOU map, are grouped together using the TOU group. The TOU group is a logical grouping of time of use codes. A TOU can exist on more than one TOU group.



The [TOU group](#) is used to define the collection of time of use codes available for a TOU map. In addition, you will see TOU group used in other areas of interval billing where a collection of time of use codes is required.

TOU Sequence

If desired, you may use sequence number to indicate the relative position or relative priority of each TOU code within a TOU group. This sequence number is not used by any system functionality, but is available for you to use in a plug-in algorithm.

For example, assume that your customer's contract states that if the usage for the time period "ON" is below a certain contract limit, they will be charged the "ON" price. However, if the usage exceeds this contract limit, they will be charged a higher price for this usage. Let's assume the same is true for "OFF". You may choose to implement this as follows:

- Besides TOU codes of "ON" and "OFF", create additional TOU codes called "ONEXCESS" and "OFFEXCESS"
- Define the TOU codes within the TOU group with sequences values as follows:

Time of Use	TOU Sequence
ON	1

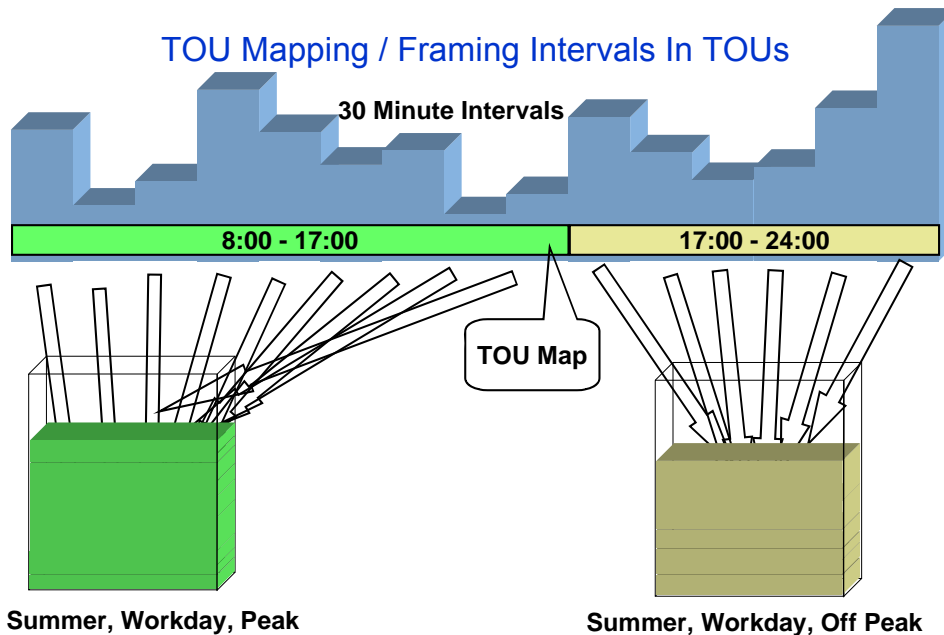
ONEXCESS	2
OFF	3
OFFEXCESS	4

- Your TOU map will only define time periods for ON and OFF, but you will have TOU prices for all four time of use values.
- Design your TOU pricing algorithm so that if the usage for a given TOU is within the contract limit, the price for this TOU is used. However, if the usage exceeds the contract limit, then it will find the TOU with the next highest sequence and use the price for that TOU.

Time of Use Mapping and Pricing

The following section describes the logic used by the system to map interval quantities to time of use codes using a TOU map, and subsequently apply prices to these mapped quantities.

At some point during billing, the system will take interval quantities linked to the service agreement and map them to time of use quantities based on a TOU Map.



For simple billing, this type of data manipulation is typically handled using an SQ rule. Keep in mind that SQ Rules are processed before applying the rate components and therefore they do not have knowledge of system breaks that may cause price proration.

Proration occurs when a given price covers a period smaller than the billing period AND the system does not have readings for the period covered by the price. When this occurs, the system will prorate the usage to apply the correct price. Refer to [Effective Dates & Price Proration](#) for more information about proration.

If you read carefully the price proration information, you'll see that if the service quantities are peak quantities, the quantities are not prorated but the charges are prorated. For other cases, the system prorates the service quantities.

When your interval quantities are peak quantities, the same should apply. The mapping should occur prior to price breaks, for example, using an SQ rule. Then, the TOU pricing rate components would apply prices to the mapped SQ quantities.

For non-peak interval quantities, readings exist for any price break situation. As a result, no service quantity proration is necessary. The mapping simply needs to occur after all price breaks are determined – at the rate component level.

Contents

- [TOU Pricing Rate Component](#)
- [TOU Map Used For Mapping & Pricing](#)
- [TOU Price Breaks](#)
- [Overriding Mapped Quantities](#)

TOU Pricing Rate Component

A special rate component exists to perform time of use mapping and time of use pricing. In fact, this rate component uses a calculation algorithm, so the algorithm can be written to perform whatever logic you need it to perform. The system is shipped with an algorithm, which performs mapping only and another algorithm, which may perform mapping and pricing, or just pricing.

- The mapping algorithms populate the read details collection by default because this collection can contain the quantities for each UOM/TOU/SQI along with a date range (This is important for [price breaks](#).) In addition, you may configure the algorithm to also produce SQ quantities, which would represent the total quantities for each UOM/TOU/SQI.
- The pricing algorithm may be configured to apply prices to the SQ collection (for measures peak scenarios) or the read details collection (for non-measures peak scenarios).

Refer to [Designing Your Time of Use Rate Components](#) for help in designing rate components of this type. Refer to [Setting Up TOU Pricing Rate Components](#) for more information about setting up this type of rate component.

The TOU pricing rate component may reference either a [TOU bill factor](#), which contains prices that differ for each time of use code or a regular bill factor, which will contain a single price to apply to all mapped quantities.

TOU Map Used For Mapping & Pricing

The TOU map used by the rate component for mapping is defined either directly on the rate component or it is linked to the service agreement.

- If ALL service agreements linked to the rate component's rate should use the same TOU map, then link the TOU Map to the rate component
- If some service agreements use different TOU maps than others, the TOU map must be linked to the service agreement. In this case, you must indicate a TOU Map Relationship Type on the rate component. The TOU map relationship type defines the [business role of TOU map](#).

Refer to [Sharing TOU Map Data](#) for more information.

TOU Price Breaks

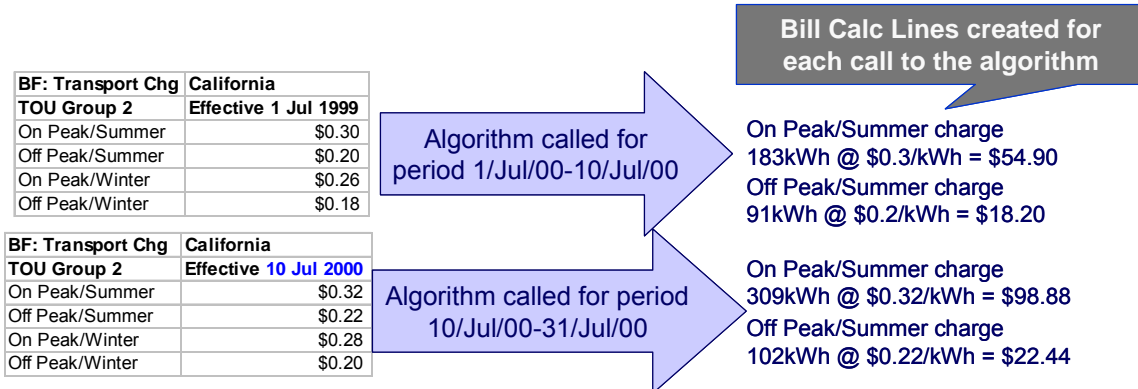
This section describes functionality related to price changes for time of use bill factor values. This logic is only applicable when the interval quantities do not measure peak quantities.

Contents

Algorithms are Called for Each TOU Price Break Algorithms Are Responsible for Storing Calculation Details

Algorithms are Called for Each TOU Price Break

Rate application logic understands how to determine price breaks for the bill factor referenced on your rate component and calls the calculation algorithm for each price break. The TOU pricing algorithm will be passed the appropriate date/time range to handle mapping and pricing and will produce separated bill calculation lines for each time period.



Single TOU Group. If your rate component references a TOU bill factor, the price breaks for the bill factor are determined using a single TOU group. If multiple TOU maps are applicable for the billing period and each refers to a different TOU group (via its [TOU map type](#)), the system uses the SA type's rate selection date to select the appropriate TOU group to use to determine the collection of prices to apply.

Algorithms Are Responsible for Storing Calculation Details

When the TOU mapping algorithms calculate the mapped quantities for a given date/time range, the results are stored in the [calculation/audit read details](#) collection on the bill segment. This collection enables the system to record the mapped quantities along with period represented by each quantity.

Your algorithms may opt to additionally capture the total quantity for each time of use in the SQ Details collection as an audit.

Read Details

OnSum	01/Jul/00-10/Jul/00	183kWh
OffSum	01/Jul/00-10/Jul/00	91kWh
OnSum	10/Jul/00-31/Jul/00	309kWh
OffSum	10/Jul/00-31/Jul/00	102kWh

Algorithms use the read detail collection to store interim UOM/TOU/SQI details with date and time

SQ Details

OnSum	492kWh
OffSum	193kWh

The base algorithms also provide the option of populating the SQ collection with the total quantity for each UOM/TOU/SQI

Note. Bill Factor prices have effective dates only. There is no effective time. When a price break occurs, the algorithm will determine the time for the price breaks in the same manner that billing time is determined. Refer to [Start and End Times for Billing](#) for more information.

The system provides one sample algorithm [RCTMS](#) that performs TOU mapping and another algorithm [RCTMPS](#) that performs TOU mapping and pricing or pricing alone. These algorithms support TOU price break logic as follows:

- The mapping functions for both algorithms populate the calculation/audit read details collection and allow you to set a parameter to indicate whether SQ entries should also be created.
- Based on a parameter, the pricing portion of [RCTMPS](#) applies the appropriate TOU prices to the mapped quantities stored in the calculation/audit read details collection.

If RCTMPS has been configured to perform mapping and pricing, it will perform the two functions as separate steps: first it maps the quantities and populates the read details, then it applies TOU prices to the read details. It does this for the following reasons:

- You may have a rate that requires manipulation of mapped quantities prior to applying prices. For example, perhaps you must add service quantities from a non-interval meter to your mapped quantities prior to applying prices.
- You may need to [override the mapped quantities](#) when recalculating a bill. The base algorithms will not re-map the quantities if the details were overridden. The pricing portion applies the prices to the overridden quantities.

Overriding Mapped Quantities

The mapping algorithms provided with the base product record the mapped quantities in the [calculation/audit read details](#) collection on the bill segment. When regenerating a bill segment, it is possible that circumstance requires you to override the details in this collection rather than allowing the system to map the interval quantities again.

Refer to [How To Override Service Quantities On A Specific Bill Segment](#) for more information.

The pricing functionality in the algorithms provided by the base product detect when mapped quantities have been overridden and apply the TOU prices to the overridden quantities.

Physical Attributes of TOU Map vs. Its Role

You will see, as you learn more about the design of TOU mapping, that there are two control tables that are important for defining TOU maps:

- TOU Map Type – this defines the physical attributes of the TOU Map
- TOU Map Relationship Type – this is needed for TOU maps that are linked to a service agreement. It defines the role that the TOU Map is playing for that service agreement. You can also think of this as defining the business purpose of the data.

Contents

[Physical Attributes of a TOU Map](#)

[Business Role of a TOU Map](#)

[The Business Role Defines Valid Physical Attributes for a TOU Map](#)

Physical Attributes of a TOU Map

When defining a TOU map, there are basic attributes, which need to be defined:

- The collection of possible time of use codes to which the data will be mapped. These are grouped together into a [TOU group](#).
- Minutes per Interval
- Is this data owned by a service agreement or is it common data?

These are physical attributes of the data and have nothing to do with the business purpose of the data. This information is defined on the TOU Map Type.

The [TOU map type](#) may also include an algorithm used to [automatically create data for the TOU map](#).

This is the same model used for interval profile data. Refer to [Physical Attributes of Interval Data](#) for more information.

Business Role of a TOU Map

The TOU Map Relationship Type is used for TOU maps that are linked to the service agreement. It indicates the “role” that a TOU Map is playing for a particular service agreement. The following are some examples of roles that a TOU Map may play:

- Map for Measured Demand
- Map for Contract Demand
- Map for Excess Demand
- Map for Reactive Energy

If your rate component requires a TOU map but the TOU map is not common to all service agreements for the rate, then the rate component must indicate a TOU Map Relationship Type. The rate component algorithm uses this information to find the appropriate TOU map linked to the service agreement. Refer to [TOU Maps Serve a Role for a Service Agreement](#) for more information.

This is the same model used for interval profile data. Refer to [Business Role of Interval Data](#) for more information.

The Business Role Defines Valid Physical Attributes for a TOU Map

To enable proper setup, you will need to define the valid [TOU map types](#) for each TOU map relationship type.

For more information about defining TOU maps and recording the time period data, refer to [Time of Use Mapping Background Topics](#).

Designing Interval Billing Options

Your interval billing options control how interval data may be maintained and billed for your customers with this capability.

For more information about interval billing, see [The Big Picture Of Interval Billing](#).

Warning! There are many ways to design your interval billing options. The flexibility of the system may add to the challenge of determining the best way to set up your control tables. In this section, we provide information to help you understand the ramifications of the various options. Before you set up your production data, we encourage you to gain an intuitive understanding of these options by using the system to prototype the alternatives.

The design of your interval billing options is an iterative process. Over time, you will develop intuitive skills that will allow you to skip some iterations. We recommend using the various steps in this section as a guide. When you are finished with this guide, you will be able to set up your interval billing options.

Contents

- [Designing Your Interval Billing Rate Options](#)
- [Designing Your Interval Billing Controls](#)
- [Designing Your Raw Data Options](#)
- [Designing Your Time of Use Options](#)
- [Designing Your Contract Options](#)
- [Designing Your SA Interval Billing Options](#)

Designing Your Interval Billing Rate Options

This section assumes that you are familiar with the Rates subsystem and especially with the section [Designing Rate Components](#). Your interval billing rate components will likely contain a combination of standard rate components and interval billing rate components. The focus in this section will be on designing rate components for interval billing and time of use mapping.

Contents

- [Designing Your Interval Rate Components](#)
- [Designing Your Billing Factors](#)

Designing Your Interval Rate Components

Although your rates will likely contain a combination of interval pricing and time of use pricing, we have separated these topics with respect to walking you through the control table setup.

To set up the rate components that support interval billing, we recommend using the following table as your guide.

RC Type	Bill Factor	Interval Profile Relationship Type	Algorithm	TOU Map Relationship Type

Obtain copies of existing bills that use the rate in question. If the rate is new, then write up EXACTLY how the information should appear on the customers' printed bills.

- Next, try to identify the components of the bill that are related to interval billing.
 - Are any of the lines produced as a result of direct billing? In other words, were interval prices applied to interval quantities?
 - Are any of the lines produced using interval quantities applied to a fixed price?

Start filling out the table with descriptions of what is needed to produce each line.

- Any line that is produced as a result of taking interval quantities and applying prices will be defined with the "Interval Pricing" rate component type.
 - This rate component type will require an Interval Profile Relationship Type (which is used to define the source data to price).
 - A bill factor will need to be defined. This bill factor may contain interval quantities or simple values.
 - An algorithm, which knows how to apply the prices to the quantities, is needed. The system provides the following rate component algorithms as examples.
- Rate Components related to time of use mapping and pricing are discussed below. Refer to [Designing Your Time Of Use Rate Components](#).

RC Type	Bill Factor	Interval Profile Relationship Type	Algorithm	TOU Map Relationship Type
Interval Pricing	Strike Price	Hedge Cover	Apply prices to quantities assuming a continuous curve	N/a
Interval Pricing	Spot Market Prices	Amount in Excess of Hedge Cover	Apply prices only to positive quantities assuming a continuous curve	N/a

Designing Your Billing Factors

To apply prices to interval quantities, bill factors are recommended. In fact, to be able to handle interval prices, you will need to use bill factors.

To design and set up your bill factors, refer to [Setting Up Bill Factors](#) for more information.

Designing Your Interval Billing Controls

Contents

[Designing Interval Profile Relationship Types](#)

[Designing Interval Profile Types](#)

Designing Interval Profile Relationship Types

Recall that the interval profile relationship type can be thought of as the role that interval data will play. You can also think of it as the business description of the interval data.

The starting point for designing your interval profile relationship types is the rate. First, identify the relationship types that your rate needs in order to produce a bill

Using the above rate as an example, two profile relationship types have been identified

Profile Relationship Type	Description
HEDGECSR	Hedge cover
OVRHEDGE	Amount over hedge cover.

Now, you must think about what other relationship types are required to produce the above relationship types needed by rates. In our above example, we will need to define a profile relationship type whose data is compared to the hedge cover data to produce the OVRHEDGE data. Let's assume that this new profile relationship type is related to measured demand.

Profile Relationship Type	Description
HEDGECSR	Hedge cover
OVRHEDGE	Amount over hedge cover.
MEASDMD	Measured Demand

In this manner, you will be able to successfully define your profile relationship types.

Designing Interval Profile Types

Now that you have your profile relationship types defined, you need to begin defining profile types. The interval profile type defines the physical attributes of the interval data.

The essential attributes of any profile type are the UOM/SQI, the minutes per interval and the associated algorithms. The easiest way to start defining your interval profile types is to start with the profile relationship type and determine what type of data each customer may have for the same profile relationship type.

Let's start with our profile relationship types and make some assumptions regarding the physical attributes of the data that a customer may have for each of these roles:

- Let's assume that a customer's demand may be measured in either 30-minute intervals or 60-minute intervals. In either case, the UOM is KW and SQI is not applicable. Let's also assume that the measured data may be interfaced from an external source or may be aggregated from interval register data.

- Let's assume that the hedge cover is **common**, but that these values may also be in 30-minute intervals or 60-minute intervals.
- Finally, let's assume that our spot market prices are at 60-minute intervals and that the algorithm, which rates uses to apply the interval prices, expects the "amount in excess of the hedge cover" to be in 60-minute intervals. This means that we need two different algorithms for our **OVRHEDGE** data. One algorithm is used for a customer with 60-minute intervals and simply subtracts the two curves and produces the resulting data. The second algorithm is used for a customer with 30-minute intervals. It performs two steps. It subtracts the hedge from the measured demand and adds together each resulting 30-minute pair to produce a 60-minute **OVRHEDGE** curve.

We will assume that all of our profile data will follow the same seasonal time shift as our base time zone. Refer to [Time Zone and Time Changes](#) and [Designing Your Time Options](#) for more information.

Prof. Rel. Type	Profile Type	Common/ SA Owned	UOM/ SQI	Min/ Interval	Seasonal Time Shift	Algorithm
MEASDMD	DMDKW60	SA Owned	KW	60	USShift	N/a
	DMDKW30	SA Owned	KW	30	USShift	N/a
	DMDKW60 AGG	SA Owned	KW	60	USShift	Aggregate interval register data
	DMDKW30 AGG	SA Owned	KW	30	USShift	Aggregate interval register data
HEDGECVR	HEDGE60	Common	KW/ HG	60	USShift	N/a
	HEDGE30	Common	KW/ HG	30	USShift	N/a
OVRHEDGE	OVRHG60	SA Owned	KW/ OVR	60	USShift	Compare MEASDMD to HEDGECVR to produce 60-minute over hedge curve
	OVRHG30	SA Owned	KW/ OVR	30	USShift	Compare MEASDMD to HEDGECVR and add 30-min interval pairs to produce 60-min 'over-hedge' curve

Note. The above table accomplishes two steps: defining profile types and defining the valid profile types for a profile relationship type. When setting up this data, you will need to define your profile types first and then link them to the appropriate profile relationship type.

As with most of your control table design, this is an iterative process. As you design your profile types, you may see the need for new profile relationship types. You may find that a profile type will be valid for more than one profile relationship type.

During your definition of the profile types, you may determine that new UOMs need to be defined. Refer to [Setting Up Unit Of Measure Codes](#) for more information about defining units of measure.

During your definition of the profile types, you may determine that new SQI values need to be defined. Refer to [Defining Service Quantity Identifiers](#) for more information about defining SQI values.

Note. Now that you have designed the control table values required to support your rate, we recommend that you set up [start options](#) for your SA types to assist a CSR in setting up a customer for this rate.

All of the above steps will need to be repeated for each interval billing rate that your company offers. For each rate, you need to define your rate components, your billing factors, your profile relationship types and then the appropriate profile types.

Designing Your Raw Data Options

Note. If your company uses an external system for collecting, adjusting and aggregating raw interval data, then you may skip this section.

The term “Channel” is often used for devices that may store interval data. A physical channel may hold interval data or index readings.

- Interval channels will contain collections of interval data
- Index channels are a collection of time of use registers and readings for these registers are no different than standard register readings

In either case, a “channel” is represented in the system by a register and a meter will represent a logical grouping of channels (index or interval). Refer to [The Structure Of A Meter](#) for more information.

Contents

[Designing Your Interval Register Types](#)
[Designing Your Raw Data TOU Groups](#)
[Designing Your Meter Configuration Types](#)
[Designing Your Meter Types](#)

Designing Your Interval Register Types

Each register, which will record raw interval data will require an interval register type.

The essential attributes of an interval register type are the UOM/SQI, the minutes per interval and any validation algorithms.

- For our examples, let’s assume that raw interval data may be measured in kilowatts or megawatts and may be measured in either 30-minute intervals or 60-minute intervals.
- For our examples, let’s assume that the data may be standard recorded data or it may be calculated excess data. We will use an SQI to further label the excess demand.
- For each register type in our example, we will use a validation algorithm, which will verify that the intervals are correct according to the interval size on the interval register type.

We will assume that all of our raw interval data will follow the same seasonal time shift as our base time zone. Refer to [Time Zone and Time Changes](#) and [Designing Your Time Options](#) for more information.

Interval Register Type	UOM/ SQI	Min/ Interval	Seasonal Time Shift	Validation Algorithm
KW60	KW	60	USShift	Validate interval size
KW60EXC	KW/EXCESS	60	USShift	Validate interval size
KW30	KW	30	USShift	Validate interval size
KW30EXC	KW/EXCESS	30	USShift	Validate interval size
MW60	MW	60	USShift	Validate interval size
MW60EXC	MW/EXCESS	60	USShift	Validate interval size
MW30	MW	30	USShift	Validate interval size
MW30EXC	MW/EXCESS	30	USShift	Validate interval size

We expect that you will need more algorithms than we supply. Your algorithms will be based on any number of factors. Be aware that new algorithms may require programming. See [How To Add A New Algorithm](#) for more information.

Designing Your Raw Data TOU Groups

In addition to designing TOU groups to be used by time of use maps, you may also set up [TOU groups](#) to define a valid collection of time of use values for a given meter configuration.

Designing Your Meter Configuration Types

You must define a meter configuration type for each valid collection of interval and index registers. For meter configuration types with index registers, you may indicate a TOU group, which contains the valid collection of time of use codes for the registers. For each interval register, you must check the Interval switch and indicate the appropriate Interval Register Type. A meter configuration may contain

- A single interval register
- Multiple interval registers
- A collection of index registers
- A combination of interval registers and index registers

Refer to [Setting Up Meter Configuration Types](#) for more information.

Designing Your Meter Types

You will need to set up appropriate [meter type codes](#) for the meters, which will be used to define or group index and interval channels. You must turn on the **Allow Interval Registers** switch for any meter type used by a meter, which will contain interval registers. Indicate the valid meter configuration types for this meter type.

Note. If a meter type will only be used for meters linked to index channels, nothing special is required. Their behavior is similar to standard non-interval meters.

Designing Your Time of Use Options

Contents

- [Designing Your Time Of Use Rate Components](#)
- [Designing Your Time of Use Codes](#)
- [Designing Your TOU Groups](#)
- [Designing Your TOU Map Relationship Types](#)
- [Designing Your TOU Map Types](#)
- [Designing Your TOU Map Templates](#)

Designing Your Time Of Use Rate Components

As you know from the rates chapter, the system can handle billing of quantities with different time of use periods using the SQ rate component type. Refer to [How To Set Up Service Quantity Rate Components](#) for more information. A more sophisticated rate component type is available, which can map interval profile data into time of use definitions and then apply prices based on time of use to produce multiple bill lines.

Obtain copies of existing bills that use the rate in question. If the rate is new, then write up EXACTLY how the information should appear on the customers' printed bills.

- Identify all the lines that represent charges for individual time of use periods.
- Determine how the quantities for the time of use periods are calculated. For example, what is the source data? Which time of use map is used to define the time periods? A **TOU Pricing** rate component type may be used to map each curve to its time of use quantities, apply time of use prices to mapped quantities or do both mapping and pricing. It all depends on how your algorithm is written.
 - In order to perform mapping, this rate component type needs to know where to get the interval data. You must indicate an Interval Profile Relationship Type (which is used to define the source data to map). Refer to [Billable Interval Quantities for a Service Agreement](#) for more information.
 - In order to perform mapping or pricing, this rate component type needs to know the TOU map. If the TOU map is common to ALL service agreements for the rate, you must indicate the appropriate TOU map. If the TOU map differs for different service agreements, you must indicate a TOU Map Relationship Type. The algorithms use this to find the appropriate [TOU map linked to the service agreement](#).
 - In order to perform pricing, this rate component type requires either a regular [bill factor](#) or a [TOU bill factor](#). The bill factor may contain the collection of prices directly, or may indicate that the prices are customer specific and can be found as contract quantities for the service agreement.
 - A calculation algorithm to map and / or price the quantities is needed. For mapping, this algorithm must apply the TOU map to the quantity curve and produce entries in the [calculation/audit read details](#) collection. For pricing, the algorithm will need to use the appropriate bill factor to apply prices to the quantities in the read details collection. If this bill factor is a TOU bill factor, a separate price exists for each time of use.

For our example, let's assume that curves exist for both active demand and reactive demand and assume each curve is mapped with a different TOU map. In addition, let's assume that excess demand used during a specific curtailment period will be priced at a different rate.

- For active demand, we have four time periods: on-peak winter, off-peak winter, on-peak summer and off-peak summer.

- For reactive demand, there are no charges in the winter so we have only two time periods: on-peak summer, off-peak summer. This TOU map is common to ALL service agreements for the rate.
- The curtailment charge will only appear if there is a curtailment event and the customer used more than their maximum demand defined for the period.

RC Type	UOM/ TOU	Bill Factor	Interval Profile Relationship Type	Algorithm	TOU Map Relationship Type	TOU Map ID
TOU Pricing	N/a	DMDTOU	Active Demand	Map quantities to time of use periods; apply prices	Active Demand Map	
TOU Pricing	N/a	RCTVTOU	Reactive Demand	Map quantities to time of use periods; apply prices		13849374
TOU Pricing	N/a	CURTAL	Curtailment charge	Map quantities to time of use periods, compare with maximum demand; apply prices to excess	Curtailment	

Resulting bill calculation lines from the above TOU mappings (assuming this bill crosses the summer and winter seasons and a curtailment event occurred).

Charge for active kW On peak winter nn kW @ \$0.0353/KW: \$nn.nn
Charge for active kW Off peak winter nn kW @ \$0.0298/KW: \$nn.nn
Charge for active kW On peak summer nn kW @ \$0.0483/KW: \$nn.nn
Charge for active kW Off peak summer nn kW @ \$0.0327/KW: \$nn.nn
Charge for reactive kV On peak summer nn kV @ \$0.0293/Kv: \$nn.nn
Charge for reactive kV Off peak summer nn kV @ \$0.0231/Kv: \$nn.nn
Charge for curtailment excess nn kW @ \$0.0593/kW: \$nn.nn

Once you have your rate components designed, you will be able to design the other control tables needed to set up your time of use billing customer.

Note. Your time of use rate also requires interval profile relationship types. Refer to [Designing Interval Profile Relationship Types](#) and to [Designing Interval Profile Types](#).

Designing Your Time of Use Codes

The next most logical step in designing your time of use mapping controls is to define your time of use codes. To do this, look at the time of use periods to which your usage needs to be mapped. These values will likely correspond to the time of use quantities that your rate bills for. (Although it's possible that you are not billing for every time of use period.) Be sure to consider special time periods in your contracts such as holidays, curtailment days and interruption days.

Building on the above example, we have the following time of use codes defined:

Time of Use	Description
ONWIN	On Peak Winter
OFFWIN	Off Peak Winter
ONSUM	On Peak Summer
OFFSUM	Off Peak Summer
CURTAIL	Curtailment period

For more information about time of use, refer to [UOM versus TOU versus SQL](#).

Designing Your TOU Groups

To further aid in designing time of use mapping, the TOU Group enables you to group together all the time of use codes that are used in a single map. In our above example, we will have two TOU Groups because the active and reactive energy quantities are mapped to different sets of time periods. You must also decide if you want to use a sequence number to define the relative order of a TOU within a TOU group.

Refer to [Grouping of TOU Codes](#) for more information.

TOU Group	Description	Time of Use	TOU Sequence
4PARTS	Group for a 4-part map including On and Off Peak for Winter and Summer.	ONWIN	0
		OFFWIN	0
		ONSUM	0
		OFFSUM	0
2PARTS	Group for a 2-part map including On and Off Peak for Summer	ONSUM	0
		OFFSUM	0
CURTAIL	Curtailment group, contains only one time of use	CURTAIL	0

Designing Your TOU Map Relationship Types

The starting point for designing your TOU map relationship types is the rate. First, identify the relationship types that your rate needs in order to successfully map quantities.

Using the above rate as an example, only two TOU map relationship types are required because the Reactive Demand map is linked directly to the rate component. However, let's add a relationship type for reactive demand also in case it is needed in the future.

TOU Map Relationship Type	Description
ACTVDMD	Active Demand
RACTVDMD	Reactive Demand
CURTAIL	Curtailment

Now you must think of other types of maps that may need to be linked to the service agreement in order to successfully produce a bill. For example, are there any data derivation algorithms that require a TOU Map? If so, you need to define an appropriate TOU map relationship type to define the role for this map.

Designing Your TOU Map Types

Now that you have your TOU map relationship types and your TOU groups defined, you can begin defining TOU map types. Recall that the TOU map type defines the physical attributes of the TOU map.

The essential attributes of any profile type are the TOU Group and the minutes per interval and the associated TOU map data creation algorithms.

The easiest way to start defining your TOU map types is to start with the TOU map relationship type and determine what type of TOU map each customer may have for the same TOU map relationship type.

Let's start with our TOU map relationship types and make some assumptions regarding the physical attributes of the data that a customer may have for each of these roles:

- For Active demand, the quantities may be recorded in 15-minute or 30-minute intervals. To facilitate efficiency in processing the data, we recommend that the TOU map data is stored in the same minutes per interval as the interval data being mapped. We'll assume that this is a **common** TOU map type.
- For Reactive demand, the quantities are recorded in 60-minute intervals. This is linked directly to the rate and is therefore a **common** TOU map type.
- For curtailment, we will assume that the override map will be in 15-minute, 30-minute or 60-minute intervals, based on the interval size of the actual demand data. In addition, we require a TOU map creation algorithm. This algorithm will produce a data for the curtailment TOU map based on the existence of a contract option event. It will generate the time of use data based on a [TOU map template](#). Refer to [Contract Option Background Topics](#) for more information. This will be an **SA owned** TOU map type.

Note. We said above that we don't need a TOU map relationship type for Reactive Demand for our rate example. However, TOU map types must be defined. The table below includes the TOU map relationship type because we said we would add it just in case.

We will assume that all of our TOU data will not follow any seasonal time shifting. It will always be displayed in standard time. Refer to [Time Zone and Time Changes](#) and [Designing Your Time Options](#) for more information.

TOU Map Rel. Type	TOU Map Type	Common/ SA Owned	TOU Group	Min/ Interval	Seasonal Time Shift	Creation Algorithm
ACTVDMD	4PART15	Common	4PART	15	NoShift	N/a

	4PART30	Common	4PART	30	NoShift	N/a
RACTVDMD	2PART60	Common	2PART	60	NoShift	N/a
CURTAIL	CURT15	SA Owned	CURTAIL	15	NoShift	Create based on events for 'curtail' contract option type.
	CURT30	SA Owned	CURTAIL	30	NoShift	Create based on events for 'curtail' contract option type.
	CURT60	SA Owned	CURTAIL	60	NoShift	Create based on events for 'curtail' contract option type.

Now, you're ready to set up your interval billing options.

Note. The above table accomplishes two steps: defining TOU map types and defining the valid TOU map types for a TOU map relationship type. When setting up this data, you will need to define your TOU map types first and then link them to the appropriate TOU map relationship type.

Designing Your TOU Map Templates

In order to help your users to create and maintain data for TOU maps, you may define TOU map templates, which can be used to generate data for a TOU map. The templates may be used to define standard data for a TOU map as well as data for special periods, such as interruptions and holidays. The TOU map templates will reference a TOU map type. The system will use the TOU group and the minutes per interval from the map type to verify the setup of the template.

For our example, let us first design templates for the 15 minutes active demand TOU map type. Let's assume:

- On Peak for both winter and summer is from Monday to Friday, from 9am to 5pm, inclusive
- Off Peak for both winter and summer is all day Saturday and Sunday and from Monday through Friday, Off Peak is from 12AM to 8:45AM, inclusive and from 5:15pm to 11:45pm inclusive.

We'll worry about the season definitions for winter and summer later.

In addition, let's design the curtailment map template. Assume:

- Curtailment covers 9am to 5pm inclusive on the given curtailment day.

Contents

[Designing Daily Templates](#)
[Designing Weekly Templates](#)
[Designing Holiday Templates](#)
[Designing Calendar Templates](#)

Designing Daily Templates

We need to define four daily templates: one for summer weekdays, one for summer weekend days, one for winter weekdays and one for winter weekend days.

Before defining the template, take note of some points about defining template components:

- For our winter weekday, we can define the off peak period with one entry covering 5:15pm through 8:45am using the Start and End Sequence Numbers to indicate that the end of the time period is on the second day.
- The TOU map generator will create intervals for the first period AFTER the start time up to and including the end time. As a result, our Start Time must be the interval before the first interval covered by the time of use code.

TOU Map Template	TOU Map Type	Template Type	Start Seq	Start Time	End Seq	End Time	Time of Use
15MinWntr Day	4PART15	Daily	1	8:45AM	1	5PM	ONWIN
			1	5PM	2	8:45AM	OFFWIN
15MinWntr Wknd	4PART15	Daily	1	12AM	2	12AM	OFFWIN
15MinSmr Day	4PART15	Daily	1	8:45AM	1	5PM	ONSUM
			1	5PM	2	8:45AM	OFFSUM
15MinSmr Wknd	4PART15	Daily	1	12AM	2	12AM	OFFSUM
15MinCurtail	CURT15	Daily	1	8:45AM	1	5PM	CURTAL

Designing Weekly Templates

Now that we have our daily templates defined, we can define our weekly templates. For our example, we will need two weekly templates: one for a typical winter week and one for a typical summer week. For weekly templates, you must indicate the start day of the week.

We do not need a weekly template for the curtailment option.

TOU Map Template	TOU Map Type	Template Type	Week Start	Start Seq	Start Day	Start Time	End Seq	End Day	End Time	Template
15MinWntrWk	4PART15	Weekly	Mon	1	Mon	12AM	1	Sat	12AM	15MinWntr Day
				1	Sat	12AM	2	Mon	12AM	15MinWntr Wknd
15MinSmrWk	4PART15	Weekly	Mon	1	Mon	12AM	1	Sat	12AM	15MinSmrDay
				1	Sat	12AM	2	Mon	12AM	15MinSmr Wknd

Finally, we need to decide if we want to create a Calendar template for our example. The decision of whether or not to create a calendar template will be based on a few factors:

- How many different seasons exist in the TOU map, where each season has different weekly templates?
- How do your holidays behave?

First, let's consider holiday behavior.

Designing Holiday Templates

When designing your TOU map templates, a special consideration should be made for holidays that occur throughout the year. Look at your TOU map contracts and ask the following questions:

- Are special prices applicable on holidays? Do these special prices require new time of use codes?
- If holidays do not get special prices, does the daily template look like the daily template for other days for this TOU map, for example like a weekend day?
- Do your holidays all have the same time period definitions as each other?

For our example, let's assume that our holidays use the Off Peak time of use for the season that it is in. In other words, winter holidays will use OFFWIN for every interval in that day and summer holidays will use OFFSUM for every interval in that day. As a result, we will not need to create a new template just for holidays. Instead, we can use the **15MinWntrWknd** daily template for winter holidays and the **15MinSmrWknd** daily template for summer holidays.

Designing Calendar Templates

The calendar template is used to define daily and weekly templates for month and day ranges. The map components may represent dates within a single year or may cross into the following year.

Remember that a main reason for creating templates is for use on the [TOU Map Generation](#) page to create data for a TOU map. On that page, you will specify the dates you want to generate data for. In addition you must keep in mind that you can specify one holiday template for the time period you are generating.

In our example, we have two seasons and our holiday template is different in each season. Creating a calendar template to define the seasons will not help us, because on the TOU map generation page, we wouldn't be able to indicate a single holiday template for the whole year. For our example, rather than creating a calendar template, we could simply go to the TOU map generation page and indicate:

- 1st Nov <year> through 30th April <year>, use the **15MinWntrWk** template. For holiday overrides, use the **15MinWntrWknd** daily template
- 1st May <year> through 31st October <year>, use the **15MinSmrWk** template. For holiday overrides, use the **15MinSmrWknd** daily template

In fact, in our example, even if our holiday template would not vary by season, one would have to weigh the advantage of creating a calendar template to define 2 seasons vs. running the TOU map generation twice. If you have a very small number of customers using the same templates, then a calendar template may not be worth creating. However, if you have many customers using the same template, it may be worth it.

Here is a possible guideline to follow for deciding whether or not to create calendar templates:

- If the same holiday template may be used for all holidays, and your customer's contracted time period definitions contain more than one season, then creating a calendar to define the seasons is recommended.
- If the holiday template changes throughout the year based on the season, then you must weigh the advantages of your possibilities:

- You may just create weekly templates for each season and define the dates and the holiday template when using the TOU map generation page. This option is recommended when the templates apply to a very small number of TOU maps. The time taken to generate the data based on the weekly templates would be less than the time you would have spent on defining the calendar.
- You may define your calendar to explicitly indicate your seasons and your holidays. Then when generating the data for this template, you would not indicate an override holiday template. This option is recommended when the template applies to a larger number of customers. The time taken to define the calendar would be less than that spent on generating multiple seasons for each TOU map.

Let's go ahead and design a calendar template using our weekly templates to understand the setup. For this purpose, let's assume our holiday template would be the same all year round.

Assume the seasons are defined as follows: Winter is 1st Nov through 30th April, Summer is 1st May through 31st October. This template shows that the summer schedule includes the interval 12AM on 01, November. This means that summer's last interval is the one that covers from 11:46PM on the 31st of October through 12AM on the 1st of November. The first interval in winter is 12:15AM on the 1st of November, which covers the period from 12:01AM through 12:15AM. The similar setup is true for the end of winter / beginning of summer.

TOU Map Template	TOU Map Type	Template Type	Start Seq	Start Date	Start Time	End Seq	End Date	End Time	Template
15MinCIndr	4PART 15	Calendar	1	01May	12AM	1	01Nov	12AM	15MinSmr Wk
			1	01Nov	12AM	2	01May	12AM	15MinWntr Wk

Designing Your Contract Options

Contents

[Designing Your Contract Option Types](#)

[Designing Your Contract Option Event Types](#)

Designing Your Contract Option Types

Contract options are used for service agreements, which may be subject to special overrides or alterations in their rate for certain temporary periods. Contract options are used to define the possible special situations and to record the actual override events. Your contract option type is a high level categorization of the possible special situation applicable to a customer or group of customers.

To design your contract option types, you will need to review each rate for your interval customers and determine whether or not there are special options. Some examples of special options are:

- Interruptions. Perhaps you warn your customers of interruption periods and any usage incurred during that period will be subject to special prices.
- Curtailment. Perhaps you define periods where you ask a customer to shed their load below a certain subscribed demand and any demand, which exceeds this subscribed demand during that period, will be subject to a special charge.

Next, must determine whether or not a given option behaves differently under different circumstances. For example, perhaps you have different types of interruptions. If these different types of interruptions are applicable to the same customer, then you will require a single contract option type and multiple contract option event types for that option type. If these different types of interruptions are not applicable to the same customer, then you would probably want to define separate contract option type values.

You need to also consider whether or not contract option events for this type of contract option may overlap in their effective periods. If you do allow overlap, then you must be sure that any algorithm, which may process the contract option events, must know how to process the overlaps.

Your next step is to determine whether or not characteristic values will be needed for each contract option of this type. The characteristic values are available for use by the algorithms, when processing the contract options. For example, perhaps subscribed demand is defined at the contract option level. On the contract option type, you must define the possible characteristic types for contract options of this type. You may also define a default value, if applicable.

Finally, you will need to determine whether or not you wish to create validation algorithms to validate your contract option event data. Algorithms may be created to be executed upon add or change of the data, where the status is **Pending**, **Frozen** or **Canceled**.

For our purposes, let us assume that we have only two contract option types: Interruption and Curtailment. Neither one allows overlap. The Curtailment option type will define subscribed demand as a required characteristic and will define a validation algorithm to ensure that the value of the demand falls within an appropriate range. This algorithm program will be executed for pending and frozen events.

Note. Because there are different algorithm entities, more than one algorithm type and algorithm are required, but both algorithm types may use the same program.

We will assume that both contract option types will follow the same seasonal time shift as our base time zone. Refer to [Time Zone and Time Changes](#) and [Designing Your Time Options](#) for more information.

Contract Option Type	Descr	Allow Overlap	Seasonal Time Shift	Char Type	Algorithm
INTERRUPT	Interruptions	No	USShift	N/a	N/a
CURTAIL	Curtailments	No	USShift	SubscrDmd, required, no default value	Pending: <i>SbscrDmdPV</i> (subscribed demand pending validation) Frozen : <i>SbscrDmdFV</i> (subscribed demand frozen validation) Canceled: n/a

Note. Once you have the contract option types required to support your rate and derivation algorithms, we recommend that you set up Start Options for your SA Types to assist a CSR in setting up a customer for this rate. Refer to [Designing Your Start Options](#).

Designing Your Contract Option Event Types

When you designed your contract option types, you were already considering the possible contract option event types. You will need to define a different contract option event type for each different type of event that may occur for the same contract option type.

Your next step is to determine whether or not characteristic values will be needed for each contract option event of this type. The characteristic values are available for use by the algorithms, when processing the contract option events. On the contract option event type, you must define the possible characteristic types for contract option events of this type. You may also define a default value, if applicable.

For our example, we will define two types of interruptions, which may be applicable for customers on the same rate. We will define only one type of curtailment event. None of the event types require characteristic values.

Contract Option Event Type	Descr	Contract Option Type	Char Type
InterruptA	Interruptions, type A	Interrupt	N/a
InterruptB	Interruptions, type B	Interrupt	N/a
Curtail	Curtailments	Curtail	N/a

Designing Your SA Interval Billing Options

Contents

[Designing Your SA Types](#)
[Designing Your Start Options](#)

Designing Your SA Types

Your interval service agreements will require special data to be set up, such as special rates, interval profiles, TOU maps and contract options. In addition, it will need to define a cutoff time and start day option. The system requires you to indicate a special role of *Interval* for SA types defined for interval billing service agreements.

Refer to [Defining Service Agreement Types](#) to help you design the standard fields required for all SA Types. Additionally, for interval SA Types, you will need to define the valid Profile Relationship Types, the valid Map Relationship Types and the valid Contract Option Types for this SA Type.

Let's assume that we set up only one SA Type for all interval billing customers. As a result, all the relationship types defined in our sample are valid.

CIS Division / SA Type	Special Role	Profile Relationship Type	Map Relationship Type	Contract Option Type
CA / E-INTBIL	<i>Interval</i>	HEDGEVR	ACTVDMD	INTERRUPT
		OVRHEDGE	CURTAL	CURTAL
		MEASDMD		
		ACTVDMD		
		RACTVDMD		

Note that this setup does not provide any link between the profile relationship types, the map relationship types and the contract option types. These are just a list of valid entries for the SA Type.

Note. Although we created a TOU map relationship type for **RACTVDMD** above, it is not needed for the TOU mapping rate so it is not linked to the SA Type as a valid value.

Refer to [Setting Up SA Types](#) for more information.

Designing Your Start Options

Once you have your rates defined and your SA Type defined, you should design Start Options to aid your customer service representatives in setting up these service agreements.

You will first need to design your [start options](#) with regard to standard SA information. Note that there may be other setup required before you can add your interval billing start options:

- To link **common** profiles to a start option you will need to create the appropriate [profile](#) first.
- To link **common** TOU maps to a start option you will need to create the appropriate [TOU map](#) first. This is only applicable to common maps that should be linked to the service agreement. In our example, it applies to the active demand map. The reactive demand map is linked directly to the rate component so no start option information is needed.
- To link shared contract options to a start option you will need to create the appropriate [contract option](#) first.

Note that when designing our rate components above, we did not give names to their rate schedules. Let's call the interval pricing rate schedule "INTPRC" and let's call the TOU Map rate schedule "INTTOU".

Recall that the TOU Map rate component required profile relationship types that we did not define above. We will assume that profile relationship types are defined with the same names as the map relationship types. We will also assume that the profile types define the same interval size as the map types.

You may also use Start Options to define Cutoff Time and Start Day Option, if they are different from the values defined on the installation record. Let's assume that our TOU mapping customers use a different Cutoff Time from the installation record.

The following table shows the start option definitions with the rates and the cutoff time information.

CIS Division / SA Type	Start Option	Rate Schedule	Cutoff Time	Start Day Option
CA / E-INTBIL	Interval Pricing Hedge – 30MIN	INTPRC		
	Interval Pricing Hedge – 60 Min	INTPRC		
	TOU Map – option 1	INTTOU	2:00 AM	Current
	TOU Map –	INTTOU	2:00 AM	Current

	option 2			
--	----------	--	--	--

The following table shows the profile related start options.

CIS Division / SA Type	Start Option	Rate Schedule	Profile Relationship Type	Profile Type	Profile (Profile Type)
CA / E-INTBIL	Interval Pricing Hedge – 30MIN	INTPRC	HEDGEVR		12859302 (HEDGE30)
			OVRHEDGE	OVRHG30	
			MEASDMD	DMDKW30	
	Interval Pricing Hedge – 60 Min	INTPRC	HEDGEVR		4922018 (HEDGE60)
			OVRHEDGE	OVRHG60	
			MEASDMD	DMDKW60	
	TOU Map – option 1	INTTOU	ACTVDMD	ACTV30	
			RACTVDMD	RACTV30	
	TOU Map – option 2	INTTOU	ACTVDMD	ACTV15	
			RACTDMD	RACTV60	

The following table shows the TOU map related start options.

CIS Division / SA Type	Start Option	Rate Schedule	Map Relationship Type	Map Type	TOU Map (Map Type)
CA / E-INTBIL	TOU Map – option 1; no curtailment	INTTOU	ACTVDMD		12859302 (4PART30)
	TOU Map – option 2; no curtailment	INTTOU	ACTVDMD		52829947 (4PART15)
CA / E-INTBIL	TOU Map – option 3; curtailment	INTTOU	ACTVDMD		12859302 (4PART15)
			CURTAL	CURT15	

The following table shows the contract option related start options.

CIS Division / SA Type	Start Option	Rate Schedule	Contract Option Type	Contract Option
CA / E-INTBIL	TOU Map – option 3; curtailment	INTTOU		48399239048 (CURT15)

Let's summarize what these tables define:

- The INTPRC rate requires 2 **SA Owned** profiles and one **Common** profile. The **Common** Profile needs to be created and linked to the start option. (For clarification, the Profile Type used for the profile is displayed in parentheses.) For the **SA Owned** profile, you need to indicate the profile type.
- There are two possible configurations for the rate INTPRC. As a result, there are two start options, each with the correct configuration defined.
- For the rate INTTOU, the SA will need profiles, TOU maps and, if they use the curtailment option, contract options.
 - The profiles will be **SA owned**, so only the profile relationship type and profile types are indicated.
 - For the TOU maps, the active demand maps is **common** so the actual map is indicated, whereas the curtailment map will be **SA owned** so only the map type is indicated. (The reactive demand map is linked directly to the rate component.)
 - The curtailment option will be shared by more than one service agreement, so the contract option itself is linked to the start option.

Refer to [Setting Up Start Options](#) to learn how to enter this information.

Setting Up Interval Billing Options

Contents

[Setting Up Interval Billing Control Tables](#)
[Setting Up Channel Control Tables](#)
[Setting Up Time of Use Billing Control Tables](#)
[Setting Up Contract Option Control Tables](#)

Setting Up Interval Billing Control Tables

Contents

[Setting Up Profile Relationship Types](#)
[Setting Up Interval Profile Types](#)
[Setting Up Interval Pricing Rate Components](#)

Setting Up Profile Relationship Types

Profile Relationship Types define the role that a set of interval data will serve for a customer. Open **Admin Menu, Interval Profile Rel Type** to define your profile relationship types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **Interval Profile Relationship Type** ID and **Description** for the profile relationship type.

Enter the **Interval Profile Type** collection. This is a list of valid [interval profile types](#) whose collections of data can serve the role defined by the profile relationship type.

Note, to aid in setup, the interval profile type is not a required field on this user interface. This will help you to define your high level interval relationship types first and then define the more detailed profile types. The valid profile types will need to be linked to the appropriate relationship types prior to creating service agreement interval profiles.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_INTV_PFRELT](#).

Setting Up Interval Profile Types

Interval Profile Types define values common to interval profiles of the same type. Open **Admin Menu, Interval Profile Type** to define your interval profile types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **Interval Profile Type ID** and **Description** for the interval profile type.

Indicate whether interval profiles of this type are **SA Owned** or **Common** by entering the appropriate value in the interval profile **Sub Type**.

The **Creation Priority** will be used by the [Interval Profile Derivation Process](#) to determine the order in which the data for profiles linked to an SA should be derived. The values range from **10**, being the highest priority to **90** being the lowest priority.

Note. The values for this field are customizable using the Lookup table. This field name is CRE_PRIO_FLG.

Enter the **Unit of Measure (UOM)** that data values stored for interval profiles of this type are captured in.

If a service quantity identifier is needed to further qualify data stored for interval profiles of this type, enter a valid **SQ Identifier**.

Enter the **Minutes per Interval** to define the number of minutes expected in between each row of data collected for interval profiles of this type.

If the [installation](#) record indicates that [seasonal time shift](#) is required, then you must enter the appropriate **Seasonal Time Shift** record applicable for the interval data. Please take special note of the issue described in the [Evenly Sized Intervals](#) section.

The grid contains **Algorithms** that may be used to create or validate interval data for profiles of this type. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (see the table that follows for a description of all possible events).
- Specify the **Sequence** number and **Algorithm** for each system event. You can set the **Sequence Number** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

We expect that you will need more algorithms than we supply. Your algorithms will be based on any number of factors. Be aware that new algorithms may require programming. See [How To Add A New Algorithm](#) for more information.

The following table describes each **System Event**.

System Event	Description
<i>Interval Data Creation</i>	These types of algorithms are used to derive interval data for a profile. Click here to see the algorithm types available for this system event.
<i>Interval Data Validation</i>	These types of algorithms are used to validate profile data . Click here to see the algorithm types available for this system event.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_INTV_PF_TYP](#).

Setting Up Interval Pricing Rate Components

This section describes how to define your interval pricing rate components.

Define Rate Schedule, Rate Version and other Rate Components. Before creating your interval pricing rate component, you must set up a [rate schedule](#). You will also need to set up a Rate Version and possibly set up other types of rate components. Refer to [Defining Rate Versions](#) and [Defining Rate Components](#) for more information.

When you are ready to set up your interval pricing rate component, open **Main Menu, Rates, Rate Version**. Choose the rate version that this component should belong to. Use the Rate Version context menu and select Go to Rate Component + to add a new rate component.

Description of Page

Refer to [Rate Component - Main Information](#) for information about the common fields on this page. When defining an Interval Pricing rate component additional fields become available to you. The following information will help you to set up your Interval Pricing rate components.

Indicate whether or not this is **FCPO**.

Select a **Value Type** of **Unit Rate**. This field will be gray when the rate component is referenced on another rate component. The **Value Source** will most likely be **Billing Factor**. Your bill factor will likely be one with a type of **Interval**.

Refer to [Defining Interval Values](#) for more information about setting up bill factors with interval prices.

Indicate whether or not this rate component is **Seasonal**. Refer to [Rate Component - Main Information](#) for more information about seasonal rate components.

The **Error if No Value** field is available for you to use in the calculation algorithm.

Indicate the **Calc Algorithm** that the system will use to calculate the bill line that this rate component produces.

- The system provides an Algorithm Type that is available for use here. It is called [RCIPRS](#).

If this algorithm does not provide you with the logic you require, you will need to create a new algorithm (refer to [Setting Up Algorithms](#)). The above existing algorithm should be used as a sample if you have to write a new algorithm type.

Note. The calculation algorithm's main purpose is to create bill calculation lines. However, the algorithm may populate other information for the bill, for example, it may add to the SQ or register read collection or it may overwrite the description on bill.

Indicate the **Audit Algorithm** to be used when a CSR wants to drill down into the details of a bill line that was calculated using this Rate Component.

- The system provides an algorithm type that is available for use here. It is called [RCIPRS-ADT](#).

If this algorithm does not provide you with the logic you require, you will need to create a new algorithm (refer to [Setting Up Algorithms](#)). The above existing algorithm should be used as a sample if you have to write a new algorithm type.

Note. The audit algorithm should produce the same results as the Calc Algorithm. They are separated because they have different responsibilities. For example, the Calc Algorithm should produce bill lines, but the Audit Algorithm should not. They share common logic related to accessing and processing the appropriate interval data records. As a result, it is recommended that these two programs share a common code which accesses and processes the interval data. The above algorithms provided by the system behave this way and should be used as samples.

Refer to [Interval Billing Calculation Details](#) to understand where a CSR may view the calculation details for a bill calc line, using this algorithm.

Indicate the **Interval Profile Rel Type**. This indicates to the system the profile, linked to the SA, which contains the interval quantities to be processed. Refer to [Business Role of Interval Data](#) and [Setting Up Profile Relationship Types](#) for more information.

Turn on **GL Statistical Quantity** if GL journal lines generated for this rate component should also contain the service quantity amount as a statistical quantity. You would use this option if you keep track of both dollar amounts and consumption units in your general ledger.

Enter the verbiage to appear on the customer's bill in **Description On Bill** and turn on the **Print** switch. Refer to [Rate Version - Bill Print Info](#) for more information about these fields.

Move to the [Rate Component - GL Distribution](#) window to define how to book moneys associated with this rate component in the general ledger.

Setting Up Channel Control Tables

Contents

- [Setting Up Interval Register Types](#)
- [Setting Up Meter Configuration Types for Channels](#)
- [Setting Up Meter Types](#)

Setting Up Interval Register Types

Refer to [The Big Picture of Raw Data Collection and Aggregation](#) for more information.

Interval register types define values common to interval registers of the same type. Open **Admin Menu, Interval Register Type** to define your interval register types.

Note. This page will not be available if *Meter Data Management* module is [turned off](#).

Description of Page

Enter a unique **Interval Register Type** and **Description** for the interval register type.

Enter the **Unit of Measure** (UOM) that data values stored for interval registers of this type are captured in.

If a service quantity identifier is needed to further qualify data stored for interval registers of this type, enter a valid **SQ Identifier**.

Enter the **Minutes per Interval** to define the number of minutes expected in between each row of data collected for interval registers of this type.

If the [installation](#) record indicates that [seasonal time shift](#) is required, then you must enter the appropriate **Seasonal Time Shift** record applicable for the interval data. Please take special note of the issue described in the [Evenly Sized Intervals](#) section.

The grid contains **Algorithms** that may be used to perform actions on interval data for registers of this type. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (see the table that follows for a description of all possible events).
- Specify the **Sequence** number and **Algorithm** for each system event. You can set the **Sequence Number** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

We expect that you will need more algorithms than we supply. Your algorithms will be based on any number of factors. Be aware that new algorithms may require programming. See [How To Add A New Algorithm](#) for more information.

The following table describes each **System Event**.

System Event	Description
<i>Interval Register Validation</i>	These types of algorithms are used to validate register data . Click here to see the algorithm types available for this system event.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_INTV_REG_TYP](#).

Setting Up Meter Configuration Types for Channels

You must set up [meter configuration types](#) for your interval channels and index channels.

Setting Up Meter Types

You must set up [meter types](#) to use for grouping together your index and interval registers. You must turn on the **Allow Interval Registers** switch for any meter type used by a meter, which will contain interval registers. Indicate the valid meter configuration types for this meter type.

Setting Up Time of Use Billing Control Tables

Contents

- [Setting Up TOU Map Relationship Types](#)
- [Setting Up TOU Pricing Rate Components](#)
- [Setting Up Time of Use Codes](#)
- [Setting Up TOU Groups](#)
- [Setting Up TOU Map Types](#)
- [Setting Up TOU Map Templates](#)

Setting Up TOU Map Relationship Types

TOU Map Relationship Types define the role that a TOU Map will serve for a customer. Open **Admin Menu, TOU Map Relationship Type** to define your TOU map relationship types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **TOU Map Relationship Type** ID and **Description** for the TOU map relationship type.

Enter the **TOU Map Type** collection. This is a list of valid [TOU map types](#) whose collections of data can serve the role defined by the TOU map relationship type.

Note, to aid in setup, the map type is not a required field on this user interface. This will help you to define your high level map relationship types first and then define the more detailed map types. The valid map types will need to be linked to the appropriate relationship types prior to creating service agreement TOU maps.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_TMAP_RELTY](#).

Setting Up TOU Pricing Rate Components

This section describes how to define your TOU Pricing rate components. Refer to [Time of Use Mapping and Pricing](#) for background information.

Define Rate Schedule, Rate Version and other Rate Components. Before creating your TOU pricing rate component, you must set up a Rate Schedule. Refer to [Setting Up A Rate Schedule](#) for more information. You will also need to set up a rate version and possibly set up other types of rate components. Refer to [Defining Rate Versions](#) and [Defining Rate Components](#) for more information.

When you are ready to set up your TOU pricing rate component, open **Main Menu, Rates, Rate Version**. Choose the rate version that this component should belong to. Open its context menu, and select **Go to Rate Component +** to add a new rate component.

Description of Page

Refer to [Rate Component - Main Information](#) for information about the common fields on this page. When defining a TOU Pricing rate component, additional fields become available to you. The following information will help you to set up your TOU Pricing rate components.

If the algorithm used by this rate component, only performs mapping and does not produce bill lines, indicate that this is **FCPO**.

The **Value Type** and **Value Source** are optional. If your algorithm performs pricing logic, then you will probably need to define a bill factor here. If your calculation algorithm only performs mapping, than you may not require any value here.

Indicate whether or not this rate component is **Seasonal**. Refer to [Rate Component - Main Information](#) for more information about seasonal rate components.

The **UOM** is available for use by your calculation algorithm. It could be used, for example, by a mapping algorithm to produce SQ quantities with a different unit of measure, assuming that the algorithm knows how to convert from one UOM to another. Note that the algorithms provided with the system do not use this field.

The **SQL** is available for use by your calculation algorithm. It could be used, for example, by a mapping algorithm to produce SQ quantities with a different SQL. Note that the [TOU mapping and pricing](#) algorithms provided with the system will use this field, if populated, to produce SQL quantities with this SQL value.

Measures Peak Qty and **GL Statistical Qty** behave the same way they do for SQ quantity rate components. Refer to [How To Set Up Service Quantity Rate Components](#) for more information.

The **Error if No Value** field is available for you to use in the calculation algorithm.

Indicate the **Calc Algorithm** that the system will use to map the interval quantities to time of use values.

- The system provides the following TOU mapping/pricing algorithms that are available for use here: [RCTMS](#) performs mapping of time of use only and [RCTMPS](#) performs both mapping and pricing, producing the necessary bill calculation lines.

If these algorithms do not provide you with the logic you require, you will need to create a new algorithm (refer to [Setting Up Algorithms](#)). The above existing algorithms should be used as a sample if you have to write a new algorithm type.

Note. The calculation algorithm's main purpose is to create bill calculation lines. However, the algorithm may populate other information for the bill, for example, it may add to the SQ or register read collection or it may overwrite the description on bill.

Indicate the **Audit Algorithm** to be used when a CSR wants to drill down into the details of a bill line that was calculated using this Rate Component.

- The system provides the following audit algorithm that is available for use here: [RCTPRS-ADT](#).

If this algorithm does not provide you with the logic you require, you will need to create a new algorithm (refer to [Setting Up Algorithms](#)). The above existing algorithm should be used as a sample if you have to write a new algorithm type.

Note. The audit algorithm should produce the same results as the Calc Algorithm. They are separated because they have different responsibilities. For example, the Calc Algorithm should produce bill lines, but the Audit Algorithm should not. They share common logic related to accessing and processing the appropriate interval data records. As a result, it is recommended that these two programs share a common code which accesses and processes the interval data. The above algorithms, provided by the system, behave this way and should be used as samples.

Refer to [Interval Billing Calculation Details](#) to understand where a CSR may view the calculation details for a bill calc line, using this algorithm.

Indicate the **Interval Profile Rel Type**. This indicates to the system the profile, linked to the SA, which contains the interval quantities to be mapped. Refer to [Business Role of Interval Data](#) and [Setting Up Profile Relationship Types](#) for more information.

If the TOU map differs for different service agreements, indicate the **TOU Map Relationship Type**. The system uses this information to find the correct TOU map linked to the SA. Refer to [Business Role of a TOU Map](#) and [Setting Up TOU Map Relationship Types](#) for more information.

If ALL service agreements linked to the rate use the same map, indicate the **TOU Map ID**.

Enter the verbiage to appear on the customer's bill in **Description On Bill** and turn on the **Print** switch. Refer to [Rate Version - Bill Print Info](#) for more information about these fields.

Setting Up Time of Use Codes

An important step in preparing for Time of Use billing is to define Time of Use codes. Refer to [Setting Up Time-Of-Use Codes](#) for more information.

Setting Up TOU Groups

Once you have your time of use codes defined, you will need to create your TOU groups to group the codes together. Refer to [Setting Up TOU Groups](#) for more information.

Setting Up TOU Map Types

The TOU Map Type defines characteristics that are common to TOU Maps of the same type. Open **Admin Menu, TOU Map Type** to define your TOU map types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **TOU Map Type ID** and **Description** for the TOU Map Type.

Indicate whether TOU maps of this type are **SA Owned** or **Common** by entering the appropriate value in the **Sub Type**.

The **Creation Priority** will be used by the [TOU map data creation](#) process to determine the order in which the data for TOU maps linked to an SA should be derived. The values range from **10**, being the highest priority to **90** being the lowest priority.

Note. The values for this field are customizable using the Lookup table. This field name is CRE_PRIO_FLG.

Enter the **TOU Group** that defines the collection of TOUs for this map type.

Enter the **Minutes Per Interval** to define the number of minutes expected in between each row of data collected for TOU maps of this type.

If the [installation](#) record indicates that [seasonal time shift](#) is required, then you must enter the appropriate **Seasonal Time Shift** record applicable for the interval data. Please take special note of the issue described in the [Evenly Sized Intervals](#) section.

The grid contains **Algorithms** that may be used to create TOU map data for TOU maps of this type. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (see the table that follows for a description of all possible events).
- Specify the **Sequence** number and **Algorithm** for each system event. You can set the **Sequence Number** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

We expect that you will need more algorithms than we supply. Your algorithms will be based on any number of factors. Be aware that new algorithms may require programming. See [How To Add A New Algorithm](#) for more information.

The following table describes each **System Event**.

System Event	Description
<i>TOU Map Creation</i>	These types of algorithms are used to create TOU map data automatically. Click here to see the algorithm types available for this system event.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_TMAP_TYPE](#).

Setting Up TOU Map Templates

The TOU Map Template page enables you to define daily, weekly and calendar templates for use by TOU data generation and automatic TOU data creation. Open **Admin Menu, TOU Map Template** to define your TOU map templates.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **TOU Map Template** name and a **Description** for the TOU map template.

Enter a **TOU Map Type**. **Daily** templates use the TOU map type to determine the TOU group whose TOU codes are valid for this template. All templates use the TOU map type to determine the interval size to ensure that start and end times are valid for the interval size. For example, if your interval size is 60 minutes, then the amount of minutes between your start and end times must be a multiple of 60. For example, you would not be able to enter a start time of 10:00 and an end time of 10:15.

Use the **TOU Map Template Type** to indicate whether this template is **Daily**, **Weekly** or **Calendar**.

The fields in the remaining portion of the page will depend on the TOU map template type.

A **Daily** template is used to define time of use periods applicable for hour/minute time ranges in an abstract day. Data in this template has no association with a specific day of the week or a specific calendar date. Each entry in the collection is used to define the time periods applicable for a given Time of Use code in this day. Enter the following information for your collection of time periods for the daily template.

Use the **Start Sequence Number** and the **End Sequence Number** to indicate whether the time period covered by this entry starts and ends on the same day or on different days. If the time period covered by the entry starts and ends on the same day, then the same number should be entered in both fields. If the time period ends on a different day (most likely one day later) than it starts, then the End sequence would be greater than the start sequence.

Enter the **Start Time** and **End Time** applicable for this **Time of Use** code. The valid time of use codes are limited to those belonging to the TOU group on the template's TOU map type.

Note: The times on the TOU map components are in legal time. During the generation process the system will convert the time definitions into standard time, taking daylight savings into account. Time shifting is based upon the Seasonal Time Shift defined on the template's TOU Map Type. Refer to [Seasonal Time Shift](#) for more information.

Note. When the TOU data is generated, intervals for this time of use code will be generated starting from the first time period AFTER the Start Time up to and including the End Time interval. Therefore, the end Time is inclusive and the start time is not inclusive.

A **Weekly** template is used to define the collection of daily templates that make up an abstract week. Data in this template has no association with specific calendar dates. You must indicate the **Week Start Day** to tell the system which day of the week is considered "day one".

Enter the following information for your collection of daily templates for the weekly template.

Use the **Start Sequence Number** and the **End Sequence Number** to indicate whether the time period covered by this entry starts and ends in the same week or in different weeks. If the days covered by the entry are in the same week, then the same number should be entered in both fields. If the time period ends in a different week (most likely one week later) than it starts, then the End sequence would be greater than the start sequence.

Enter the **Start Week Day** and **Start Time** and **End Week Day** and **End Time** applicable for this **Reference TOU Map Template**. The valid templates are limited to those that are daily and those whose TOU map type has the same TOU group and minutes per interval as this template's TOU map type.

Note: The times on the TOU map components are in legal time. During the generation process the system will convert the time definitions into standard time, taking daylight savings into account. Time shifting is based upon the Seasonal Time Shift defined on the template's TOU Map Type. Refer to [Seasonal Time Shift](#) for more information.

Note. When the TOU data is generated, intervals will be generated starting from the first time period AFTER the Start Time up to and including the End Time interval. Therefore, the end Time is inclusive and the start time is not inclusive.

A **Calendar** template is used to define the collection of weekly and daily templates that make up specific months and dates for a given calendar.

Enter the following information for your collection of daily and weekly templates for the calendar template.

Use the **Start Sequence Number** and the **End Sequence Number** to indicate whether the time period covered by this entry starts and ends in the same year or in different years. If the dates covered by the entry are in the same year, then the same number should be entered in both fields. If the time period ends in a different year (most likely one year later) than it starts, then the End sequence would be greater than the start sequence.

Use the **Start Date** (month and day) and **Start Time** and the **End Date** (month and day) and **End Time** to indicate the time period applicable for this **Reference TOU Map Template**. The valid templates are limited to those that are daily and weekly and those whose TOU map type has the same TOU group and minutes per interval as this template's TOU map type.

Note: The times on the TOU map components are in legal time. During the generation process the system will convert the time definitions into standard time, taking daylight savings into account. Time shifting is based upon the Seasonal Time Shift defined on the template's TOU Map Type. Refer to [Seasonal Time Shift](#) for more information.

Note. When the TOU data is generated, intervals will be generated starting from the first time period AFTER the Start Time up to and including the End Time interval. Therefore, the end Time is inclusive and the start time is not inclusive.

Where Used

The [generate](#) button on the TOU map page allows you to specify a TOU map template and a holiday TOU map template to use for generating TOU data.

If you have a TOU map creation algorithm defined on a TOU map type, this algorithm will need to use a TOU map template to create TOU data. The mechanism for defining the TOU map template to use depends on how the algorithm is designed. The TOU map creation algorithm provided by the system [ITMCCOPT](#) expects the template to be defined as a characteristic of the service agreement.

Setting Up Contract Option Control Tables

This section describes the pages related to maintaining contract options.

Contents

[Setting Up Contract Option Types](#)

[Setting Up Contract Option Event Types](#)

Setting Up Contract Option Types

This section describes the pages related to maintaining a contract option type.

Contents

[Contract Option Type - Main](#)

[Contract Option Type - Algorithms](#)

Contract Option Type - Main

Contract Option Type defines control information required for contract options. Open **Admin Menu, Contract Option Type** to define your contract option types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **Contract Option Type** ID and **Description**.

Indicate whether you **Allow Overlap** of the effective period for contract events linked to contract options of this type.

If the [installation](#) record indicates that [Seasonal Time Shift](#) is required, then you must enter the appropriate **Seasonal Time Shift** record applicable for the contract option event data.

Select a **Characteristic Type** that may be used on contract options of this type. The characteristic type's Description appears adjacent.

The following fields should be defined for each characteristic type:

Sequence	This field controls the order in which the characteristics appear on the contract option page.
Required	Turn this switch on if this type of characteristic must be defined on all contract options of this type.
Default	Turn this switch on if this characteristic type should automatically appear in the characteristic scroll area of contract options of this type.
Characteristic Value	If a characteristic value can default for contract options of this type, specify the default value in this field.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI COP TYPE](#).

Contract Option Type - Algorithms

To define validation algorithms for contract option events linked to contract options of this type. Open **Admin Menu, Contract Option Type** and navigate to the Algorithm tab to define validation algorithms for this contract option type.

Description of Page

The grid contains **Algorithms** that may be used to validate contract option event information. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (see the table that follows for a description of all possible events).

- Specify the **Sequence** number and **Algorithm** for each system event. You can set the **Sequence Number** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

The following table describes each **System Event**.

System Event	Description
<i>Contract Option Event Cancel</i>	This algorithm is executed when a contract option event for a contract option of this type is <i>canceled</i> . Click here to see the algorithm types available for this system event.
<i>Contract Option Event Freeze</i>	This algorithm is executed when a contract option event for a contract option of this type is <i>frozen</i> . Click here to see the algorithm types available for this system event.
<i>Contract Option Event Pending</i>	This algorithm is executed when a contract option event for a contract option of this type is added or changed and the status is <i>pending</i> . Click here to see the algorithm types available for this system event.

Setting Up Contract Option Event Types

Contract Option Event Type defines control information required for contract option events. Open **Admin Menu, Contract Option Event Type** to define your contract option event types.

Note. This page will not be available if **Complex Billing** module is [turned off](#).

Description of Page

Enter a unique **Contract Option Event Type** code and **Description**.

Indicate the **Contract Option Type** for this event type.

Use the **Characteristics** collection to define characteristics that can be defined for contract option events of this type. Use **Sequence** to control the order in which characteristics are defaulted. Turn on the **Required** switch if the **Characteristic Type** must be defined on contract option events of this type. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI COP EVT TYPE](#).

Defining Service Credit Options

Some companies allow their customers to participate in a special rewards program. The term “service credits” is used to describe a program that rewards customers for their business. The topics in this section provide details to help you set up the control tables required to support any service credit program that your company supports.

Penalty Points. The service credits functionality is described in the documentation with the assumption that it is used for accumulating points to reward your customers. However, if your company has a business need to record penalty points for a customer, the service credits functionality may be used for that purpose as well.

Contents

[The Big Picture Of Service Credits](#)
[Designing Your Service Credit Options](#)
[Setting Up Service Credit Options](#)
[Service Credit Examples](#)

The Big Picture Of Service Credits

The topics in this section provide background information about service credit functionality.

Contents

[Service Credit Membership](#)
[How Are Service Credits Earned?](#)
[How Are Service Credits Redeemed?](#)
[Service Agreements For a Membership](#)

Service Credit Membership

Let's look at some examples of special programs that may use service credits functionality:

- Capital credits - When customers receive their utility service from a cooperative, they are considered "members" of the cooperative and may over time receive capital credit allocations from the cooperative based on their service history and the cooperative's profits allocated during that time period.
- Frequent flier miles - Perhaps your company has made an agreement with one or more airlines to allow customers to accumulate frequent flier miles for every x amount spent on service.
- Free pay-per-view movies - Maybe your cable service offers free pay-per-view movies under certain conditions. Using service credits, you can set up your system to accumulate the free pay-per-view movies and use the free movies to offset actual movies viewed by the customer.
- Any other type of loyalty program where the customer earns credits that may later be redeemed in some way.

To participate in a program such as those described above, the customer is linked to a service credit membership. The membership record provides the following functionality:

- It defines the accounts that are linked to the membership.

Refer to [Who are the Members?](#) for information about linking persons and accounts to a membership.

- It defines a [membership type](#), which controls certain behavior about the membership.
- It may define an external ID if the membership is associated with an external program, such as a frequent flier mile program.
- It may define a [service agreement](#) to use for miscellaneous financial transactions that may get created.
- It may define characteristics used to capture miscellaneous information about the membership.
- Over time, service credit events are created for a membership. The events indicate an amount that either adds or subtracts credit units (i.e., points, miles, movies, dollars, etc) for the membership.

Refer to [The Big Picture of Service Credit Membership](#) for more information about functionality related to a membership.

How Are Service Credits Earned?

Service credits may be monetary rewards for service or they may be non-monetary rewards such as free movies or frequent flier miles. In any case, how the membership earns the points or rewards depends on the business rules for the program you are offering.

A typical scenario is that the service credits are earned for a membership as a result of other services linked to the membership's accounts. For example:

- Perhaps free pay-per-view movies are earned when signing up for cable service. In this case, the pay-per-view movie membership is related to the membership account's cable service agreement. Refer to [Service Credits Earned When Starting Service](#) for more information.
- Perhaps one frequent flier mile is earned for every \$10 spent on electricity. In this case, the frequent flier membership is related to the membership account's electricity service agreements (for example, electricity distribution, electricity retail and lamp service). Refer to [service credits earned through billing](#) for more information.
- For a capital credits membership, capital credit allocations are calculated based on the amount spent by the customer for standard service, for example electricity and/or gas service. In this case, the capital credit membership is related to the membership account's electricity and/or gas service agreements. For capital credits, a background process is used to calculate the allocated amounts. Refer to [Allocating Capital Credit](#) for more information.

It is also possible to earn service credits irrespective of other service for the membership's accounts. (Again, it depends on the business rules for the program you are offering.) For example, perhaps you offer 500 frequent flier miles for signing up for service with your company, regardless of the type of service chosen. In addition, assume that no additional miles are earned for ongoing service. In this example, there is no need to link the membership to any service agreements.

Refer to [Service Agreements Contribute to a Membership](#) for more information.

Each earned service credit amount is linked to the membership via a [service credit event](#).

How Are Service Credits Redeemed?

Once service credits have been earned for a membership, how may a customer redeem these credits? The method by which the credits are redeemed depends on the business rules for the program you are offering. Here are some examples of how a service credit may be redeemed:

- For free pay-per-view movies, perhaps your customer's monthly cable bill is credited for any pay-per-view movies until all free movies are used. Refer to [Service Credits Redeemed Through Billing](#) for more information.
- For frequent flier miles, the information about the earned miles is exported to the appropriate airline. The miles are actually redeemed by the customer through the airline, not through your company. Refer to [Interface Membership Information to a Third Party](#) for more information.
- For a capital credit membership, the company periodically (maybe once a year) decides if credits should be redeemed (referred to as "retired") and if so, how much. The company runs a background process to calculate the "retirement" amount. When an amount is retired, the membership balance is reduced by the retirement amount and the amount is transferred to service agreements related to the membership. Refer to [Capital Credit Retirement](#) for more information.

Tracking Membership Balances. If service credits are redeemed via the system, your membership should probably be configured to keep track of a balance. Refer to [Event Amounts May Contribute to a Balance](#) for more information.

Each service credit amount redeemed through the system is linked to the membership via a [service credit event](#).

Service Agreements For a Membership

There are three types of service agreements that may be associated with a service credit membership. The following points describe these three types.

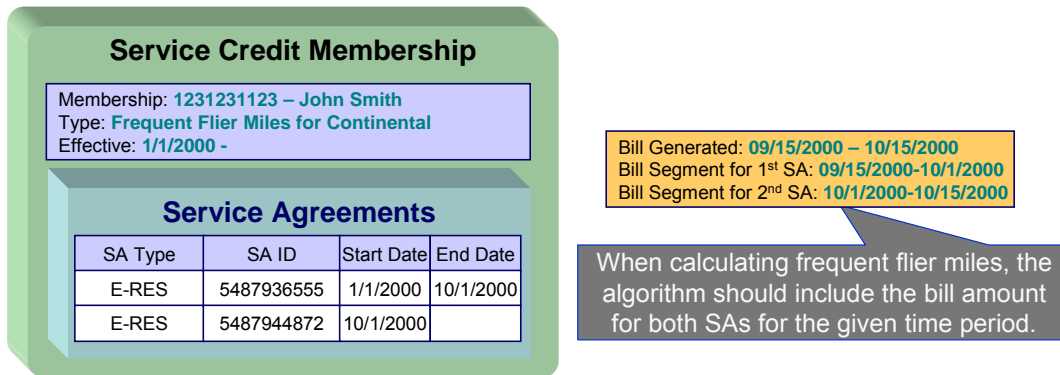
Contents

- [SAs Contribute to the Membership](#)
- [SA Used for Miscellaneous Transactions](#)
- [Membership Fee SAs](#)

SAs Contribute to the Membership

As described in [how are service credits earned](#), many memberships are related to specific service agreements for the membership's accounts. We refer to these service agreements as the SAs that contribute to the membership because often the service credit amounts earned for the membership are based on amounts spent by the customer for these services.

During the lifetime of a membership, service agreements that contribute to a membership may be stopped and other service agreements started. For example, perhaps you have a frequent flier membership that is related to electric service. Imagine that the customer starts out with a certain rate for electric service, but later decides to opt for a different type of rate that requires expiring the old service agreement and creating a new one. How does this affect your frequent flier mile calculation? It depends on how you design the [algorithm](#) that creates the frequent flier events. Essentially, the algorithm must cater for this situation. The following diagram illustrates the scenario.



The service agreements that contribute to a membership are not linked directly to the membership record. This would cause a maintenance burden, requiring links to be updated when service is stopped or started for applicable service agreements. Rather, this link is indirect. The list of service agreement types is defined for the membership type. The system can determine which service agreements are “linked” to the membership by looking at the SA types for the membership type.

Refer to [Determine The Types of Service Agreements That Contribute to the Membership](#) for more information on designing your membership type to include appropriate SA types.

SA Used for Miscellaneous Transactions

For some memberships, you may need to define a special service agreement to use for miscellaneous transactions. For example:

If events created for a membership cause an adjustment to be created to affect the general ledger, rather than allowing the system to arbitrarily pick a service agreement to use for this adjustment, the SA to use should be indicated on the membership. Refer to [An Event May Cause Other Actions to Occur](#) for more information.

Membership Fee SAs

For some memberships, a [membership fee](#) may be applicable. A special service agreement is used to hold the fee. This service agreement is not linked directly to the membership, but is simply a service agreement linked to one of the membership’s accounts.

Some fees may be refundable. The refunding of a fee must be handled by an algorithm. Refer to [SAST-RF](#) for information about the algorithm type provided with the base product.

Designing Your Service Credit Options

This section helps you to determine how to design your service credit membership types and service credit event types.

Contents

- [Designing Your Membership Types](#)
- [Designing Your Service Credit Event Types](#)
- [How Are Service Credit Events Created?](#)

Designing Your Membership Types

This section discusses the options to consider when designing your service credit membership types.

First consider the type of unit that your membership's service credit events represent.

- Is it related to a currency? If so, ensure that your currency is correctly defined on the [currency](#) page.
- Is it a non-currency unit, such as movies, points or miles? If so, you need to define the unit on the [credit unit](#) page.

Next, consider other behavior that your membership may exhibit.

- Should your membership [calculate an overall balance](#)?
- Will miscellaneous financial transactions be created for you membership over its lifetime? If so you may need to link a [service agreement to your membership](#).
- Should events linked to your membership reference a [fiscal year](#)?

The table below illustrates three types of memberships: one for capital credits, one for frequent flier miles and one for free pay-per-view movies.

Membership Type	Description	Unit Type / Currency or Credit Unit	Has Balance?	Require SA?	Fiscal Year?
STDCAPCR	Standard capital credit membership	<i>Currency</i> / \$	Yes	Yes	Yes
FFDELTA	Delta frequent flier miles	<i>Credit Unit</i> / Miles	No	No	No
FREEPPV	Free pay-per-view movies	<i>Credit Unit</i> / Movies	Yes	No	No

The following points explain the settings for each membership above:

- The capital credit membership uses a currency of US dollars for its units. Over time, capital credits are allocated and retired. The overall balance of the credits and debits should be calculated and displayed for information purposes. It is common for the allocation and retirement of capital credits to affect the GL, and as a result, a membership SA is required for posting these financial effects. Finally, in a capital credit situation, the allocation is typically related to a specific fiscal year. When calculating and displaying balances, the balance for each fiscal year must also be available. As a result, the fiscal year must be set to required.

- For the frequent flier membership type, a separate membership type must be created for each different airline. This is because a separate membership must exist for each separate airline in order to keep track of the accumulated miles correctly. The membership type does not have a balance because the accumulated miles are interfaced to the airline and the airline keeps track of the balance. In this example, no SA is required because an assumption is being made that the creation of frequent flier miles does not affect the GL. (Accumulating miles is no liability or expense for the company. The miles are simply accumulated on behalf of a third party.) Finally, the frequent flier miles do not need to indicate a fiscal year.
- For the free pay-per-view movies membership, we assume that the credits are redeemed from within the system. For example, perhaps a rate component calculation algorithm or SQ rule redeems the free pay-per-view movies over time as customers are billed for the movies. As a result, the membership should have a balance. In this example, no SA is required because an assumption is being made that any affect on the GL is posted at the time the free movies are redeemed (i.e., when calculating a bill). (This is just an example. It's possible that you may want to post to the GL when free movies are accumulated to mark a payable for the company.) Finally, free pay-per-view movies do not need to indicate a fiscal year.

More options must be considered for each membership type.

Contents

[Consider Whether the Membership Should Indicate Subcategories](#)
[Determine the Types of Service Agreements "Linked" to the Membership](#)
[Consider Special Functionality Needed When Adding, Activating or Inactivating a Members](#)
[Determine Whether The Membership Requires Additional Information](#)

Consider Whether the Membership Should Indicate Subcategories

For some types of memberships, the amount of each service credit event is further grouped by a subcategory. Refer to [Events May Indicate a Subcategory](#) for more information.

Use subcategories if multiple subtypes of credits may be accumulated and redeemed and if balances need to be tracked for each subcategory.

For our sample membership types, only the capital credits type should indicate subcategories.

Membership Type	Description	Subcategories
STDCAPCR	Standard capital credit membership	Distribution Transportation
FFDELTA	Delta frequent flier miles	
FREEPPV	Free pay-per-view movies	

Determine the Types of Service Agreements "Linked" to the Membership

Memberships typically exist to reward customers for participating in standard service with the company. Refer to [How Are Service Credits Earned?](#) for more information. If memberships of this type are related to standard service for your company, determine the SA types for these service agreements.

Standard Service SA Types. These service agreement types are probably different from the type of service agreement you may link as the membership SA. The membership SA is used for miscellaneous charges like general ledger posting. These SA types are related to electricity service, gas service, etc.

For our three examples, assume that the company provides electric service and cable. The capital credits are related only to electric service, frequent flier miles are accumulated for a combination of services, and free pay-per-view movies are related only to cable service.

Membership Type	Description	SA Types
STDCAPCR	Standard capital credit membership	E-RES (Electric Residential) AL-RES (Area Lighting)
FFDELTA	Delta frequent flier miles	E-RES (Electric Residential) AL-RES (Area Lighting) CABLE -RES (Cable Residential)
FREEPPV	Free pay-per-view movies	CABLE-RES (Cable Residential)

Consider Special Functionality Needed When Adding, Activating or Inactivating a Membership

Should a letter or bill message be generated when a membership is created or activated? Should anything happen when a membership is inactivated? If so, you may need to define one or more algorithms to be plugged in on the membership type.

For our examples, let's say that a bill message is generated when a frequent flier or free pay-per-view membership is created. Let's say that a letter is generated when a capital credits membership is activated. We also assume that when a capital credits membership becomes inactive, any outstanding balance is redeemed. For the frequent flier miles and pay-per-view memberships, we assume nothing special occurs when the membership becomes inactive.

Refer to [Lifecycle of a Membership](#) for more information about the various status values for a membership.

Membership Type	Description	Algorithm System Event	Algorithm
STDCAPCR	Standard capital credit membership	<i>Membership Activation</i>	Send Letter
		<i>Membership Inactivation</i>	Redeem Balances
FFDELTA	Delta frequent flier miles	<i>Membership Creation</i>	Generate Bill Message
FREEPPV	Free pay-per-view movies	<i>Membership Creation</i>	Generate Bill Message

Sample Algorithms. The base package does not provide sample algorithms for all the above examples. Refer to [Service Credit Membership Type - Algorithm](#) for more information about the algorithms provided with the system.

Determine Whether The Membership Requires Additional Information

Is there any information about your membership that you need to capture that is not already provided by the base system logic? If the answer is yes, you may need to define characteristics for your membership. Use the [characteristic collection](#) on the membership type to define the types of characteristics allowed for memberships of this type. You may also define default values for you membership's characteristics.

Designing Your Service Credit Event Types

Now that you have designed your membership types, you need to design the types of service credit events that may be created for your membership.

In many cases, a credit event should cause additional functionality to occur. Algorithms are executed when an event is completed and when an event is canceled and are used to perform additional functionality. The following points illustrate possible algorithms that may be needed when a credit event is completed.

- Validate the event as compared to other events. For example, perhaps a new event should never cause the overall membership balance to fall below zero. You could use an algorithm on the service credit event type to check this condition.
- Create an adjustment that posts to the general ledger. Your credit event may not affect the customer's balance when created, but perhaps it should have an effect on the general ledger. For these types of credit events, you may need to create an adjustment to post to the GL.
- Create adjustments to affect the customer's balance. When "redeeming" a credit, you may need to transfer a monetary amount to one or more of the customer's service agreements.
- Create a bill message. Perhaps when a credit is accumulated, you want to inform the customer via a message on the bill indicating the credit amount. A temporary bill message is added to one of the membership's accounts.

One Account Message of the Same Message Type. The temporary bill message collection on the account allows only one bill message of the same bill message type. If it's possible for multiple types of events to be generated for the same account, consider creating a different bill message type for each event type.

- Stamp a batch code and batch run number onto the event. This would be used when your event information needs to be interfaced to an external system.

The following points illustrate possible algorithms that may be needed when a credit event is canceled.

- Validate the event as compared to other events. For example, perhaps canceling an event should never cause the overall membership balance to fall below zero.
- Cancel adjustments that may have been created when the event was completed.

To illustrate examples of when to use some of the algorithms above, we'll design event types for the membership types designed above.

Contents

[Designing Capital Credit Event Types](#)
[Designing Frequent Flier Event Types](#)
[Designing Pay-per-view Event Types](#)

Designing Capital Credit Event Types

The following event types illustrate typical events for a capital credits membership.

SC Event Type	Description	Membership Type	Algorithm System Event	Algorithm
ALLOCATECCR	Capital credit allocation	STDCAPCR	<i>Event Creation</i>	Create Simple Adjustment

SC Event Type	Description	Membership Type	Algorithm System Event	Algorithm
			<i>Event Creation</i>	Generate Bill Message
			<i>Event Cancellation</i>	Validate Balance Not < Zero
			<i>Event Cancellation</i>	Cancel Related Adjustments
RETIRECCR	Retire capital credit (apply to customer's balance)	STDCAPCR	<i>Event Creation</i>	Validate Balance Not < Zero
			<i>Event Creation</i>	Create Adjustments to Affect Customer's Balance
			<i>Event Cancellation</i>	Cancel Related Adjustments
FORFEITCCR	Forfeit capital credit (retirement not applied to customer's balance)	STDCAPCR	<i>Event Creation</i>	Validate Balance Not < Zero
			<i>Event Creation</i>	Create Simple Adjustment (affect GL only)
			<i>Event Cancellation</i>	Validate Balance Not < Zero
			<i>Event Cancellation</i>	Cancel Related Adjustments

These event types assume the following:

- When credit events are allocated, the customer is notified via a bill message, the general ledger is affected so a simple GL only adjustment is created. If an event of this type is canceled, any adjustments that were created should be canceled. The event amount should always be positive, so checking that the membership balance does not fall below zero is only checked for event cancellation.
- When capital credits are retired, it's possible that the full membership balance is not applied to the customer's balance. For the portion of the retirement that does affect the customer's balance, you need an algorithm that applies the credits to the customer's service agreements via adjustments. Cancellation of this event should cause any related adjustments to be canceled. It's assumed that the amount of this event is a credit so checking that the membership balance does not fall below zero is only checked for event completion.
- For the portion of the retirement that is not applied to the customer's balance, the event amount should simply affect the GL so a GL only adjustment is created. Cancellation of this event should cause any related adjustments to be canceled. It's assumed that the amount of this event is a credit so checking that the membership balance does not fall below zero is only checked for event completion.

Service credit event types are independent of subcategories. A capital credits membership typically uses subcategories. When events are created for different subcategories, the same service credit event type may be used. As a result, all subcategories use the same event completion and event cancellation algorithms.

Designing Frequent Flier Event Types

The following event types illustrate typical events for a frequent flier membership:

SC Event Type	Description	Membership Type	Algorithm System Event	Algorithm
ADDMILES	Add miles to the membership	FFDELTA	<i>Event Creation</i>	Generate Bill Message

SC Event Type	Description	Membership Type	Algorithm System Event	Algorithm
			<i>Event Creation</i>	Populate Batch Information
			<i>Event Cancellation</i>	Event may not be canceled.

This event type assume the following:

- A new event should generate a bill message.
- Information about the event amount should be interfaced to an external system so batch information should be populated when the event is completed.
- Events of this type may not be canceled because the information is interfaced to an external system. Rather, to reverse an event, simply create a new event whose amount is a credit. This credit amount is also interfaced to an external system.

Sample Algorithm. The system does not provide a sample cancellation algorithm that prevents the event from being canceled.

- There is no validation to ensure that the balance does not fall below zero. Recall that this membership was defined as not requiring a balance.

Notice that only one type of event has been defined for this membership. That is because the credits for this membership are not redeemed via this system. Rather they are accumulated on behalf of an external system.

Designing Pay-per-view Event Types

The following event types illustrate typical events for a free pay-per-view movies membership:

SC Event Type	Description	Membership Type	Algorithm System Event	Algorithm
ADDPV	Free pay-per-view movies	FREEPPV	<i>Event Creation</i>	Generate Bill Message
			<i>Event Cancellation</i>	Validate Balance Not < Zero
REDEEMPPV	Redeem pay-per-view movies	FREEPPV	<i>Event Creation</i>	Validate Balance Not < Zero
			<i>Event Creation</i>	Generate Bill Message

These event types assume the following:

- Any type of new event should generate a bill message.
- For adding free movies, it is assumed that the quantity is positive so when the event is cancelled, the algorithm verifies that the balance does not become negative.
- For redeeming free movies, it is assumed that the quantity is negative so when the event is completed, the algorithm verifies that the balance does not become negative.
- These events do not affect the general ledger and they do not directly affect the customer's balance so no adjustment algorithm is needed.

How Are Service Credit Events Created?

Now that you have designed the behavior of your service credit events, an important issue is to determine how these events are created. Consider your business practice for each type of membership.

Let's use our sample memberships to work through different ways that you may create service credits.

Contents

[Service Credits for Capital Credit Memberships](#)

[Service Credits for Frequent Flier Memberships](#)

[Service Credits for Free Movies Memberships](#)

Service Credits for Capital Credit Memberships

For a capital credit membership, capital credit allocations are calculated once a year based on billing history and the cost of service. To accomplish this, a background process calculates the amount and creates the service credits. Refer to [allocating capital credits](#) for a sample process provided with the base package.

Credits are redeemed via the retirement process. The company determines when to retire capital credits based on analysis of their financial situation. This retirement process is also handled by a background process. Refer to [capital credit retirement](#) for a process provided with the base package.

For capital credits memberships, special functionality is required when a member dies. The capital credits are considered part of the person's estate and may need to be retired and applied to a beneficiary's account. This process depends on the company's business practice. However, typically, the membership status would change to inactive so that new capital credit allocations are not created for the membership. The system provides a sample inactivation algorithm called [SCMI-RB](#) that transfers part or all of the outstanding credit balance to the member's service agreements. From there, a user can cut a check to the beneficiary.

Some Credits Are Never Retired. For many cooperatives, some types of allocated credits are never retired. Refer to [Partial Retirement](#) for more information.

Service Credits for Frequent Flier Memberships

In our example, frequent flier miles were related to both the electricity and phone service. Let's assume that frequent flier miles are accumulated every \$x spent on electricity and phone service. In this example, a bill completion algorithm creates service credits based on the bill segment amounts for these service agreements. Refer to [Service Credits Earned Through Billing](#) for more information.

As mentioned before, the frequent flier miles are not redeemed using this system, but are interfaced to a third party for redemption.

Service Credits for Free Movies Memberships

Sample Algorithms. No base package algorithm types are provided to support the logic described in this example.

In this scenario, let's suppose that a customer receives three free pay-per-view movies when signing up for new service. To handle this, perhaps an SA creation algorithm creates the membership and the service credits when the service is started. Or perhaps you want to wait until the first bill is generated and a bill completion algorithm is used to generate the first credit. It depends on your business practice.

For redeeming the free pay-per-view movies, it is assumed that the movies are credited during billing after it is determined that the customer has been billed for a movie. The number of movies used for the membership is reduced until all the free movies are used.

Refer to [Service Credits Redeemed Through Billing](#) for more information.

Setting Up Service Credit Options

Contents

- [Setting Up Credit Units](#)
- [Setting Up SC Membership Inactive Reasons](#)
- [Setting Up Service Credit Membership Types](#)
- [Setting Up Service Credit Event Types](#)

Setting Up Credit Units

Credit unit is used for service credit membership types whose [events](#) record non-monetary units. Open **Admin Menu, Credit Unit** to set up credit units.

Description of Page

The following fields display for each credit unit:

Credit Unit	The unique identifier of the credit unit.
Description	The description of the unit. This also acts as a label for the unit when displaying information about a service credit event.
Symbol / Label Position	Indicates whether or not the label for this credit unit appears as a Prefix or as a Suffix to the service credit event amount.
Decimal Positions	Indicates the number of decimal positions used for this credit unit. This information should be used by any algorithm or background process that creates a service credit event to determine how to store the event amount. It is also use to correctly display the service credit amounts.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_CR_UNIT](#).

Setting Up SC Membership Inactive Reasons

The service credit membership inactive reason must be specified when the status of a service credit membership changes to *inactive*. Open **Admin Menu, SC Membership Inactive Reason** to set up service credit inactive reasons.

Description of Page

The following fields display for each inactive reason:

Inactive Reason	The unique identifier of the service credit membership inactive reason.
Description	The description of the inactive reason.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI_SCM_NCTV_RSN](#).

Setting Up Service Credit Membership Types

Contents

- [SC Membership Type - Main](#)
- [SC Membership Type - Algorithm](#)
- [SC Membership Type - Characteristics](#)

SC Membership Type - Main

Service credit memberships have a membership type. Open **Admin Menu, Service Credit Membership Type** to define the membership type.

Description of Page

Enter a unique **Service Credit Membership Type** and **Description** for each membership type.

Use the **SA Requirement** flag to indicate whether a [miscellaneous SA](#) must be linked to memberships of this type. The possible values are *SA Required* and *SA Not Allowed*.

Default Note. The value defaults to *SA Required*.

Use the **Fiscal Year Requirement** flag to indicate whether events linked to memberships of this type should indicate a [fiscal year](#). The possible values are *Fiscal Year Required* and *Fiscal Year Not Allowed*.

Default Note. The value defaults to *Fiscal Year Not Allowed*.

Use the **SCM Event Balance** flag to indicate whether or not a [balance of service credit events](#) linked to memberships of this type should be calculated and displayed. The possible values are *Has a Balance* and *No Balance*.

Default Note. The value defaults to *Has a Balance*.

Use the **SC Membership Type Unit** flag to indicate whether or not the unit for the event amounts for events linked to memberships of this type is **Currency** or **Credit Unit**. When the unit is currency, indicate the **Currency Code**. When the unit is credit unit, indicate the **Credit Unit**.

If events linked to memberships of this type must indicate a subcategory, enter a valid **Subcategory Name** and **Description** for each subcategory applicable to this membership.

Refer to [Events May Indicate a Subcategory](#) for more information.

Use the **SA Types** grid to indicate the types of [service agreements that contribute to memberships](#) of this type. Indicate the **CIS Division** and **SA Type** for each type of service agreement that is related to a membership of this type.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI SCM TYPE](#).

SC Membership Type - Algorithm

Open **Admin Menu**, **Service Credit Membership Type** and navigate to the **Algorithm** tab to define any algorithms that are associated with a membership type.

Description of Page

The grid contains **Algorithms** that control important functions in the system. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (descriptions of all possible events are provided below).
- Specify the **Sequence** and **Algorithm** for each system event. You can set the **Sequence** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

The following table describes each **System Event** for which you can define algorithms.

System Event	Optional / Required	Description
<i>Membership Creation</i>	Optional	Use this system event for algorithms that are executed when a membership is created. Click here to see the algorithm types available for this system event.
<i>Membership Activation</i>	Optional	Use this system event for algorithms that are executed when a membership becomes <i>active</i> . Click here to see the algorithm types available for this system event. Creating Memberships In Active Status. A membership may be created in either <i>pending</i> status or in <i>active</i> status, based on your business practice. For memberships created in <i>active</i> status, the system executes the membership creation algorithms and the membership activation algorithms.

<i>Membership Inactivation</i>	Optional	Use a system event of <i>Membership Inactivation</i> for algorithms that are executed when a membership becomes <i>inactive</i> . Click here to see the algorithm types available for this system event.
--------------------------------	----------	---

SC Membership Type - Characteristics

To define characteristics that can be defined for service credit memberships of a given type, open **Admin Menu, Service Credit Membership Type** and navigate to the **Characteristics** tab.

Description of Page

Use the characteristics collection to define characteristics that can be defined for service credit memberships of a given type. Use **Sequence** to control the order in which characteristics are defaulted. Turn on the **Required** switch if the **Characteristic Type** must be defined on service credit memberships of a given type. Turn on the **Default** switch to default the **Characteristic Type** when service credit memberships of the given type are created. Enter a **Characteristic Value** to use as the default for a given **Characteristic Type** when the **Default** switch is turned on.

Setting Up Service Credit Event Types

Service credit events created for a service credit membership have an event type. Open **Admin Menu, Service Credit Event Type** to set up service credit event types.

Description of Page

Enter a unique **Service Credit Event Type** and **Description** for each event type.

Indicate the **Service Credit Membership Type** to which this event type belongs.

The grid contains **Algorithms** that control important functions in the system. You must define the following for each algorithm:

- Specify the **System Event** with which the algorithm is associated (descriptions of all possible events are provided below).
- Specify the **Sequence** and **Algorithm** for each system event. You can set the **Sequence** to 10 unless you have a **System Event** that has multiple **Algorithms**. In this case, you need to tell the system the **Sequence** in which they should execute.

The following table describes each **System Event** for which you can define algorithms.

System Event	Optional / Required	Description
<i>Event Completion</i>	Optional	Use a system event of <i>Event Completion</i> for algorithms that are executed when an event is completed. Refer to An Event May Cause Other Actions to Occur for more information. Click here to see the algorithm types available for this system event.
<i>Event Cancellation</i>	Optional	Use this system event for algorithms that are executed when an event is <i>canceled</i> . Click here to see the algorithm types available for this system event.

Where Used

Follow this link to open the data dictionary where you can view the tables that reference [CI SC EVT TYPE](#).

Service Credit Examples

In this section, we provide examples of how to define your control tables to support functionality related to different types of service credit memberships. While your company may not define your environment exactly the same way, this section should help solidify your understanding of how to set up your company's service credit options.

Contents

- [Defining Control Tables for a Refundable Fee](#)
- [Defining Control Tables for a Nonrefundable Fee](#)
- [Defining an SA Type for Miscellaneous Transactions](#)
- [Using Campaigns/Packages to Set Up Membership](#)
- [Defining Another Person for Your Account](#)
- [Service Credits Earned When Starting Service](#)
- [Service Credits Earned Through Billing](#)
- [Service Credits Redeemed Through Billing](#)
- [Designing Your Rate Options for Capital Credits](#)
- [Partial Retirement](#)
- [Interface Membership Information to a Third Party](#)

Defining Control Tables for a Refundable Fee

If your membership requires payment of a [fee](#) that is refundable, you must define an SA type to use for the refundable fee.

Contents

- [Adjustment Type for Refundable Fee](#)
- [SA Type for Refundable Fee](#)
- [Start Option for Refundable Fee](#)
- [Algorithm for Refunding the Fee](#)

Adjustment Type for Refundable Fee

This example assumes that the fee is set up similarly to a deposit. The adjustment used to charge the fee would affect the current balance, but not the payoff balance or the general ledger because the fee is not considered a "receivable", rather it is an amount collected and held for the customer.

Create an [adjustment type](#) for levying the fee. Indicate the fee amount and indicate the adjustment FT creation algorithm [ADJT-CA](#), which affects the current balance, but not the payoff balance or the general ledger.

Adjustment Type Profiles. Be sure to add this adjustment type to an appropriate [adjustment type profile](#) and ensure this profile is linked to your SA type.

SA Type for Refundable Fee

Create an SA type to use for the fee. This SA type should be marked as not billable.

Indicate the appropriate adjustment type profile that includes the adjustment type to levy the fee.

Indicate an appropriate payment segment type that references the [PSEG-NM](#) payment segment FT algorithm. This algorithm affects the payoff amount and current amount by the payment amount, which should cause the current amount to become zero and the payoff amount to become a credit for the fee amount when the fee is paid.

Start Option for Refundable Fee

For this SA type, define a start option that causes an adjustment to be levied as part of the start service process.

Refer to [Including The Membership Fee](#) on campaigns and packages for more information about levying the fee via the order transaction.

Algorithm for Refunding the Fee

The base product provides an algorithm type [SAST-RF](#) that refunds a fee when [service agreements that contribute to the membership](#) are stopped. You must create an algorithm for this algorithm type and enter the SA type created above as an input parameter.

This SA stop algorithm must be plugged in on all SA types that you defined for the [membership type](#).

Defining Control Tables for a Nonrefundable Fee

If your membership requires payment of a [fee](#) that is not refundable, you can set this up in two ways:

- You can create a special SA type to handle charging the fee. For this SA type, define a start option that causes an adjustment to be levied as part of the start service process. This adjustment contains your fee. You may use this option if the membership is related to multiple types of services and not all services need to be present in order to create the membership.
- You can levy an adjustment on one of the other service agreements that is being started. To do this, you would use a start option to define an adjustment to be levied as part of the start service process. You may use this option if the membership is related to a single service (for example, free pay-per-view movies).

For our example, we will set up the SA for the nonrefundable fee with a separate 'fee' SA type.

Contents

- [Adjustment Type for Nonrefundable Fee](#)
- [SA Type for Nonrefundable Fee](#)
- [Start Option for Nonrefundable Fee](#)

Adjustment Type for Nonrefundable Fee

Create an [adjustment type](#) for levying the fee. Indicate the fee amount and indicate the adjustment FT creation algorithm [ADJT-NM](#), which affects the current balance and payoff balance by the adjustment amount, and affects the general ledger.

Adjustment Type Profiles. Be sure to add this adjustment type to an appropriate [adjustment type profile](#) and ensure this profile is linked to your SA type.

SA Type for Nonrefundable Fee

Create an SA type to use for the fee. This SA type should be marked as not billable.

Indicate the appropriate adjustment type profile that includes the adjustment type to levy the fee.

Indicate an appropriate payment segment type that references the [PSEG-NM](#) payment segment FT algorithm. This algorithm affects the payoff amount and current amount by the payment amount, which causes the current amount to become zero and the payoff amount to become a credit for the fee amount when the fee is paid.

Start Option for Nonrefundable Fee

For this SA type, define a start option that causes an adjustment to be levied as part of the start service process.

Refer to [Including The Membership Fee](#) on campaigns and packages for more information about levying the fee via the order transaction.

Defining an SA Type for Miscellaneous Transactions

Does your membership require a [service agreement](#) to support miscellaneous transactions? If so, you need to consider the SA type to use for this service agreement. This SA type should be marked as not billable.

If you choose to use the order transaction to set up the membership, this SA type must be defined on the [algorithm that creates the membership](#).

It is possible that you may require a start option for this SA type, for example if you want to define a characteristic for service agreements of this type.

Using Campaigns/Packages to Set Up Membership

If enrollment in a membership is a common occurrence for your customers when starting service, you should consider using the [order](#) page to start service for the customer and to create the membership as well.

Note. This section only makes sense if you are familiar with the [Sales and Marketing](#) chapter.

The recommendation is to use a question/miscellaneous field to ask the customer service representative to indicate the appropriate membership type. Algorithms validate this membership type and use it to create a membership of that type.

This section walks you through how to set up the campaign / package required to support this.

Contents

[Column Reference for Membership Type](#)
[Algorithms to Create Membership via Order](#)

Define a Campaign for Creating a Membership When Starting Service Including the Membership Fee

Column Reference for Membership Type

In order to ask the customer service representative to define an appropriate service credit membership type, you must define a column reference for the membership type.

Add a new column reference with the following information:

- Column Reference: SCM-TYPE
- Description: SC Membership Type
- FK Reference: **SCM TYPE**
- Long Description: Service Credit membership type to use when creating a service credit membership at start time.

Column Reference Algorithms. This column reference indicates a validation algorithm and a posting algorithm. However, we have not defined them yet so for now simply save this information.

Algorithms to Create Membership via Order

The base product provides two algorithm types to support the creation of a membership record via the order page: a validation algorithm type and a posting algorithm type.

Validate Membership Information

This algorithm is a column reference validation algorithm that checks that an input membership type is valid. Refer to [CRVL-ME](#) for more information. You must define an appropriate algorithm for this algorithm type, and on that algorithm you must define the column reference used to identify the membership type. For example:

- Algorithm: VAL MEM TYPE
- Description: Validate SC Membership Type
- Algorithm Type: **CRVL-ME**
- Parameter1: (Column Reference for Membership Type): SCM-TYPE

Post Membership Information

This algorithm is a column reference posting algorithm that creates a membership using the membership type indicated by the user. Refer to [CRPS-ME](#) for more information. You must define an appropriate algorithm for this algorithm type, and on that algorithm you must define the column reference used to identify the membership type. For example:

- Algorithm: CREATE MEMBRSHIP
- Description: Validate SC Membership Type
- Algorithm Type: **CRPS-ME**
- Parameter1: (Column Reference for Membership Type): SCM-TYPE

- Parameter2: (Service Credit Membership Status): **10**-Pending or **20**-Active. Refer to [Lifecycle of a Membership](#) for more information.
- Parameters3-5: (Division, SA Type, Start Option): Indicate the information needed to create an SA for miscellaneous transactions.

Note. This algorithm first looks for an existing service agreement of this division / SA type linked to the membership's accounts. If one exists, it uses that SA to link to the membership. If an SA does not exist, it creates one with the input division, SA and (optional) start option.

Update the Column Reference. Now that you have defined the validation and posting algorithms, return to your [column reference for membership type](#) and define the algorithms there.

Although your company may support multiple types of memberships, this column reference and its algorithms have been designed such that only one column reference for membership type would be needed to set up any type of membership. Although the posting algorithm for the membership type column reference indicates an SA type to use for miscellaneous transactions, a service agreement is only created and linked if your membership type indicates that an SA is required. As a result, you may use the same column reference for both memberships that require an SA and those that don't require an SA. However, if you have different membership types that require an SA and each uses a different SA type or a different start option for the membership SA, you need to define more than one posting algorithm and, as a result, more than one column reference.

Define a Campaign for Creating a Membership When Starting Service

Many factors must be considered when [designing your campaigns and packages](#). The possible creation of a membership when using the order page is simply another factor to consider.

If you plan to define a question/miscellaneous field to capture a service credit membership type, the available packages linked to the campaign should be ones that are related to memberships. For example, if your membership is related to electric service, it doesn't make sense to create a membership for a campaign designed to generate a one-time charge.

When defining a question/miscellaneous field, you must indicate its applicability. Consider whether a membership type is required, optional or only applicable for certain packages. This helps ensure that your users capture this information when appropriate.

Including the Membership Fee

You must determine the best way to setup your campaign/package to handle the levying of your membership fee, if applicable. Consider some of the following questions.

- If the membership has a fee, is it a refundable fee or a non-refundable fee?
- Is the fee always applicable for the membership? Or is it waived under certain conditions? For example, maybe the fee is applicable if a customer signs up for a single service, but the fee is waived if the customer signs up for two or more services.

The SCM type column reference and question/miscellaneous field are set up to ask the user what type of membership to create. Because the applicability of the fee may differ for each membership type, you should carefully consider the campaign / package setup to levy the fee correctly.

- If a fee is always applicable for a membership, you may consider creating a membership creation algorithm that creates a fee SA with a start option to levy the fee when the membership is created.

Sample Algorithm. The base product does not provide a membership creation algorithm to do this.

- If the fee is not always applicable for a membership, you must determine when the fee is applicable. The recommendation is to include the fee SA type in the SAs-to-create collection for the appropriate package. For example,
 - If the fee is applicable when the customer signs up for a single service, you should define a package for each single service that includes one SA to create for the single service and the fee SA as another SA to create.
 - If the fee is waived when the customer signs up for two or more services, you should define a package for each combination of the multiple services. These packages do not include the fee SA in the SAs to create.
 - If the customer signing up for service is a former customer who has returned, perhaps the fee was paid earlier when the customer originally had service. In this case, maybe your business practice is to waive the fee at this time. To do this, you should set up a question/miscellaneous field for the user to mark that this is a returning customer. Based on the answer to this question, perhaps only packages that do not levy any fee are eligible for selection.

Defining Another Person for Your Account

It is common for a capital credit membership to define more than one person for the account being turned on and linked to a membership. A typical example is a married couple. Both spouses are indicated on the account and financially responsible persons and as a result, both are considered [members](#).

Column reference algorithms have been provided by the base product to enable linking a second person to your account via the [order](#) page.

This section walks you through how to set up the column reference and campaign to support this.

Linking A Second Person to the Account. This logic is not restricted to service credit functionality. Any campaign may be designed to include the ability to link a second person to the account being started.

Contents

[Column References for Additional Person](#)
[Algorithms to Link Additional Person via Order](#)
[Design a Campaign to Include Linking an Additional Person](#)

Column References for Additional Person

In order to ask the customer service representative to link an additional person for the account, you must define several column references to use as questions/miscellaneous fields.

- We should assume that the person may already exist in the database. As a result, a question to record the person ID is needed.
- If the person does not already exist, the user should capture the person's name, an ID type and an ID number. Questions for these three fields are needed. The system uses this information to create a new person.
- Whether we are using an existing person or creating a new one, the person's link to the account must include an account relationship type. A question to record the account relationship type is needed.

Add a new column reference for person ID:

- Column Reference: PERSON-ID
- Description: Person ID
- FK Reference: **PER**
- Long Description: Person ID of the additional person to link to the order's account.

Add a new column reference for person name:

- Column Reference: PERSON-NAME
- Description: Person Name
- FK Reference: not applicable
- Long Description: Name of a new person to create.

Add a new column reference for identifier type:

- Column Reference: PER-ID-TYPE
- Description: Identifier Type
- FK Reference: **ID TYPE**
- Long Description: Identifier type for the primary ID of the additional person to link to the order's account.

Add a new column reference for identifier number:

- Column Reference: PER-ID-NUM
- Description: Person ID Number
- FK Reference: not applicable
- Long Description: Identifier number for the primary ID of the additional person to link to the order's account.

Add a new column reference for account relationship type:

- Column Reference: ACCT-REL-TYPE
- Description: Account Relationship Type
- FK Reference: **ACCT REL**
- Long Description: Relationship type to use for the link between the additional person and the order's account.

Column Reference Algorithms. One of the column references above must indicate a validation algorithm and a posting algorithm. However, we have not defined them yet so for now simply save this information. We recommend using the account relationship type record because that is used for all additional persons.

Algorithms to Link Additional Person via Order

The base product has provided two algorithm types to support the linking of an additional person to the account via the order page: a validation algorithm type and a posting algorithm type.

Validate Addition Person Information

This algorithm is a column reference validation algorithm which checks that either a person id or person name, a valid ID type and ID number are provided and that a valid account relationship type is provided. Refer to [CRVL-PE](#) for more information. You must define an appropriate algorithm for this algorithm type and on that algorithm you must define the column reference used for the five fields required for this validation. For example:

- Algorithm: VAL ADD PER
- Description: Validate Additional Person Info
- Algorithm Type: **CRVL-PE**
- Parameter1: (Column Reference for Person ID): PERSON-ID
- Parameter2: (Column Reference for Person Name): PERSON-NAME
- Parameter3: (Column Reference for Person ID Type): PER-ID-TYPE
- Parameter4: (Column Reference for Person ID Number): PER-ID-NUM
- Parameter5: (Column Reference for Account Relationship Type): ACCT-REL-TYPE

Post Addition Person

This algorithm is a column reference posting algorithm that may link an existing person to the order's account or create a new person and link that person to the order's account. Refer to [CRPS-PE](#) for more information. You must define an appropriate algorithm for this algorithm type, and on that algorithm you must define the column reference used for the five fields required for this logic. For example:

- Algorithm: LINK ADDNTL PER
- Description: Link additional person to order's account
- Algorithm Type: **CRPS-PE**
- Parameter1: (Column Reference for Person ID): PERSON-ID
- Parameter2: (Column Reference for Person Name): PERSON-NAME
- Parameter3: (Column Reference for Person ID Type): PER-ID-TYPE
- Parameter4: (Column Reference for Person ID Number): PER-ID-NUM
- Parameter5: (Column Reference for Account Relationship Type): ACCT-REL-TYPE

Update the Column Reference. Now that you have defined the validation and posting algorithms, return to your [column reference for account relationship type](#) and define the algorithms there.

Design a Campaign to Include Linking an Additional Person

Any campaign related to a specific account may include the questions and miscellaneous fields defined here to create/link an additional person to the account.

Simply create entries in the questions and miscellaneous fields collection to prompt the user to ask for the required information. A question should exist for each column reference created above. The following table illustrates a possible setup.

Seq	Description	Prompt on Order	Column Reference	Dependency
10	Account relationship type for additional person.	If you would like to link another person to this account, please enter an account relationship type.	<i>ACCT-REL-TYPE</i>	Must have account
20	Person ID for additional person	If the additional person already exists in the database, enter the person id.	<i>PERSON-ID</i>	Must have account
30	Person name for additional person	If you would like to create a new person, please enter the person name.	<i>PERSON-NAME</i>	Must have account
40	ID type for additional person	Please enter the ID type of the new person.	<i>PER-ID-TYPE</i>	Must have account
50	ID Number for additional person	Please enter the ID number of the additional person.	<i>PER-ID-NUM</i>	Must have account

Service Credits Earned When Starting Service

For some memberships, you may want to add service credits when starting the program. For example, the customer gets three free pay-per-view movies when signing on for cable service. Or a customer receives 500 frequent flier miles for signing up for a combination of gas and electric service.

There are various ways that you can accomplish this. You should work with your implementers to consider the various options to determine the method that best suits your business practice.

- You could use an SA activation algorithm to create a membership and an event for the initial points. Use this method if the points are related to a single type of service agreement and the points are earned automatically when starting service (i.e., without any human decision to be made).
- You could use a membership creation or membership activation algorithm to generate a service credit event automatically. This method assumes that the decision to create the membership has been made and that the free initial points are always earned for this type of membership (i.e., regardless of the type of service created).

- You could use questions and miscellaneous fields for a campaign/package to determine a customer's eligibility for participation in the membership and for the initial free points. A column reference posting algorithm could create the membership and/or the service credit event for the free points based on the answers to the questions.
- Perhaps the initial free points are only earned after the first bill is generated. Use a bill completion algorithm to generate the initial points.

You may think of other plug-in spots that could be used to generate free initial points based on your business needs.

Service Credits Earned Through Billing

For some memberships, you may accumulate points as a result of billed amounts for other services. For example, perhaps your customers earn one frequent flier mile for every \$10 spent on the combined electricity, gas and water bills each month.

To accomplish this, you must design an algorithm to be executed at billing time. There are various plug-in spots executed at billing time that you may use, but the recommended plug-in is a post completion algorithm on the customer class. This plug-in is executed after all bill segments are frozen and most of the completion logic has occurred.

Your algorithm should determine the [service agreements that contribute to the membership](#) and calculate the service credit amount for those service agreements bill segments.

This algorithm should also consider what to do when bill segments that contributed to the event are canceled. The algorithm provide with the base product [CBCM-SC](#) checks to see if any canceled bill segments are referenced on any previous events. If so, it includes the canceled amount on the new event. This may cause the new event to be a negative amount. The assumption is that over time, earned credits will compensate for the negative event amount. For a membership that [interfaces information to a third party](#), it is assumed that the negative event amount is also interfaced.

Service Credits Redeemed Through Billing

For some memberships, your customer may redeem their earned points by receiving a discount on their bill. For example, if your cable customer has earned one free pay-per-view movie, you can give them a credit the next time they get billed for a pay-per-view movie.

The base product has not provided any algorithms to credit a bill based on earned service credits. This section will identify considerations your implementers should follow when designing algorithms to redeem service credits through billing.

The recommendation is to credit the customer's bill by generating an adjustment using a pre-bill completion algorithm. This algorithm's responsibilities are as follows:

- Determine if the appropriate bill segment(s) contain the charges that need to be credited. For example, if the service credit is for a free pay-per-view movie, determine if the customer has been billed for a pay-per-view movie.
- Determine if the customer's current service credit balance for this program. For example, how many free pay-per-view movie credits are left?
- Create an adjustment to credit the appropriate service agreement by the eligible credit amount.

- Update the service credit membership balance by creating a new SC event with a credit for the number of points redeemed. Link the adjustment to the event as a **contributed to FT**.
- Consider cancel/rebill situations. If a cancel/rebill has occurred, determine if there is a change to the redeemed credits. For example, if the original bill had one pay-per-view movie that was credited and the new bill also has one pay-per-view movie, no change is needed to the service credit balance. If the original bill had more pay-per-view movies than the new bill, perhaps one or more redeemed service credit points should be reinstated. If the original bill had fewer pay-per-view movies than the new bill, perhaps more points should be redeemed.

You may wonder why we don't recommend crediting the customer's bill while generating the bill segment. For example, use an SQ rule to determine if any points should be redeemed and use a rate component to generate a bill calculation line with the credit amount.

The reason for this is that cancel/rebill logic is not straightforward. Algorithms executed during rate application should NOT perform any updates, such as updating the service credit membership balance. The balance should be updated using a bill segment freeze algorithm or a bill completion algorithm.

When a cancellation occurs, the service credit balance should be updated to reinstate the redeemed points. Again, this should occur when the cancellation is frozen or at bill completion time. If you perform a cancel/rebill, the calculation of the rebill segment does not have the up-to-date information about the service credit balance because the reinstatement of the points by the canceled segment has not occurred yet.

Designing Your Rate Options for Capital Credits

The capital credit allocation background process relies on certain data configuration in order to function correctly. This section identifies the required data setup.

Refer to [Allocating Capital Credits](#) for more information.

Contents

[Identifying SQ and Sales Information for Historical Bill Segments](#)
[Designing Bill Factors for Credit Allocation](#)

Identifying SQ and Sales Information for Historical Bill Segments

To allocate capital credits, the background process retrieves billing history for each SA that contributes to the membership for the given fiscal year. The process needs to calculate the service quantity (SQ) amount billed and the sales amount billed for the SA in that year.

Sales Amount. The sales amount refers to the monetary amount billed. For a capital credit allocation, this amount would generally exclude taxes and may exclude other line item amounts from the bill.

In order to calculate the amounts correctly, the background process must determine which bill calculation lines for each bill segment contain the SQ and/or sales information. Characteristics on the bill calculation lines identify which bill lines should be used.

The rest of this section uses examples to illustrate how you may configure your rate options to support this.

Contents

- Define Characteristics for SQ and Sales
- Define Rate Components
- Define Batch Control Parameters

Define Characteristics for SQ and Sales

Char Type for identifying bill lines that hold **SQ information**

- Char Type: CCA-SQ
- Description: Capital credit allocation usage
- Type of Char: Pre-defined
- Values: Y
- Char Entities: Rate Component, Bill Calculation Line

Char Type for identifying bill lines that hold **sales information**

- Char Type: CCA-SALES
- Description: Capital credit allocation sales
- Type of Char: Pre-defined
- Values: Y
- Char Entities: Rate Component, Bill Calculation Line

Define Rate Components

Identify each rate schedule that may be linked to a service agreement that contributes to a capital credit membership.

For each of these rate schedules, identify the rate components whose resulting bill calculation line will contain a billable service quantity that should be included in the SQ calculation for allocating capital credits. For each rate component that qualifies, define the **CCA-SQ** char type (defined above) and a char value of **Y**.

For each rate schedule, identify the rate components whose resulting bill calculation line amount should be included in the sales calculation for allocating capital credits. For each rate component that qualifies, define the **CCA-SALES** char type (defined above) and a char value of **Y**.

Characteristic Information. The system automatically copies characteristic info from a rate component to its resulting bill calculation line if the char type entities include both rate component and bill calc line.

Define Batch Control Parameters

The background process to allocate capital credits **CPCRALOC** receives the char type and char value to identify the bill calculation lines that contain the SQ and sales amounts. Once you have your characteristics defined, update your batch control to include these values as default parameter values.

Designing Bill Factors for Credit Allocation

The [capital credit allocation](#) process uses an allocation factor in its calculation. A typical capital credit membership may define multiple subcategories, meaning that allocation amounts are calculated each year for the multiple subcategories. The calculation is the same for each subcategory, but the allocation factor differs.

The process has been designed to calculate the allocation for a single subcategory. If your organization requires allocations calculated for multiple subcategories, the process must be run for each subcategory. The allocation process receives a bill factor as an input parameter. As a result, a different bill factor should be set up to define the allocation factor for each subcategory.

For each subcategory, the allocation factor may differ further for the type of customer. For example, the allocation for a commercial customer may differ from the allocation factor for a residential customer. The allocation background process expects the bill factor for each subcategory to define a characteristic type of revenue class. The process determines each service agreement's revenue class by looking at the value defined on its SA type.

Following is an example of bill factors set up for a capital credit membership with two subcategories: transportation and generation.

Contents

- [Characteristic Type for Allocation Bill Factor](#)
- [Bill Factor for Transportation](#)
- [Bill Factor for Generation](#)

Characteristic Type for Allocation Bill Factor

Char Type: REV-CLASS

Description: Revenue Class

Type of Char: Pre-defined

Values: (define all the valid revenue class values)

Char Entity: Bill Factor

Bill Factor for Transportation

Bill Factor Id: CCAF-TRANS

Description: Transportation Allocation Factor

Char Type: **REV-CLASS**

Char Source: **Characteristic Collection**

Char Values: (for each year, the new transportation allocation factor for each revenue class must be defined)

Bill Factor for Generation

Bill Factor Id: CCAF-GEN

Description: Generation Allocation Factor

Char Type: **REV-CLASS**

Char Source: **Characteristic Collection**

Char Values: (for each year, the new generation allocation factor for each revenue class must be defined)

The characteristic source is characteristic collection. It is the responsibility of the background process to determine the SA's revenue class and to pass this value into the bill factor routines to retrieve the correct bill factor value.

Estimating Allocation Factors. Often the company needs to estimate the allocation factors for the new fiscal year and may adjust the values several times until the calculated allocation amounts are satisfactory. Refer to [Allocating Capital Credit](#) for more information.

Partial Retirement

In the cooperative business, it is common to never retire certain capital credit allocation amounts. The amounts that do not retire should be assigned their own subcategory.

When executing the retirement background process, the subcategory to retire may be input to the process. If you have certain subcategories that you do not retire, you would simply run the background process for the subcategories that do retire.

Cooperatives typically retire amounts and transfer the amounts to a beneficiary when a member dies. This is known as "estate retirement". Refer to [Service Credits for Capital Credit Memberships](#) for more information. If your business practice designates that certain subcategories of allocated amounts do not get retired, this probably holds true for estate retirement as well. If that is the case, your membership inactivation algorithm should be designed to only retire the appropriate amounts by subcategory.

Interface Membership Information to a Third Party

For some memberships, you may accumulate points for a third party, for example accumulating frequent flier miles for an airline. For these types of memberships, you must interface the event information to the third party.

To interface information to a third party, you may choose one of the following options:

- Design an extract program to interface the information
- Use workflow and notification to interface the information via the XAI tool

Contents

[Interface Via an Extract Program](#)

[Use Workflow & Notification to Interface Info](#)

[Other Considerations For Interfacing Info to a Third Party](#)

Interface Via an Extract Program

The service credit event may indicate a batch code and batch run number. Design a program to extract event information to a third party. This extract program would select service credit events marked with its batch code and the current run number.

Your service credit event must define a completion algorithm that stamps the appropriate batch code and run number. The base product provides an algorithm type to perform this logic. Refer to [SCEC-BT](#) for more information.

Use Workflow & Notification to Interface Info

Workflow and notification allows you to send information to a third party via an extract program or via XAI. The service credit event has its own logic for interfacing via an extract program, so you would use workflow and notification only if you need to interface to the third party via XAI.

Refer to [How To Design Outgoing Messages](#) for more information.

Other Considerations For Interfacing Info to a Third Party

Because event information is extracted to a third party, you must consider how to handle adjustments to the event amount. For example, if your event is generated based on the creation of a bill segment, what should happen if that bill segment is cancelled?

You may want to prevent these types of events from getting canceled. Validation like this may be added via a user exit or using an event cancellation algorithm.

Sample Algorithms. The product does not provide any base algorithms to prevent an event from being canceled.

You should allow negative event amounts to be created so that this information may also be sent to the third party's system.

