PRIMAVERA

P6 Reporting Database Ver 3.0 Planning and Sizing

An Oracle White Paper
December 2011

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Contents

Introduction

Detailed planning is essential to successfully deploying the P6 Reporting Database and P6 Analytics. Both products have multiple components and dependencies. With a deep understanding of all the moving pieces, and a clear strategy to gather the critical requirements, it becomes easier to make these products an integral part of your project management infrastructure.

These products are essentially a generic data warehousing implementation. The process does not differ greatly from any custom data warehouse implementation in that it involves several databases, servers, and a controlling ETL process. Oracle Primavera provides the code to perform the ETL process.

When planning each deployment, you need to consider:

- The requirements for physical storage and CPU processing are very demanding.
- The implementation includes a star schema and an optional operational database
- The P6 Analytics module that contains a pre-built application including dashboards, reports and underlying metadata to provide end users with easier access to information.
- Differences in data volumes, frequency of data changes, and business requirements.

Implementation planning and sizing process are critical for the success of the P6 Analytics and the optional Reporting Database deployment and operations.

This document assumes a firm understanding of the P6 Reporting Database and P6 Analytics architecture while detailing the following tasks necessary for a successful deployment of the project management business intelligence:

1) Review the critical performance factors for the product.
2) Outline a methodology for planning an installation.
3) Determine the physical storage requirements of the data warehouse.
4) Address the server performance requirements of the ETL process.

Critical Performance Factors

Four Key Areas of the ETL

While the ETL process has many individual aspects, there are four general areas that will affect performance:

1) Pulling data between servers.
2) Managing the updates of the component databases.
3) Performing PL/SQL and direct SQL transformation operations on the database server.
4) Project management data dimension and fact table generations.

Prior versions (2.1 and earlier) work with Oracle Primavera P6 EPPM 8.0 or earlier. In those versions, nearly half of the critical fields for reporting are not physically stored in the P6 EPPM Database schema. Because of this, the P6 Integration API has to be called during the ETL process. This had a significant impact on performance and poses a key challenge to the P6 Data warehouse solution that other applications did not need to address.

Starting with version 2.2, the STAR schema and ODS work with Oracle Primavera P6 EPPM 8.1 and later versions, within which all the prior absent summary and hierarchical data have been persisted in the database. The Transform part of the ETL process is minimal, and the whole ETL performance is drastically improved over the prior versions.

Pulling Data between Servers

As with any ETL process, there are elements of data movement revolving around the E (Extract) and L (Load) in ETL. In a typical implementation architecture, the P6 EPPM Database and data warehouse (ODS/STAR) are deployed on separate physical servers. As a result, the bandwidth must be maximized and latency minimized between servers. Ideally, the servers reside in the same data center with gigabit ethernet connection between servers. (Oracle Primavera performance tests are performed with servers in a central data center with gigabit connections.) Throughput of server communication should be verified. This can be done by performing basic file copy, or FTP operations, between the servers.

The data movement processes is based on the standard SQL, either with INSERT-SELECT or CREATE TABLE AS (CTAS) syntax using an Oracle database link. While the Oracle RDBMS is efficient at moving data through the link, the overall performance is dependent on the physical network connection.

Merging Updates into Target Database

Current P6 Analytics and Reporting Database release (2.2 and later) significantly changes the method of updating tables incrementally in the target schema. This is the process by which the changes are merged into the base tables.

P6 Analytics and Reporting Databases leverage PL/SQL Bulk operations to process smaller batches while performing interim commits. This results in linear scaling of update operations (for example, if it takes five minutes to update one million rows, it will take ten minutes to update two million rows). Rows per second performance remains constant, regardless of amount of change or data size.

Many of the processes, including the PL/SQL Bulk operation processes, are run in parallel threads. The use of parallel PL/SQL processes not only increases the potential scalability of this process, but also the demand for CPU powers and network bandwidth on the server.

PL/SQL-based Transformations

Some of the transformation process is done with PL/SQL. The largest portion of this is referred to as *direct SQL*. These are SQL update statements run directly against tables to perform simple transformation that does not require business logic. These processes are generally executed in parallel, and are very CPU intensive, on the database server. With the large chunk of data processed, the disk IO performance is the key limiting factor. A high performing SAN system greatly improves the overall system response time.

Planning Process

Why Planning is Key

As previously mentioned, the product is essentially a data warehouse. Without proper planning, a successful deployment will be difficult to achieve. Achieving a successful implementation requires a structured approach that will give you the necessary insights to make correct decisions about the physical and logical aspects of the implementation. This section outlines the planning methodology to guide you through the process step-by-step, including:

1) Requirements Phase
2) Planning Phase
3) Testing Phase
4) Initial ETL Phase
5) Operational Phase

Requirements Phase

The first phase in any data warehouse implementation is to understand what the users of the system want to get from the solution. This includes:

- The types of reports

- Level of detail
- Time frame
- Data freshness

This information must be gathered before making any hardware or architecture decisions. Time spent at this phase will greatly reduce the risk during the rest of the implementation and the subsequent operation of the data warehouse.

A wide variety of reporting and analytics results can be achieved with the product. However, not all of these may be required in a given installation. There are two broad categories of reporting solutions:

- Operational
- Analytics/Business Intelligence.

Operational reporting covers the day-to-day, actionable reports used by project managers, resource managers, and other tactical personnel in the organization. This type of reporting is typically the traditional, tabular reporting that is repeated on a daily basis. A key consideration of operational reporting is the scheduling and delivery of the reports. The combination of the spread data persistency in the P6 EPPM schema and Oracle BI Publisher addresses the scheduling, execution, and delivery of the reports.

Using Oracle Database Standard Edition or Enterprise Edition (without partitioning) for the Star Schema

If you are connecting multiple P6 EPPM data sources to your STAR schema, Oracle Database Standard Edition or Enterprise Edition (without partitioning) is not supported. Oracle Enterprise Edition with partitioning is required.

If your STAR schema is installed using Oracle Standard or Enterprise Edition(without partitioning), the P6 EPPM database should fall within the range of a small-sized database as defined in the 'Planning Revisited' section of this document. Partitioning helps keep performance consistent over time for each ETL run. This is particularly important when enabling activity or WBS-level history. If you choose not to use partitioning, activity and WBS-level history gathering is not recommended. If the P6 EPPM database is larger than the defined criteria of a small-sized database, Oracle Database Enterprise Edition with Partitioning is strongly recommended.

Without partitioning, the amount of time it takes to run the ETL process will increase over time, particularly on steps 20 and 21. Track the row counts in the w_project_history_f, w_wbs_history_f, w_activity_history_f. As the size of these tables increases the need for partitioning will also increase. These run times can be monitored in the staretl.html and staretlprocess.log located in the <installation directory>\log folder. Oracle Database Enterprise Edition with Partitioning is the solution for growth over time within the history tables.

Key Questions to Ask about Operational Reporting

- **When will reports be run?** Perhaps the individual users will need to execute on demand. Often reports are needed prior to the start of work on a given day. These considerations will affect the timeliness of the data. The latest version of the ETL process is a batch process. It can be run one or more times per day, and limited by the duration of each run which is proportional to the data volume and date range selected by the customers.
- **How will reports be delivered?** Getting the right reports to the end users at the right time is the key measuring metric of success. Oracle BI Publisher offers multiple ways to deliver reports from the Oracle ODS Database. These include email, HTTP, WEBDAV, direct printing, et al. The logistics of setting up these delivery methods must be considered during the planning process. Moreover, any leading third-party reporting tools can be utilized to generate and deliver the reports.
- **What will the reporting load be on Oracle ODS Database?** One of the major considerations affecting subsequent decisions will be the load on the reporting server.
- **Oracle ODS Database:** Queries will be executed against the Oracle ODS database to fulfill reporting requests. This usage will likely peak during specific times of the day. This peak load must be considered as the requirement. Since the exact types of queries are unknown at this point, it is important to gain a broad understanding of what the load will be:

- **How many users are accessing the database at the same time?** This will determine the maximum load on the database server.
- **Is the reporting on individual projects or across the entire database?** Aggregate queries will put more load on the server than project-specific queries.
- **Is the reporting done in batch or interactively?** More interactive reporting will increase the demands on both the server CPU and I/O subsystem.

- Many of the same considerations previously mentioned for the database should be applied to the Oracle BI Publisher reporting server.

Operational reporting has the distinct advantage of being very well defined and constant. On a day-to-day basis, the reporting load will be fairly consistent. This is not the case for P6 Analytics. P6 Analytics reporting is, by nature, dynamic. The Oracle star Database schema and OBI integration was designed to allow a very rich environment. This means that the daily load on the data warehouse server and OBI will vary greatly.

The BI Server component is capable of robust caching of query results, which can greatly mitigate performance concerns. The effectiveness of caching depends on how much users share security. If every user's project access is distinct (including the level of access to cost fields) then the cache will only be effective for each user individually.

Key Questions to Ask about P6 Analytics/Business Intelligence

- **Who will access P6 Analytics?** A diverse set of users from the CEO to resource/project managers may require access to P6 Analytics. Each may have different requirements and use cases to consider.
- **What are the default ways of filtering?** By default, user requests for analytic information will include all the data accessible by that user. That may be more time consuming, and may include more information than necessary. Consider ways of filtering data, such as Project Codes and Portfolios.
- **What codes are used for reporting?** While the Oracle ODS Database includes all the data from the P6 EPPM database, the Oracle Star Database schema includes only a subset of codes among activity, resource, and project codes. Before moving forward, you must determine which codes are critical for analysis.

Planning Phase

Once the requirements of the resulting data warehouse are well understood, the planning for the installation can really begin. As with any data warehouse, physical storage demands are high. Because the calculation process places a unique demand on the ETL, this data warehouse implementation may require higher CPU/memory requirements. More detailed information will be provided in subsequent sections of this document. First we will examine the high-level aspects of planning the implementation.

Full ETL Process Only

Unlike the prior release, which consisted of initial full ETL and subsequent incremental ETL processes, this release runs only in full ETL mode. Since the data has been pre-processed in the PMDB schema, direct loads of the data into ODS and STAR are much more efficient.

Monitoring Usage

Understanding the data growth to key tables is critical to the performance of the usage monitoring process. This can be easily gathered on a daily basis from the audit columns on each table (for example, CREATE_DATE) and from the REFRDEL table. The amount of change should be monitored for several weeks, and periodic peak activity should be noted. The peak usage times are important to keep track of, since they will be used as the basis of hardware decisions. The following tables should be monitored:

- PROJECT
- TASK
- TASKACTV
- TASKRSRC
- TASKMEMO
- UDFVALUE
- RSRCHOUR
- PROJWBS

Basic Monitoring

Growth to the table (insert) can be counted with the following query. (This assumes running the query at the end of the day to get all the inserts from that day.) This query is repeated for all the critical tables.

```
SELECT count(*)
FROM <table>
WHERE create_date > trunc(sysdate)
```

Delete rows are queried from REFRDEL in a single step

```
SELECT TABLE_NAME, count(*)
FROM REFRDEL
WHERE delete_date > (sysdate)
GROUP BY TABLE_NAME
ORDER BY TABLE_NAME
```

A more precise picture of usage can be gained using P6 Auditing. While this can be used, it is not yet necessary since at this point only a general understanding of the amount of change is required.

Physical Sizing

Overview of Physical Space Usage

The physical space requirements of the data warehouse consist of more than just copies of the project management data. Space requirements will vary with the amount of data processed from the P6 EPPM Database. The system uses space for the following types of data:

- **Core Project Management Data** - This includes all the physical fields that exist in the P6 EPPM Database. This is approximately all the data in the P6 EPPM Database.
- **Logical Fields** - The fields become physically stored as part of the P6 EPPM Database in Px tablespace. While this is less than the size of the P6 EPPM Database, it may be as much as 50% of the total.
- **Fact/Spread Data** - Spread and Fact data total size depend on the number of activities and resource assignments, the average length of activities and the total window (date range) of the data warehouse. Because of this, it will be treated as a distinct group. It is part of the ODS and is fundamental to the dimensional schema (STAR).
- **Other ETL Tables** - There is some database space usage specific to the ETL process. This space is trivial relative to the core P6 EPPM data.
- **ETL Process Installation** - This includes shell scripts, sql files and JAR files.

Table Partition for Multiple Data Source and History Data

- **Purpose of Table Partition** - improved performance, and ease of data management
- **Multiple Data Source** - In this release, the STAR support multiple P6 databases as the data source.

  By default, the related tables are value partitioned based on the DATASOURCE_ID.

  The partitioned tables are:

  W_ACTIVITY_D
  W_ACTIVITY_RISK_D
  W_ACTIVITY_SPREAD_F
  W_COST_ACCOUNT_D
  W_EPS_D
  W_EPS_HIERARCHY_D
  W_OBS_D

W_PROJECT_D
W_PROJECT_PORTFOLIO_D
W_PROJECT_SECURITY_S
W_RESOURCE_ASSIGNMENT_SPREAD_F
W_RESOURCE_D
W_RESOURCE_LIMIT_F
W_RESOURCE_LIMIT_SUM_F
W_RESOURCE_ROLE_D
W_RESOURCE_SECURITY_S
W_RESOURCE_TEAM_D
W_RISK_D
W_ROLE_D
W_UDF_ACTIVITY_F
W_UDF_PROJECT_F
W_UDF_WBS_F
W_USER_S
W_WBS_D
W_WBS_HIERARCHY_D

- **History Data** - History data can grow quickly, especially if the history interval is set to Weekly, and the history level is set to Activity.
  By default, these tables are range partitioned based on the DAY_WID value, and then sub partitioned by the value of the DATASOURCE_ID. Allocation of separate tablespace for each partition is recommended. The 3 history tables are:

  Project history: W_PROJECT_HISTORY_F
  WBS history: W_WBS_HISTORY_F
  Activity history: W_ACTIVITY_HISTORY_F

  The indexes on these tables are LOCAL indexes per each partition.

- **Number of Partitions** - 3 partitions are created by default. If more partitions are needed, the DBAs should execute the partition split command. Examples are given below.

- **Value Partition Addition**:
  alter table W_ACTIVITY_D add partition P4 values (4) tablespace star_hst1;

- **Range Partition Split**:

```
alter table w_project_history_f
split partition R3 at (11870)
into (
    partition R3 tablespace star_hst1,
    partition R4 tablespace star_hst1
);
```

The value for the split point can be determined by query the ROW_WID column in W_DAY_D table.

Physical Components

There are three physical components to consider in sizing the data warehouse. When discussing physical components further, they will be treated as separate instances of the Oracle database, or a physical server, although this is not necessarily required (see Processor Sizing for more details). There is no direct size impact on the P6 EPPM Database beyond normal space usage, and we will not consider it as one of the physical components to be sized. The components are:

- ETL Process Server
- Oracle Operational Data Store Database
- Oracle Star Database

ETL Process Server

While this server is the central controller of the ETL process, it represents only a small portion of physical space used. The only files, other than the ETL process files, are log files from each run and the Temporary Flat files for xlat (language translations). Unlike the versions prior to 2.2, the size for the flat files are minimal in this version.

Calculating Spread Sizes

The actual number of buckets depends on factors such as calendar work days, large differences in dates relative to the data date, ETL time duration, and the general exclusion of zero value rows. A quick calculation uses a "best guess" on the average number of days for activities and resource assignments.

```
    Total Activities: 1,000,000 X Average Activity Length: 5 = Total
Spreads: 5,000,000
```

Queries for Spread Sizing

Activity Spread Estimate Based on Median Activity Length

```
select
median(
greatest(nvl(target_end_date,to_date('12122000','mmddyyyy'))
 ,nvl(act_end_date,to_date('12122000','mmddyyyy'))
             ,nvl(reend_date,to_date('12122000','mmddyyyy'))

,nvl(rem_late_end_date,to_date('12122000','mmddyyyy')) )
-
least(nvl(target_start_date,to_date('12122199','mmddyyyy'))
             ,nvl(act_start_date,to_date('12122199','mmddyyyy'))
             ,nvl(restart_date,to_date('12122199','mmddyyyy'))

,nvl(rem_late_start_date,to_date('12122199','mmddyyyy')) )
) * count(*) Spread_Rows
from task t inner join project p on p.proj_id = t.proj_id and
orig_proj_id is null
where task_type in ('TT_Task','TT_Rsrc')
```

Resource Assignment Spread Estimate Based on Median Activity Length

```
select
median(
greatest(nvl(target_end_date,to_date('12122000','mmddyyyy'))
             ,nvl(act_end_date,to_date('12122000','mmddyyyy'))
             ,nvl(reend_date,to_date('12122000','mmddyyyy'))
             ,nvl(rem_late_end_date,to_date('12122000','mmddyyyy
')) )
-
least(nvl(target_start_date,to_date('12122199','mmddyyyy'))
             ,nvl(act_start_date,to_date('12122199','mmddyyyy'))
             ,nvl(restart_date,to_date('12122199','mmddyyyy'))
             ,nvl(rem_late_start_date,to_date('12122199','mmddyy
yy')) )
) * count(*) Spread_Rows
from taskrsrc tr inner join project p on p.proj_id = tr.proj_id and
```

```
orig_proj_id is null
inner join task t on t.task_id = tr.task_id
where task_type in ('TT_Task','TT_Rsrc')
```

Oracle ODS Database

The Oracle ODS database is an optional target database for operational level reporting. It materialized the views of the P6 EPPM base tables and their correspondent spread data tables. The space usage for the Oracle ODS Database can be derived from a combination of the size of the PMDB database, and the size of the spread data. The Oracle ODS Database has the following types of table data:

- **Spread Tables** - This is a combination of the detailed, daily spread data and aggregate tables.
- **Hierarchy Tables** - These tables map the underlying hierarchical relationships (such as EPS, WBS, etc.).

Indexing in the Oracle ODS Database defaults to the same indexing as the P6 EPPM Database. This should be augmented and adjusted based on site specific reporting needs.

Estimating the Size of the Oracle ODS Database

The estimated size of the Oracle ODS Database is derived from two primary components: the estimated size of PMDB project data and the estimated size of the spread data. The majority of Oracle ODS Database data is simply a copy of the P6 EPPM Database tables. The remaining space usage comes mostly from the ActivitySpread and ResourceAssignmentSpread tables. Any remaining data will be estimated as a percentage of the spread data (including aggregate spread tables and hierarchies).

Data Component	Calculation	Rows	Size Example
ActivitySpread	300 bytes/row	5,000,000 x 2*	3.0 GB
ResourceSpread	175 bytes/row	5,000,000 x 2*	1.6 GB
Other	30% of Spreads	n/a	1.4 GB
Total			6 GB

* Temporary data stored during loading process results in copies of spread data during initial ETL

Oracle Star Database

The Oracle Star Database contains a dimensional data model that includes four fact tables (plus the history fact tables if configured) and the supporting dimensions. Excluding the configurable history data tables, the STAR Database will contain fewer rows because baseline projects are not directly accessible. In general, the Oracle STAR database is still much smaller than Oracle ODS Database.

The fact data represents the largest portion of data in the Oracle STAR Database. As with any STAR schema, this data is the most granular data and, by default, there are no aggregate tables built to support rollup queries.

Depending on the History Interval and History Level settings, the Activity and WBS history tables are likely to be the largest.

The primary two fact tables contain activity and resource assignment spread data respectively. The size of these tables will be the same as the corresponding Oracle ODS Database tables (ActivitySpread and ResourceAssignmentSpread).

The next largest fact table contains resource utilization data (W_RESOURCE_LIMIT_F). This differs from other fact tables in that the data size is not a function of the number or size of projects. Instead, it is a function of the number of resources in the database, and the size of the data warehouse reporting window. There is a daily value for everyday of the reporting period, and for each resource. For example, if the reporting window spans five (5) years (1,825 days), and there are 1,000 resources in the database, the total records in the fact table will be 1,825,000.

The final fact table is the smallest, and it has only project-level data. The difference is that this table is a trending table, with snapshots of the data over time. The amount of snapshots depends on the interval chosen during installation (weekly, monthly, financial period). The granularity of this fact table is only down to the project-level; it contains no spread information. Calculate the number of rows using the total non-baseline projects times the number of snapshots. This will grow over time, so the yearly total for a 10,000 project database with weekly snapshots will be 520,000 rows.

Estimating the Size of Oracle STAR Database

Only the fact tables will be considered for Oracle Star Database sizing purposes because they are responsible for most of the data. Of the four fact tables in Star, two of the tables (W_ACTIVITY_SPREAD_F and W_RESOURCE_ASSIGNMENT_SPREAD_F) are identical to the equivalent spread tables in ODS. See Estimate the Size of ODS for details. The rows for the remaining fact tables were calculated in the previous sections. Spread and resource limit data is initially loaded into holding tables (_FS suffix), so sizes are doubled for these tables.

Data Component	Calculation	Rows	Size Example
W_ACTIVITYSPREAD_F	300 bytes/row	5,000,000 x 2	3.0 GB
W_RESOURCE_ASSIGNMENT_SPREAD_F	175 bytes/row	5,000,000 x 2	1.6 GB
W_RESOURCE_LIMIT_F	70 bytes/row	1,825,000 x 2	0.125 GB
Dimensional and Temporary	20% of Spread	n/a	0.9 GB
W_Project_History_F	265 bytes/row	6000	0.04 GB
Total			5.6 GB

Note: The above sizing does not include the W_Activity_History_F and W_WBS_History_F tables, which will be the largest if the History Interval is set at the Weekly, and the History Level is set at the Activity.

Physical Hardware

When evaluating the physical hardware requirements, there are two distinct areas to consider.

- The performance of the ETL process
- The performance and concurrency of the online reporting solution.

While the ETL process is fixed regarding concurrency, the reporting needs will vary greatly. The demands on P6 Reporting Database and P6 Analytics may change from day-to-day. For performance sizing of OBI, please refer to the technical documents for the specific component (BI Publisher, BI Server, and Answers/Dashboards). This document will focus on the performance of the ETL process and queries generated against the warehouse databases (ODS and Star).

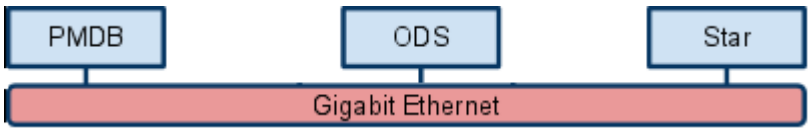
Consideration for ETL Scalability and Performance

The ETL process for P6 Reporting Database R3.0 was designed with multi-core processor systems in mind. Instead of a serialized process, Java is used to create a multi-threaded process to run concurrent threads of SQL*Plus, SQL*Loader, and the P6 Integration API. This can result in multiple runnable threads on the various servers. This also means that the process can be adversely affected by having to compete with other applications sharing the same resources. Therefore, an ideal configuration would have dedicated cores available for each component in the process. Moreover, dedicated database instance with sufficient SGA allocation and high throughput IO system is crucial for the ETL performance.

This is an ideal configuration that is meant to minimize contention. By dedicating resources to each of the physical components, concurrent performance will be maximized during peak usage. Different steps in the ETL process put a variety of load on each component. In this release, there is no concurrent processing occurring simultaneously on both ODS and Star servers. Therefore, from the ETL perspective, they could share the same physical hardware. Concurrent reporting usage should be considered in determining the correct CPU requirements for ODS and Star.

Network

While there is a distinct advantage to separating the components, there is an underlying assumption that the network connections between servers have very high bandwidth and very low latency. These servers are preferred to be on the same local area network, with gigabit connections. Any increase in latency will have a significant effect on the ETL performance.



In this release, we have enabled the ability to support multiple P6 EPPM data sources for a single data warehouse instance. With the internationalization of the corporate operations, it's likely that the data sources might be distributed across geographic regions, and connected via corporate WAN instead of the high-speed, low-latency LAN. Data extraction is a high volume operation, and therefore the bandwidth and network latency directly impact the performance. We have a 10-10 rule for the corporate WAN connections:

- Minimum network bandwidth: 10 mbps
- Maximum network latency: 10 ms

For a more successful and satisfactory deployment, the network parameters should be much better than these minimum requirements.

Memory

With a large number of parallel processes running on large sets of data, the demands on memory will be very high. The components of the data warehouse system should only be run on 64-bit operating systems to allow for large memory allocations. Constraining memory quickly reduces performance.

The database servers need both block buffer and individual process memory. These servers should always be setup using Dedicated Server (not Shared Server). For an Oracle 11g database, the recommend minimum MEMORY_TARGET is 2 GB to 10GB depending of the P6 EPPM Database (for 10g, set SGA_TARGET to the same minimum value).

The Java process on the ETL Process Server is running multiple threads in the same process. Only run with a 64-bit version of the JRE to allow for larger memory allocation. The maximum memory allocation for the Java process is configurable during setup (Max Heap Size). The default is 1 GB. This may be inadequate for many datasets, however, and may cause failures in the ETLCalc process. Start with a minimum of 4 GB of memory for the Java process.

Extract and Load

Customized ETL processes are implemented for the data extraction and loading. These processes are completely PL/SQL based, and use only resources on the database server. At any given time, there may be multiple database threads running PL/SQL. These are bulk PL/SQL inserts, updates and deletes making this very resource intensive. This is the reason for recommending 10 or more cores on the database server, so each thread has an available CPU in which to run. The performance of each individual core will ultimately determine the speed (rows/second) of the extract and load processes.

Summary of Physical Hardware Sizing

When planning for the physical hardware for the P6 Reporting Database, consider the following basic guidelines.

Size of P6 EPPM Database

Overall, the size of the database is going to play a large role in the performance. There is a direct relationship between the database size and the performance of the full ETL process, since all records must be processed. There is some relationship between database size and the ETL process, since more project data will likely translate into more usage and more records in the database.

Amount of Change

Day-to-day, it is the performance of the Px service that is the primary concern. While database size plays a role, it is ultimately the volume of change that will affect performance. Even a small user community can generate tremendous amount of change with functions like copy/paste, delete project, and create baseline as well as simple changes to global resources like calendars. Careful monitoring of the amount of change prior to installation will enable you to better plan.

In this version of the P6 Reporting dabases, ODS and STAR are always full ETL processes. Fresh data is pulled from the P6 EPPM schema. Existing history table data are preserved during the ETL process. It's up to the customer to implement a history data purge process.

Complexity of Project Data

Consider the actual data in the P6 EPPM Database. A very complex EPS/WBS structure will greatly increase the processing time because these hierarchical elements require global processing even when the changes are small. The average size of activities can increase the memory requirements of the calculation process. Larger projects will cause more volume of change because the entire project is recalculated based on any changes within the project. This does not include changes to non-scheduling data, such as activity codes and UDFs.

Expectations and Operations

There may be an expectation that for operational purposes the ETL needs to complete in a smaller time frame. These considerations may increase the hardware requirements.

Following is the hardware recommendations based on P6 EPPM Database size. For Database Deployment size refer to Timing Estimation Section.

Small Database Deployment

ETL Server, Oracle ODS Database (and Oracle STAR Database) on same box

CPU	Quad Core 3.46 GHz or equivalent
RAM	4 GB without STAR; 8 GB with STAR

Note:

1. This assumes light Business Analytics (Oracle STAR Database) usage. Otherwise, a separate server should be allocated for the Oracle STAR Database.
2. 4 GB RAM should be added if OBIEE installed on the same box.

Medium Database Deployment

Oracle ODS Database

CPU	Quad Core 3.46 GHz or equivalent
RAM	4 GB

ETL Server, Oracle STAR Database

CPU	Quad Core 3.46 GHz or equivalent
RAM	8 GB

Large Database Deployment

ETL Server

CPU	4 Core 3.46 GHz or equivalent
RAM	8 GB

Oracle ODS Database

CPU	8 Core 3.46 GHz or equivalent
RAM	16 GB

Oracle STAR Database

CPU	8 Core 3.46 GHz or equivalent
RAM	16 GB

Recommendations Summary

Following is the overall summary of recommendations for ETL Server and Database Servers

ETL server

4 Cores for ETL Process Server

8 -10 Cores for each Database Server
Gigabit Ethernet connection between servers and in same datacenter
64 bit OS
4 GB Minimum of memory for the Java process.

Database Servers

	Oracle ODS Database	Oracle Star Database
TableSpace Autoextent	Yes	Yes
Recommended Temp Tablespace	Minimum 2 files with file size set at OS max	Minimum 2 files with file size set at OS max
Recommended UNDO Tablespace	Minimum 2 files with file size set at OS max	Minimum 2 files with file size set at OS max
Minimum Space	2.5 X P6 EPPM Database	Same as P6 EPPM Database
SGA	8GB	8GB

Planning Revisited

Timing Estimation

The following timing numbers are only an approximation. The actual numbers for your deployment depend on the data structure and the hardware used.

		Deployment Categories		
		Small	**Medium**	**Large**
Number of Objects	Projects	200	1,000	50,000
	Activities	100,000	1,000,000	5,000,000
	Activities per project	5,000	10,000	20,000
	Resources	500	1,000	4,000
	Resource Assignments	100,000	1,000,000	5,000,000
	Resource Assignments per project	5,000	10,000	20,000
	Full ETL Approximate completion time	< 1 hr	~ 7 hours	~ 10 hours

Conclusion

Following a systematic approach to evaluating, planning, and testing the architecture for your P6 data warehouse is the only way to assure a successful implementation. With careful examination of the requirements, data sizing and user activity the appropriate hardware choices can be made early in the process.

White Paper Title P6 Reporting Database Planning and Sizing
October 2011

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle Corporation provides the software that powers the internet.