

Oracle® Insurance Policy Administration

Cycle Instructions

Version 9.1.0.0.0

Part number: 15811-01

November 2009

Copyright © 2009, 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 USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 that may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software 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.

Where an Oracle offering includes third party content or software, we may be required to include related notices. For information on third party notices and the software and related documentation in connection with which they need to be included, please contact the attorney from the Development and Strategic Initiatives Legal Group that supports the development team for the Oracle offering. Contact information can be found on the Attorney Contact Chart.

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your beta trial agreement only. 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 in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

Table of Contents

| | |
|---|-----------|
| CYCLE INTRODUCTION | 4 |
| BATCH PROCESSING FOR ACTIVITY PROCESSING | 4 |
| RUNNING COMMANDS OF ACTIVITIES TO PROCESS | 4 |
| HIGH LEVEL PARTS TO RUN CYCLE | 4 |
| STARTING UP CYCLE | 5 |
| SETTING CYCLE START-UP PROPERTIES..... | 5 |
| <i>Explanation of Files</i> | 5 |
| OBTAINING THE NECESSARY .JAR FILES INTO LIB FOLDER | 7 |
| STARTING THE CYCLE SERVICE APPLICATION | 8 |
| STARTING CYCLE USING WINDOWS | 8 |
| STARTING CYCLE USING LINUX/SOLARIS..... | 9 |
| RUNNING CYCLE..... | 10 |
| STEPS TO RUN CYCLE COMMANDS | 10 |
| <i>Cycle Commands</i> | 10 |
| CYCLE TABLES | 13 |

Cycle Introduction

Oracle Insurance Policy Administration (OIPA) provides a subsystem for batch processing of insurance transactions called Cycle. Cycle is a high-performance distributed subsystem designed to process as many pending transactions as possible in the shortest amount of time. By using concurrency techniques, multiple threads, automatic failover, automatic scaling and real time configuration changes, OIPA provides you with a robust cycle solution. Various available commands allow you to run and view cycle(s) according to your business needs. Cycle is also used to advance the system date stored in the AsSystemDate table.

Batch Processing for Activity Processing

When using OIPA cycle for pending activity processing, you may configure activities to process according to OIPA system levels. Applicable levels for processing batch activities are:

- Company
- Client
- Plan
- Policy

Running Commands of Activities to Process

Additionally OIPA can process these activities according to their effective dates, processing order and the current system date. OIPA cycle separates activities to process according to date as follows:

- Pre-Company
- Pre-Client
- Pre-Plan
- Post-Company
- Post-Client
- Post-Plan
- Policy

A full explanation of how processing is handled is explained below in the [Cycle Command](#) section.

High Level Parts to Run Cycle

There are two parts involved in setting-up the cycle service. First you will set-up the cycle properties and then you will start cycle. The files needed to start cycle are in the cycle.service and cycle.client folder with the application files.

Note: Batch Screens in OIPA are used for manually processing or spawning the processing of multiple activities in one activity at one time for one policy, where cycle runs all pending activities on various policies at a specified point in time.

Starting Up Cycle

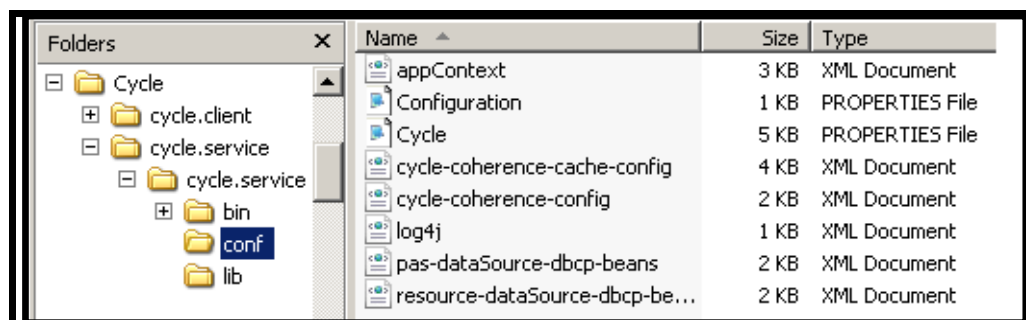
The initial set-up of cycle requires in-depth knowledge of the following areas:

- how cycle splits activities to process
- the server cluster that will be used
- the number of threads that should be designated for optimal performance.

You should not run cycle or change any of the properties unless you have in-depth knowledge. Prior to starting the cycle service, you will want to update properties in various files found in the conf folder.

Setting Cycle Start-Up Properties

There are various files that need to be updated to reflect preferences, database settings and various locations needed to start-up cycle. These files are found in the **cycle** folder in the Oracle Insurance Policy Administration application, which can be downloaded from E-delivery. Once you have downloaded the zip file, open the **OIPA_version number** folder. Open the cycle folder and extract the cycle.service zip file. Inside that file you will find a config folder. Open the config folder to access the necessary cycle files.



Cycle.service folder

Explanation of Files

- **Cycle.PROPERTIES** – The first three properties to set in this file relate to cycle and control the number of threads, the time intervals cycle will check for work, and the maximum number of cycle tasks that can be run simultaneously. The rest of the properties in this file should match what are in the OIPA properties file. There are descriptions of each property in the actual Cycle.PROPERTIES file.
- **Configuration.PROPERTIES** – By default this file points to the Cycle.PROPERTIES file. You can point to other .PROPERTIES files if necessary by editing this file.
- **log4j.xml** - This file controls the level of detail that reports to the log files. Only alter this file if you have an in-depth understanding of this technology.

- **pas-dataSource-dbc-p-beans.xml** – Contains the log-in information to the database you are connecting to. This will need to be changed accordingly. See the next section, [Obtaining the Necessary .jar Files into lib folder](#), which will have specifics on the driver information.
- **resource-dataSource.dbc-p-beans.xml** – Contains the log-in information to the database you are connecting to for the international database. This will need to be changed accordingly. See the next section, [Obtaining the Necessary .jar Files into lib folder](#), which will have specifics on the driver information.
- **cycle-coherence-config.xml** – Controls the clusters found in coherence. Only if you understand coherence should you alter this file. The information here is based how your installation was done. The well known address is how you filter out what machines are allowed to join the cycle cluster. You will receive an error if you try to access and you are not on the cluster.
- **cycle-coherence-cache-config.xml** – Default file defines how we configured our coherence caches in terms of best practices.
- **appContext.xml** – Do not alter this file.

Obtaining the Necessary .jar Files into lib folder

Before starting cycle, you must obtain the necessary proprietary and open source executable .jar files, which are required. Then copy them into the lib folder. There are different jar files that you will need to include depending on whether you are using DB2 or SQL for your database. No jar files need to be downloaded if you are using an Oracle database.

Using DB2 Database

The jar files that need to be copied in the lib folder:

- db2jcc
- db2jcc_license_cisuz
- db2jcc_license_cu

Note: **db2jcc**, **db2jcc_license_cisuz**, and **db2jcc_license_cu** are included with the purchase of the DB2 software. These files are not available for download. Contact your IT department if you need assistance locating these files.

Using SQLServer Database

The jar files that need to be downloaded in the lib folder:

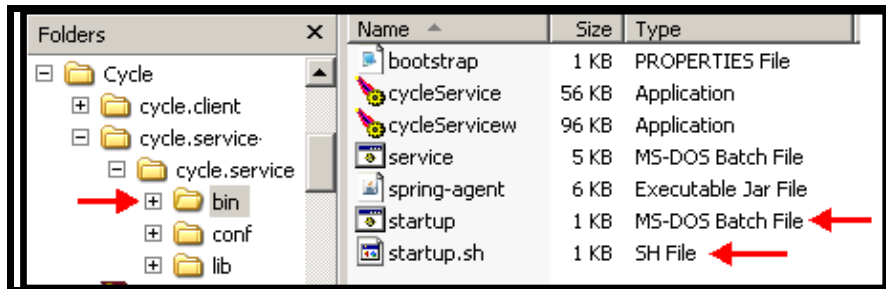
- jtds

Note: Download **jtds** from the following site: <http://sourceforge.net/projects/jtds/>

1. Select the **Download jtds – SQL Server and Sybase JDBC driver** link.
2. Scroll down to the area that lists the previous File Releases and download the distribution file for version 1.2 (**jtds-1.2-dist.zip**). Save the download .zip file to the temporary directory created to store the .zip files.
3. Open the downloaded .zip file and extract the file **jtds-1.2.jar** from the root of the .zip file.
4. Rename this file **jtds.jar**.

Starting the Cycle Service Application

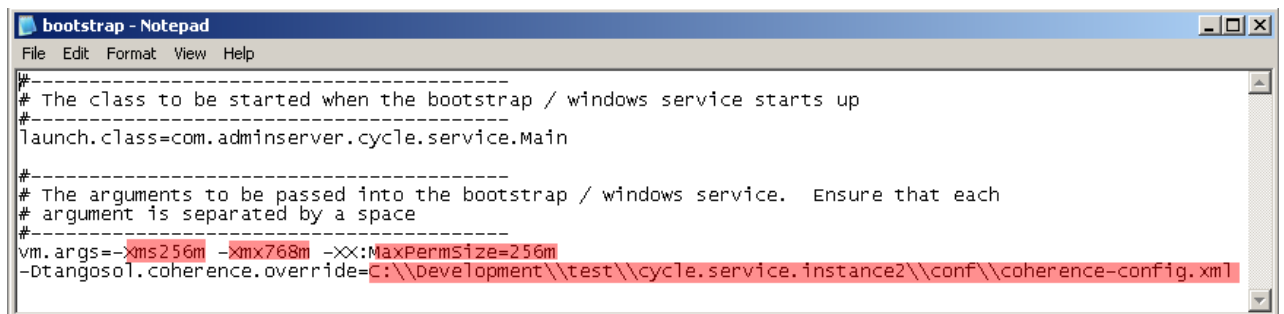
To start up the cycle server, select either **startup.bat** or **startup.sh**, depending on whether you are using Microsoft® Windows or Unix/Linux. See the directions below that are necessary for starting up cycle.



Cycle start-up files

Starting Cycle Using Windows

Prior to starting cycle, you must go in the bootstrap.PROPERTIES as this is what is loaded when registered as a Microsoft® Windows service. Modify the arguments such as allotted memory or where the coherence-config.xml file is located.



Bootstrap.PROPERTIES file

If you are using Windows, you should set-up cycleService as a Windows service. After installing it is important to understand that if you want this to run automatically you must use your Administrative tools to do so. When running cycle as a Windows service, a logs directory is created and the log files are stored in this directory. Log files will contain messaging and errors that occurred.

Steps to install:

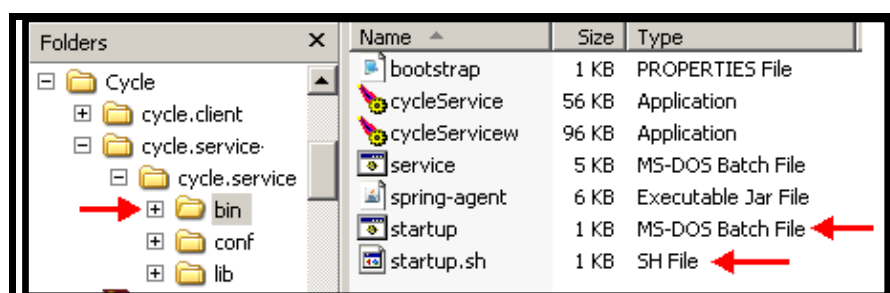
1. Open a command line.
2. Navigate to the path where the application is installed and to the cycle.agent\bin folder and type 'service install'.

Steps to remove:

1. Open a command line.
2. Navigate to the path where the application is installed and to the cycle.agent\bin folder and type 'service remove'.

Starting Cycle Using Linux/Solaris

The shell script file, startup.sh, to start the cycle service application is located in the *service/bin* subdirectory of the Cycle service installation directory. It may be run manually or scheduled using any standard scheduling utility. The configuration file *bootstrap.properties* is not necessary for this environment.



Cycle start-up files

Running Cycle

Running cycle allows you to set how you want a set of pending activities to process. You must use the run.bat file for a Microsoft ® Windows environment or the run.sh for a Unix/Linux environment to run the client.

Steps to Run Cycle Commands

1. Open a command line.
2. Navigate to the path where the application is installed and then to the cycle.client folder. Type 'run'. This will run either the run.sh or run.bat, depending on your environment.
3. Type in the letter or number associated with the command you want to run. Explanations of the commands are below.
4. Refer to the log files to see messages and errors.

```

C:\WINDOWS\system32\cmd.exe - run
2009-06-09 11:01:35.116/17.315 Oracle Coherence 3.4.2/411 <Info> <thread=main, member=n/a>: Loaded o
perational overrides from file "C:\Development\test\cycle.client\conf\cycle-coherence-config.xml"
2009-06-09 11:01:35.116/17.315 Oracle Coherence 3.4.2/411 <D5> <thread=main, member=n/a>: Optional c
onfiguration override "/custom-mbeans.xml" is not specified

Oracle Coherence Version 3.4.2/411
Grid Edition: Development mode
Copyright (c) 2000-2009 Oracle. All rights reserved.

2009-06-09 11:01:35.647/17.846 Oracle Coherence GE 3.4.2/411 <Info> <thread=main, member=n/a>: Load
d cache configuration from resource "file:/C:/Development/test/cycle.client/conf/cycle-coherence-cac
he-config.xml"
2009-06-09 11:01:37.049/19.248 Oracle Coherence GE 3.4.2/411 <D5> <thread=Cluster, member=n/a>: Serv
): Asking member 1 for 128 out of 128 primary partitions
2009-06-09 11:01:38.721/20.920 Oracle Coherence GE 3.4.2/411 <D5> <thread=DistributedCache, member=2
): Deferring the distribution due to 49 pending configuration updates
2009-06-09 11:01:39.292/21.491 Oracle Coherence GE 3.4.2/411 <D5> <thread=TcpRingListener, member=2>
: TcpRing: connecting to member 1 using TcpSocket{State=STATE_OPEN, Socket=Socket[addr=/10.1.100.98,
port=1685, localport=80891]}
2009-06-09 11:01:43.788/25.987 Oracle Coherence GE 3.4.2/411 <D5> <thread=Invocation:InvocationServi
ce, member=2>: Service InvocationService joined the cluster with senior service member 1

*****
OIPA Cycle Command Interface
Menu:
P - Pause
R - Resume
D - Advance System Date
A - Abort Current Cycle
Q - Exit
01 - Begin Pre-Company
02 - Begin Pre-Plan
03 - Begin Policy
04 - Begin Post-Plan
05 - Begin Post-Company
06 - Begin Plan Status
07 - Begin Pre-Client
08 - Begin Post-Client
*****
ENTER COMMAND \>03
  
```

After running the menu options will appear

Enter the applicable command

Command line with Cycle command

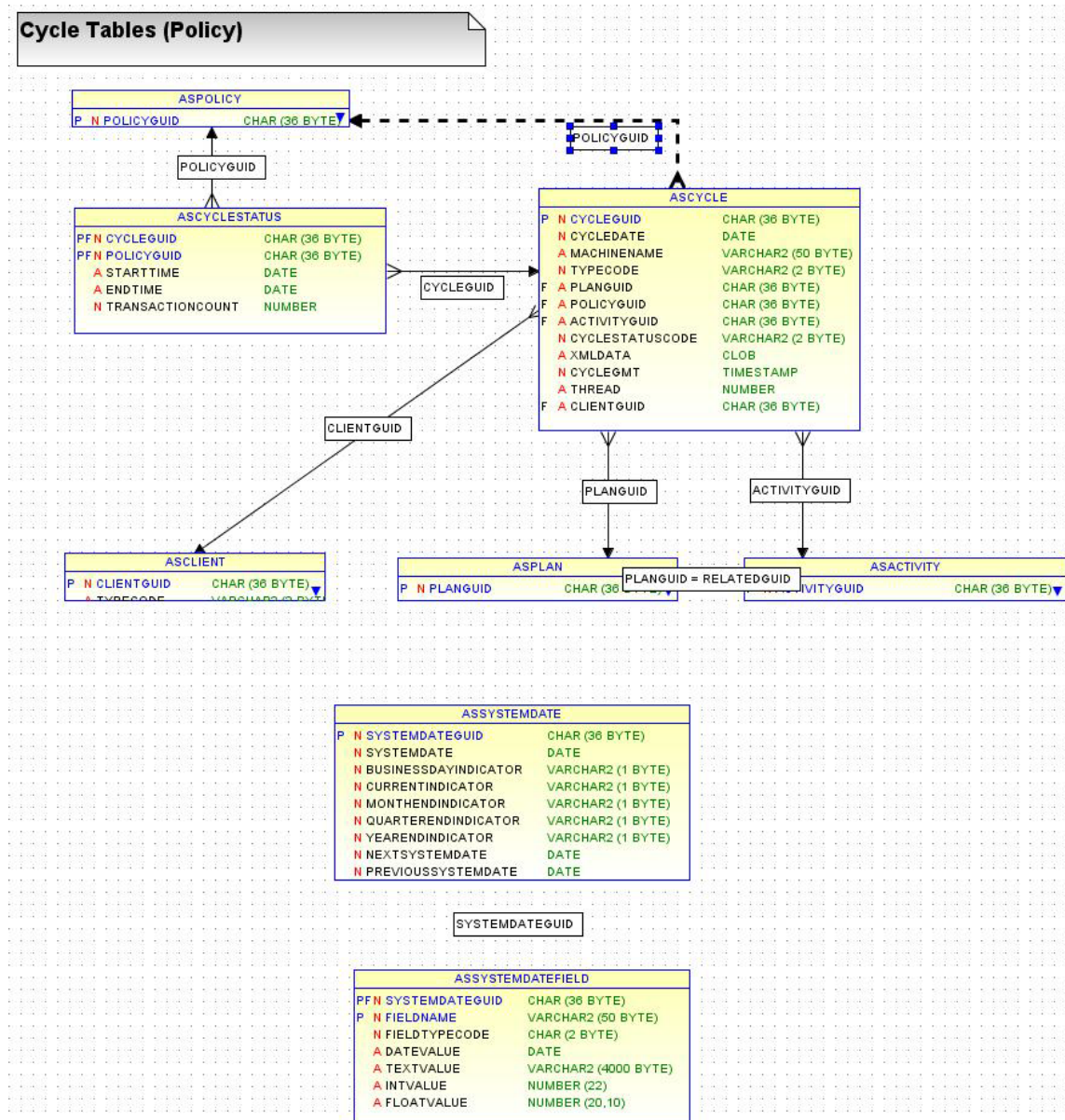
Cycle Commands

- **P Paused** – Cycle will not process, although it will still check to see if activities should be processed, but sees that the status is still in pause and returns to sleep status.
- **R Resume** – Takes cycle out of pause status.

- **D Advance System Date** - Advances date in the AsSystemDate table.
Should be run after command(s) 01 thru 08 are run.
- **A Abort Current Cycle** – This should only be used when there is a serious error. This is a non recoverable scenario and a shut down will occur. This will mark all records in the database as 99.
- **Q Exit** – Exits cycle.
- **01 Pre-Company** - Runs only the pre-company portion of cycle.
 - i. Cycle will attempt to process all company level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order of one-thousand or less.
 - ii. A non-automatically overridden business error or a system error will cause cycle to abort further processing at this level and move onto the next task.
- **02 Pre-Plan** - Runs only the pre-plan portion of cycle.
 - i. Cycle will attempt to process all plan level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order of one-thousand or less.
 - ii. Plans are executed in parallel using JMS threading.
 - iii. A non-automatically overridden business error or a system error will cause cycle to move onto the next plan in the queue or the next processing level task if any.
- **03 Policy** - Runs the policy level portion of cycle processing.
 - i. All policy level activities with an effective date less than or equal to the current system date will be processed in the following order: effective date (ascending), processing order (ascending), entry gmt (ascending).
 - ii. Policies are processed in parallel using JMS threading.
 - iii. A non-automatically overridden business error or a system error will cause cycle to abort further processing on the policy and move onto the next policy or processing task (if any).
- **04 Post-Plan** - Runs only the post-plan portion of cycle
 - i. Cycle will attempt to process all plan level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order greater than one-thousand.
 - ii. Plans are executed in parallel using JMS threading.
 - iii. A non-automatically overridden business error or a system error will cause cycle to move onto the next plan in the queue or the next processing level task if any.

- **05 Post-Company** - Runs only the post-company portion of cycle.
 - i. Cycle will attempt to process all company level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order **greater** than one-thousand.
 - ii. A non-automatically overridden business error or a system error will cause cycle to abort further processing at this level and move onto the next task.
- **06 Plan Status** – Only used for V7 cycle.
- **07 Pre-Client** - Runs only the pre-client portion of cycle.
 - i. Cycle will attempt to process all client level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order of one-thousand or less.
 - ii. Clients are executed in parallel using JMS threading.
 - iii. A non-automatically overridden business error or a system error will cause cycle to move onto the next client in the queue or the next processing level task if any.
- **08 Post-Client** - Runs only the post-client portion of cycle.
 - i. Cycle will attempt to process all client level activities with an effective date less than or equal to the current system date with one of the pending statuses and a processing order greater than one-thousand.
 - ii. Clients are executed in parallel using JMS threading.
 - iii. A non-automatically overridden business error or a system error will cause cycle to move onto the next client in the queue or the next processing level task if any.

Cycle Tables



Cycle Tables