

---

---

# Unicenter

## Working with NetMaster Reporter

Version 2.0



**Computer Associates**  
The Software That Manages eBusiness



This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

© 2002 Computer Associates International, Inc. (CA)

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.



# Contents

---

## Chapter 1: Web-based Reporting

The Purpose of this Guide .....	1-1
Reports .....	1-2
Report Scheduling .....	1-2
Report On Demand .....	1-2
Report Accessibility .....	1-2
Data Consolidation from Multiple Systems .....	1-3
Mainframe Installation and Web Hosting .....	1-3
For More Information .....	1-3

## Chapter 2: Concepts

Data Regions .....	2-2
Report Center .....	2-2
NetMaster Java Framework .....	2-3
Aggregation Service .....	2-3
Control Service .....	2-3
Database Expiry Service .....	2-3
Data Warehouse Service .....	2-4
Prompt Generator Service .....	2-4
Report Expiry Service .....	2-4
Report Generator Service .....	2-4
Scheduler Service .....	2-4
Timeframe Service .....	2-5
Directory Structure .....	2-5
Distributed Code .....	2-5
Working Data .....	2-6
Log .....	2-6
Report Output .....	2-7

---

## Chapter 3: Implementation

Checklist .....	3-2
Implementing Java 2 Technology Edition .....	3-3
Interfacing with the DB2 Database .....	3-3
JDBC and SQLJ .....	3-4
Location Name .....	3-4
REPORTER Parameter Group .....	3-4
Implementing UNIX System Services Authorization .....	3-5
Authorizing the NetMaster Java Framework and the Report Center Region .....	3-5
Authorizing a User .....	3-7
Installing the Software .....	3-7
Creating the Reporter Databases .....	3-8
Configuring Your NetMaster Regions .....	3-8
Implementing the Report Center .....	3-9
Implementing the Web Interface Port .....	3-9
Verifying the Implementation .....	3-10
Implementing Data Collection .....	3-10
Managing the NetMaster Java Framework Started Task .....	3-11

## Chapter 4: Displaying Reports

Accessing Reporting Functions .....	4-1
Displaying a Sample Report .....	4-2
Setting a Report Up for Quick Retrieval .....	4-3

## Chapter 5: Running Reports On Demand

Generating a Sample Report .....	5-2
Displaying the Generated Report .....	5-3

## Chapter 6: Scheduling Report Runs

Schedule Sample Reports .....	6-2
Displaying the Generated Reports .....	6-2
Maintaining Reporting Schedules .....	6-3

---

## Chapter 7: Troubleshooting

Cursor Is Missing from Comment Text Box .....	7-1
Database Connection Cannot Be Established .....	7-1
Database Connection Cannot Be Established – No Suitable SQLJ/JDBC Driver .....	7-2
Database Information Cannot Be Retrieved .....	7-2
Key Range Does Not Exist .....	7-3
Logon Is Forced on Close of Child Window .....	7-4
NetMaster Java Framework Cannot Be Started .....	7-4

## Appendix A: Implementing UNIX System Services Authorization

Task 1: Assigning an OMVS UID .....	A-1
Task 2: Defining the OMVS Segment .....	A-1
CA-ACF2 Security Subsystem .....	A-1
RACF Security Subsystem .....	A-2
CA-Top Secret Security Subsystem .....	A-2
Task 3: Creating the Home Directory .....	A-2

## Appendix B: Database Structure

Data Model .....	B-1
Resource Table .....	B-2
Cisco Channel Card Interface .....	B-3
Cisco CLAW .....	B-4
Cisco TN3270 Server .....	B-4
Communications Storage Manager .....	B-5
Enterprise Extender .....	B-5
File Transfer Event .....	B-6
Host Channel .....	B-6
IP Application Address Space .....	B-7
IP Node .....	B-8
Open Systems Adapter Support Facility .....	B-9
SNA Node .....	B-10
TCP/IP Stack .....	B-11
TCP/IP Stack Local Interface .....	B-12

---

## Appendix C: Data Retention Rules

Syntax	C-2
Example 1	C-3
Example 2	C-3
Example 3	C-3
Example 4	C-3

## Appendix D: Report Definition Structure

Top Level	D-1
Report Heading	D-2
Report Sections	D-2
Section Headings	D-2
Section Content	D-3
Charts	D-4
Pie Charts	D-4
Bar Charts	D-5
Line Charts	D-5
Combined Bar and Line Chart	D-5
X-axis Labels	D-5
Queries	D-6
Crosstabs	D-6
Parameters	D-7
Time Frames	D-8
Document Type Definition	D-9

## Appendix E: Migrating IPTREND Data to the Database

Determining Whether the IPTREND Data Should Be Migrated	E-2
Conversion Issues	E-3
Preparing Your VFS File	E-3
Migrating Your IPTREND Data Set	E-4
Disabling the IPTREND Data Set	E-4
Frequently Asked Questions	E-5

---

## Appendix F: DB2 Administration

Improving the Response Time for the DB2 Queries .....	F-1
Reserving Free Space in Table Spaces and Indexes .....	F-2
Analyzing Buffer Pools to Support an Analytical Workload .....	F-2
Backing Up and Recovering .....	F-2

## Index



# Web-based Reporting

---

Unicenter NetMaster Reporter brings you web-based historical and trend reporting of collected data from file transfer and TCP/IP activities. It combines the familiarity and stability of the enterprise environment with the usability of the web, providing integrated mainframe-to-browser presentation of your network performance data.

## The Purpose of this Guide

This guide introduces Unicenter NetMaster Reporter to you in an efficient and visual manner. By the time you have finished reading this guide, you will have an overview of the wide scope of the product and its usability will be familiar to you. It is important to us that you feel comfortable with Unicenter NetMaster Reporter before you begin to use it.

## Reports

Unicenter NetMaster Reporter comes with a variety of predefined reports. All reports include graphical charts. A database of sample data is also distributed.

The reports are produced from the following data:

- Enterprise TCP/IP stack and device performance (You collect this data by using the Unicenter NetMaster Network Management for TCP/IP or the Unicenter NetMaster Operations for TCP/IP products.)
- File transfer traffic broken down by address, product, and user (You collect this data by using the Unicenter NetMaster File Transfer Management product.)

If you have collected real data, then by running the predefined reports, you can display and analyze the different facets of the monitored environment immediately from a web browser.

You can run reports over a selection of time frames. Supported time frames include today, yesterday, last seven days, last month, last quarter, and many more.

## Report Scheduling

Unicenter NetMaster Reporter provides an integrated and flexible report scheduler service that allows automatic running of reports at daily, weekly, monthly, or quarterly intervals. You can schedule the same report to run over different time frames at different intervals.

## Report On Demand

You can request a report directly at any time.

## Report Accessibility

Unicenter NetMaster Reporter stores generated reports in the hierarchical file system (HFS) of UNIX System Services (or UNIX for short). You can use a web browser, or any z/OS or OS/390 hosted web application server to view those reports directly.

## Data Consolidation from Multiple Systems

You only need to implement Unicenter NetMaster Reporter Report Center on a single system. Collected data can be fed to it from products on other systems. For information about the Report Center, see the chapter “Concepts.”

## Mainframe Installation and Web Hosting

All Unicenter NetMaster Reporter components are mainframe based. Full hypertext transfer protocol (http) access from a web browser is provided by the NetMaster web interface. No PC server installation is required.

## For More Information

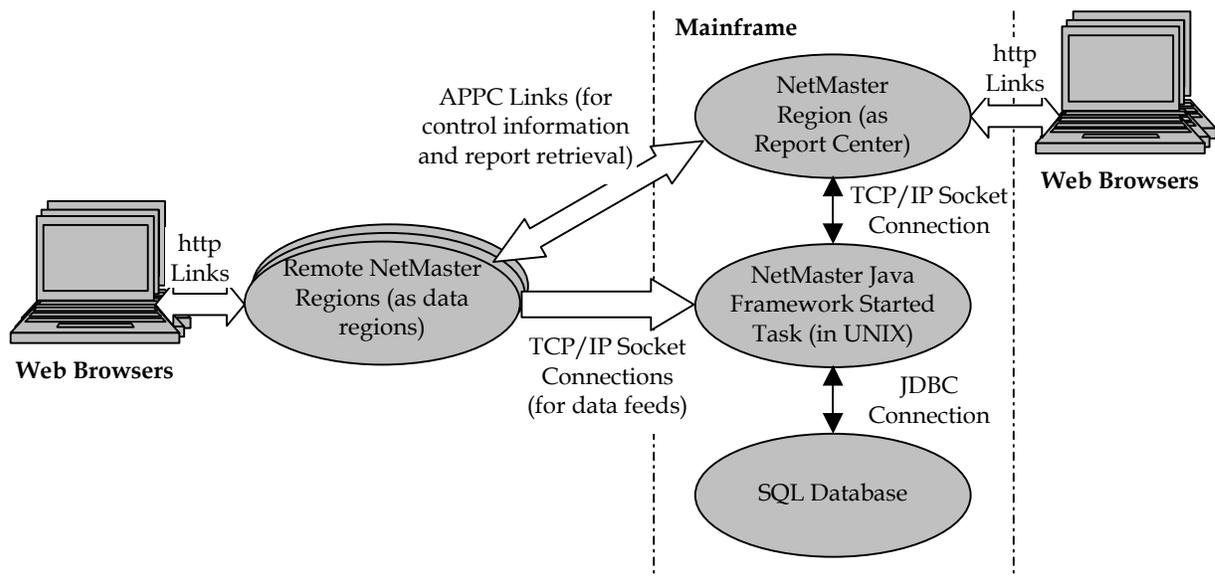
After reading this *Working with NetMaster Reporter* guide, you can refer to the following manuals for additional information:

- *Unicenter Mainframe Network Management Installation and Setup Instructions* that tells you how to install and set up Unicenter NetMaster Reporter
- Online help that provides information about the user interface

In addition, [esupport.ca.com](http://esupport.ca.com) offers procedural information and answers to frequently asked questions.



The following diagram shows a typical configuration.



The implementation requires the following software on the same system:

- A structured query language (SQL) database that receives data from the data regions through the Data Warehouse Service of the NetMaster Java Framework by using the Java database connectivity (JDBC) connection—You use Unicenter NetMaster Reporter to report on this data.
- The NetMaster Java Framework that provides the services to manage and generate reports—It runs as a started task under the UNIX System Services Java Virtual Machine (JVM) environment.
- A NetMaster region that acts as a Report Center—The Report Center interfaces with the UNIX System Services and the NetMaster Java Framework. The region can also simultaneously act as a data region.

You only need to implement the Report Center on one system. It can receive data from multiple data regions locally and remotely.

Access to reports is through a web browser, which communicates with the Report Center through the NetMaster web interface. You can also access reports from a data region through the APPC link.

## Data Regions

A data region is a NetMaster region that collects and sends data to the Data Warehouse Service of the Java Framework. How and what data is collected depend on the products running in the region.

Data collection can be local in the same region as the Report Center or remote in a different region. If the data region is remote, it uses an APPC link to retrieve information from the Report Center. The information enables the data region to create a REPORTER\$DWS data warehouse server and to establish a TCP/IP socket connection for sending data to the Data Warehouse Service of the Java Framework.

The APPC link also enables you to request, schedule, and retrieve reports by using a web browser from a data region.

For information about how to link a data region to a Report Center, see *Configuring Your NetMaster Regions* in the chapter “Implementation.”

## Report Center

The Report Center is the NetMaster region that is designated to perform all UNIX System Services-based operations on behalf of the data regions linked to it. These operations include issuing commands through the TCP/IP socket connections and accessing HFS files. This Report Center region must be on the same system as the database.

## NetMaster Java Framework

The Java Framework is the Java code that provides the framework used by Reporter. The Java Framework runs under the UNIX System Services Java Virtual Machine (JVM) environment. It provides a number of services. Some services are persistent (that is, they are always available). Other services are timer-based (that is, they do not start until triggered by a timer). A single daily timer specified by the Report Center controls these timer-based services. The following sections describe all these services.

### Aggregation Service

This service is started once a day by the daily timer. It updates the database and aggregates hourly fact records (data) into daily fact records, and daily fact records into monthly fact records.

### Control Service

This service is always available. It provides a generalized command and control interface for the Report Center to access the Java Framework services. It also processes the status and information commands for the Java Framework infrastructure itself. It is used to pass on demand requests to the Report Generator Service.

### Database Expiry Service

This service is started once a day by the daily timer. It updates the database and deletes expired fact table records. (For information about the database tables, see the appendix “Database Structure.”)

Fact records expire according to their retention period and resource class. The data retention period varies with the type of data. For example, hourly data is kept for a few days, monthly data may be kept for many years. To find out what data is in the database, use the DataWarehouseStatus report.

The default data retention rules are in the Adaptor.ini file. For the location of the file, see [Distributed Code](#) in this chapter. For information about the syntax of a retention rule, see the appendix “Data Retention Rules.”

## Data Warehouse Service

This service is always available. It maintains socket connections with data regions to receive performance data as extensible markup language (XML) documents, parses the documents, transforms the data into data warehouse fact rows, and adds them to a structured query language (SQL) database using Java database connectivity (JDBC).

## Prompt Generator Service

This service is started once a day by the daily timer. It reads the database and updates the hierarchical file system (HFS) to build lists of values for report criteria prompts. These values appear in the drop-down list boxes on the web pages. The lists are customized for the specific attributes and resource class that are included in the selected report.

## Report Expiry Service

This service is started once a day by the daily timer. It updates the HFS to delete any expired report output. The entire directory for an expired run is deleted. Report output expires after 48 hours for on demand reports and after the number of days chosen by the user for scheduled reports.

## Report Generator Service

This service is always available. It takes report run requests from a queue and processes them serially. For each request, it reads the report definition file from the HFS and uses that information to query the database for each chart or table in the report. It generates .PNG graphics files for the charts and the hypertext markup language (HTML) for the final report. It writes the HTML and other files to the HFS.

## Scheduler Service

This service is started once a day by the daily timer. It reads the schedule definitions from the HFS and assesses whether the schedules are due today. If the schedules are due, it adds them to the report queue for processing by the Report Generator Service.

## Timeframe Service

This service is started once a day by a special daily timer specified in the Adaptor.ini file. It is set at one minute past midnight. It updates the Timeframe table in the database to map absolute day values to relative time frames.

## Directory Structure

Reporter uses the HFS to store files that include the following:

- Distributed, read-only code
- Working data such as report schedule definitions
- Log file
- Report output

The files are stored in directories with prefixes that are set by the Report Center. For information about how to specify the prefixes, see Implementing the Report Center in the chapter “Implementation.”

## Distributed Code

The distributed code is in the following files:

Files	Description
.../reporter20/code/*.*	Code
.../reporter20/adaptor/Adaptor.*	Control files
.../reporter20/prompts/Prompts.ini	Prompt Generator Service control file
.../reporter20/prompts/sample/*.txt	Sample prompt values files
.../reporter20/reports/common/*.*	Attribute description, HTML skeleton, logo, and style sheet files
.../reporter20/reports/definitions/Reporter.dtd	Document type definition file for report definitions
.../reporter20/reports/definitions/report-name.xml	Report definitions

## Working Data

The working data is in the following files:

Files	Description
.../reporter20/prompts/*.txt	Prompt values files
.../reporter20/schedules/report-name/file-subdirectory-prefix/report.pty	Log for a schedule
.../reporter20/schedules/report-name/file-subdirectory-prefix/report.sched	Details of a schedule

where *file-subdirectory-prefix* is the ID of the user that scheduled the report concatenated with an internally generated identifier.

## Log

The log is in the following file:

.../reporter20/log/Adaptornnn.log

where *nnn* is a three-digit number (for example, 001).

The number of log files and the size of each file are specified in the Adaptor.ini file. The files wrap in a cyclic fashion.

## Report Output

The report output is in the following files:

Files	Description
.../reporter20/reports/run/report-name/file-subdirectory/chartn.png	Charts for real data
.../reporter20/reports/run/report-name/file-subdirectory/report.html	Generated report page for real data
.../reporter20/reports/run/report-name/file-subdirectory/report.pty	Details of a report run on real data
.../reporter20/reports/samplerun/report-name/file-subdirectory/chartn.png	Charts for sample data
.../reporter20/reports/samplerun/report-name/file-subdirectory/report.html	Generated report page for sample data
.../reporter20/reports/samplerun/report-name/file-subdirectory/report.pty	Details of a report run on sample data

*file-subdirectory* is *file-subdirectory-prefix\_yyyymmddhhmmss*, where:

<i>file-subdirectory-prefix</i>	Is the ID of the user that scheduled or ran the report concatenated with an internally generated identifier.
<i>yyymmdd</i>	Is the date at which the report is run.
<i>hhmmss</i>	Is the time at which the report is run.



# Implementation

---

Unicenter NetMaster Reporter is designed for installation on the mainframe but accessed by using a web browser. It uses the hierarchical file system (HFS) of UNIX System Services to store the generated reports.

Implementation requires knowledge of the Java 2 Technology Edition, the DB2 database for OS/390 or z/OS, and UNIX System Services security.

## Checklist

This checklist helps you to keep track of the tasks for getting Unicenter NetMaster Reporter to work:

✓	Task	Note
For IBM Java Developer Kit, Java 2 Technology Edition:		
	<a href="#">Implementing Java 2 Technology Edition</a>	This task can be performed anytime before Task 6.
For DB2 Universal Database for OS/390 and z/OS:		
	<a href="#">Interfacing with the DB2 Database</a>	This task can be performed anytime before Task 4. See your DB2 administrator.
For UNIX System Services:		
	<a href="#">Implementing UNIX System Services Authorization</a>	This task can be performed anytime before Task 6. The task requires, at the minimum, knowledge of the HFS and the name of the NetMaster Java Framework started task (which is also the user ID of the Report Center region) that are to be specified in Task 1. See your security administrator.
For Unicenter NetMaster Reporter:		
1	<a href="#">Installing the Software</a>	This task can be performed in parallel with the previous tasks.
2	<a href="#">Creating the Reporter Databases</a>	This task follows Task 1.
3	<a href="#">Configuring Your NetMaster Regions</a>	This task follows Task 1.
4	<a href="#">Implementing the Report Center</a>	This task follows Task 3.
5	<a href="#">Implementing the Web Interface Port</a>	This task follows Task 4.
6	<a href="#">Verifying the Implementation</a>	This task follows Task 5
7	<a href="#">Implementing Data Collection</a>	This task follows Task 6.

## Implementing Java 2 Technology Edition

You must ensure that the Java 2 Technology Edition product at the SDK 1.3.1 level or later is installed on the system. You can download it from the [Getting and installing the IBM Developer Kit for OS/390, Java™ 2 Technology Edition](#) web site.

To check the version, enter the following command in OMVS from the .../bin directory of the Java 2 Technology Edition product:

```
java -version
```

## Interfacing with the DB2 Database

The Unicenter NetMaster Reporter interfaces with the DB2 database for OS/390 and z/OS by using Java database connectivity (JDBC). The following sections describe the things you must do.

**Important!** *The DB2 subsystem must run in the sysplex mode, not in the local mode. NetMaster Reporter requires the resource recovery services (RRS), which must be used in the sysplex mode.*

**Important!** *If you are using DB2 7.0, ensure that you use the SQLJ/JDBC driver with JDBC 1.2 support and that you have installed the DB2 Utilities Suite. JDBC 2.0 is not supported currently.*

## JDBC and SQLJ

Ensure that the JDBC and SQLJ are installed and configured according to the DB2 *Application Programming Guide and Reference for Java*. An outline of the steps follows:

1. Ensure that the DB2 database is at the maintenance level indicated in the *Unicenter Mainframe Network Management Installation and Setup Instructions*.
2. Set UNIX System Services environmental variables. Note the CLASSPATH setting for the db2j2classes.zip file.
3. Ensure that in the runtime properties file (db2sqljjdbc.properties), the following parameters are set as indicated:

```
DB2SQLJSSID=db2-subsystem-name
DB2SQLJMULTICONTEXT=YES
DB2SQLATTACHTYPE=RRSAF
DB2DISABLETAF=1
DB2SQLJDBRMLIB=dbrm-data-set
```

4. In OMVS, execute the db2genJDBC utility.

The utility creates four database request modules (DBRMs) and a JDBC profile.

5. Use sample job SDSNSAMP(DSNTJJCL) to bind the DBRMs.

## Location Name

Ensure that the DB2 location name is in the SYSIBM.LOCATIONS table. Note its value.

## REPORTER Parameter Group

Ensure that the REPORTER parameter group in the NetMaster region that acts as the Report Center is customized for the database (see [Implementing the Report Center](#) in this chapter).

## Implementing UNIX System Services Authorization

You must implement UNIX System Services authorization for NetMaster Java Framework and the NetMaster region that acts as the Report Center. The authorization controls access to the files in the HFS. The Java Framework and the Report Center region must have the appropriate access to the following directories:

- .../reporter20 that contains the Reporter files – read, write, and execute
- Directory that contains the shell scripts that invoke JDBC and SQLJ (for example, /usr/lpp/db2/db2610) – read and execute
- .../bin directory of IBM Developer Kit for OS/390, Java 2 Technology Edition – read and execute

By default, logon users assume the UNIX System Services authorization of the Report Center region. If required, you can implement specific authorization for individual users. However, if you do this, you must do this for all users (although the authorization can be different for different users).

### Authorizing the NetMaster Java Framework and the Report Center Region

Authorize the NetMaster Java Framework or the Report Center region as follows:

1. Determine the user ID of the region, which should be the same as the name of the NetMaster Java Framework started task.
2. Define an OMVS segment for the user ID.
3. Define the segment to your security subsystem.
4. Implement access to HFS directories.

For more information, see the appendix “Implementing UNIX System Services Authorization.”

## Authorizing the NetMaster Java Framework to Use RRSAF

To use the Resource Recovery Attachment Facility (RRSAF), you must supply the following definitions to your security product for the owner of the Java Framework started task.

You must enter these definitions at the TSO READY prompt.

**Note:** In all the following examples:

*ssnm* is the DB2 subsystem name

*userid* is the user ID of the owner of the Java Framework started task

RACF

```
REDEFINE DSNR (ssnm.RRSAF) UACC (NONE)

then
PERMIT ssnm.RRSAF CLASS (DSNR) ID (userid) ACCESS (READ)
```

ACF2

```
ACF2
SET RESOURCE (SAF)
COMPILE
$KEY (ssnm.RRSAF) TYPE (SAF)
UID (userid) ALLOW

STORE
END
```

Note the blank lines used to terminate the COMPILE of the rule.

Refresh with the operator command:

```
F ACF2, REBUILD (SAF)
```

TOP SECRET

```
TSS ADD (ALL) DB2 (DSNR.ssnm.RRSAF)
TSS PER (userid) DB2 (DSNR.ssnm.RRSAF)
```

## Authorizing a User

If you do not want your users to assume the authorization of the Report Center region, proceed as follows for each user. You must assign a UID of 0 to the region. If BPX.DAEMON is defined, the region must also have read access to the BPX.DAEMON.HFSCCTL facility.

**Important!** *The HFS file permission settings control functions such as who may schedule, view, and delete reports. Before implementing user-level security, you must have complete understanding of the Reporter HFS file hierarchy and of how to implement UNIX file permissions. (For information about the file hierarchy, see Directory Structure in the chapter "Concepts.")*

1. Define an OMVS segment to suit the authorization required for the user ID.
2. Define the segment to your security subsystem.
3. Implement access to HFS files.
4. Enable HFS user-level security through the HFSSECURITY parameter group. (To access the list of parameter groups, log on to the NetMaster region and enter the /ICS shortcut.)

## Installing the Software

The Unicenter NetMaster Reporter software must be installed before you perform the following tasks.

For information about the prerequisites and how to install and set up the software, see the *Unicenter Mainframe Network Management Installation and Setup Instructions*.

## Migrating from Version 1.0

If you have been using NetMaster Reporter 1.0 and want to migrate data stored in the Ingres database to the DB2 database, contact Technical Support.

**Important!** *If you want to migrate your NetMaster Reporter 1.0 data, you must complete the migration before performing the following tasks.*

## Creating the Reporter Databases

The installation process provides you with the following:

- The *dsnpref*.WR200.WRSAMP(CRExxxxn) JCL job to create the database and load any data, where:

*dsnpref* Is the data set prefix used during installation.

*xxxx* Is PROD for production data and SAMP for sample data.

*n* Is the DB2 version number.

- Eight data sets for the sample data:

SAMPATTR	SAMPENUM	SAMPKEYR	SAMPNUMF
SAMPPAGT	SAMPPERD	SAMPRESC	SAMPTIME

To create and implement the databases, proceed as follows:

1. Use the CRESAMP*n* JCL job to create your sample database.
2. Use the CREPROD*n* JCL job to create your production database. (The job also primes the Keyrange table by using the *dsnpref*.WR200.SAMPKEYR data set.)
3. After you create the sample and production databases, grant access to them by using the statements in the supplied *dsnpref*.WR200.WRSAMP(SYSGRANT) member.

## Configuring Your NetMaster Regions

NetMaster Reporter comprises two parts: a Report Center, and one or more data regions. The Report Center must be on the same system as the Java Framework and the database.

The simplest configuration is a single NetMaster region that contains both the Report Center and the data region.

If you have linked your regions in a multisystem environment, you must implement the Report Center in a focal point region. If none of the focal point regions are on the same system as the database, you must set up a new region on the same system as the database, make this region the Report Center, and link it into the multisystem environment as a focal point region. (For more information about multisystem operation, see the *Automation Services Administration Guide*.)

## Implementing the Report Center

You must implement a NetMaster region as the Report Center. The region must be on the same system as your database. Use the REPORTER parameter group in the NetMaster region to implement the connectivity with the NetMaster Java Framework and the DB2 database, and the HFS directories for various product files.

Customize the REPORTER parameter group as follows:

1. In the NetMaster region, enter **/ICS** to list the parameter groups.
2. Enter **U** beside the REPORTER parameter group (\$WR REPORTER) to customize it as follows:
  - a. In the Data Warehouse Server IP Address and Port Number fields, specify the address of the system on which the NetMaster Java Framework resides, and assign a port number. Then press Enter.
  - b. Validate the displayed directory prefixes for the different types of files. Then press F8.
  - c. In the Type of Data Warehouse field, enter the path and location as implemented in [Interfacing with the DB2 Database](#) in this chapter.

For more information about the parameter group, press F1 for help.

3. When you have finished customizing the parameter group, press F6 to set the parameters immediately and then press F3 to save your changes.

## Implementing the Web Interface Port

The Unicenter NetMaster Reporter software can be accessed through the Report Center region and the data regions by using web browsers. You must implement a port in each region that you want to use as an access point. Review the SOCKETS parameter group (\$NM SOCKETS) in a region to ensure that the port is implemented. The Web Access URL field displays the uniform resource locator (URL) for web access. Communicate this URL to the users.

## Verifying the Implementation

Unicenter NetMaster Reporter provides a test facility that you can use to verify your implementation. You perform the test from a web browser. You must start the NetMaster Java Framework before you perform the test.

Verify your implementation as follows:

1. Use the web access URL to log on to the NetMaster region.
2. Start the NetMaster Java Framework as follows:
  - a. Choose Utilities, Java Framework Status to display the Java Framework Status page.
  - b. Click  More Actions, and choose Start Java Framework. Confirm the action to start the NetMaster Java Framework.
3. After the NetMaster Java Framework has started, choose Reporting, Report Activity Status to display the Report Activity Status page.
4. Click the Run Function Test tab, choose a function test, and review the results for any errors. For troubleshooting information, see the chapter "Troubleshooting."

## Implementing Data Collection

To provide Unicenter NetMaster Reporter with real data, you must implement the data regions (for example, Unicenter NetMaster Network Management for TCP/IP or Unicenter NetMaster File Transfer Management). For information about how to implement these regions, see the respective product documentation.

Data collection may be in the same region as the Report Center, or it may be in a different (remote) region. For the remotely collected data to be fed to the database, you must link the remote data region to the Report Center region. To establish the link, proceed as follows in the remote region:

**WARNING!** When you use Step 1 to establish the link, you also replace the knowledge base in the remote region by that in the Report Center region. (For more information, see the Automation Services Administration Guide.)

1. Enter the `/MADMIN.SD` panel path to establish the link.
2. In the REPORTER parameter group, set the Report Center Link Name field to the ACB name of the Report Center region.

## Managing the NetMaster Java Framework Started Task

The NetMaster Java Framework started task was installed and set up as described in [Installing the Software](#) in this chapter. The task was started when you test your implementation as described in [Verifying the Implementation](#) in this chapter. The task establishes JDBC connections with the database, and TCP/IP socket connections with the Report Center and data regions. The task spawns several subtasks.

**Once started, you should leave the NetMaster Java Framework task running at all times.** Leave it running even when you shut down all the data regions and the Report Center region. However, if work is being done on the database that requires the connection between the database and the Java Framework to be removed, then you should stop the Java Framework started task.

Because the Java Framework started task is always running, it will not pick up any changes you make to the Adaptor.ini initialization file through the REPORTER parameter group. You must stop and restart the task to pick up the changes.

Consider the following when you stop the Java Framework started task:

- Queued reports will not be completed and will have a status of stopped.
- The stopping process will wait for the completion of all current tasks and might take a while.

You can start and stop the Java Framework started task from a web browser. Log on to a Report Center or data region, and choose Utilities, Java Framework Status, More Actions.

**Note:** If you bring down the DB2 subsystem, then you must recycle the Java Framework so that new threads can be obtained.



# Displaying Reports

---

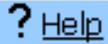
Unicenter NetMaster Reporter provides sample reports that you can display immediately. It also provides sample data that enables you to generate your own sample reports. This chapter describes how to display reports. Other chapters describe how to generate reports.

## Accessing Reporting Functions

To access the reporting functions, you must log on to the NetMaster region by using its web access URL. (For information about the URL, see *Implementing the Web Interface Port* in the chapter “Implementation.”)

## Displaying a Sample Report

The Reporting menu page enables you to report on the sample data and your own data. The procedure to display the reports is similar. In this section, you will display the distributed sample reports to see what they look like.

**Tip:** For help on a Reporter web page, click  [Help](#) to open a new window that contains further information about the page.

Display a report from your web browser as follows:

1. Choose Reporting, Sample Reports to display the Sample Reports page.



Choose a report name and then click an action.

**Information**

Use this page to:

- View the output of completed report runs.
- Run reports, schedule reports, and administer the report schedules and output.

**List Report Runs** shows the results of all runs of your chosen report. From this list, you can examine the run details, delete the report output, or click on the links to view the report output.

**Run Report On Demand** lets you pick the custom criteria you want to apply to your chosen report, and add it to the report queue to be run as soon as possible. You will be returned a link to the page that will display your final report output.

**List Schedules** shows the schedules for your chosen report. From this list, you can examine the criteria, frequency and retention details for each schedule. You can also delete schedules.

**Add Schedules** lets you add schedules for your chosen report.

**List runs for all report names** shows runs results for all report names, not just a single report name. From this window, you can click on the links to view the report output.

These tasks access data from the distributed sample database. To perform these same tasks against your live database, use the NetMaster Reports option.

2. Click  to list the reports.
3. To display a report, click the link in the Data and Time of Run column. Review the sample reports to see how you can apply them to your environment.

## Setting a Report Up for Quick Retrieval

When you see a report that you want to retrieve again, you can add its link as a bookmark in your web browser (for example, as a favorite web page in Internet Explorer). After you have added the link, you can retrieve the linked report without having to log on to the NetMaster region.

To add the link, right-click on the report page, and choose the action that adds the link.

However, when you generate reports by using real data, note the following:

- Reports generated on demand are deleted after 48 hours.
- Scheduled reports are deleted according to the schedule definition.

**Tip:** You can add a special URL that enables you to retrieve the latest run of a particular report. For example, to retrieve the latest ApplicationOverview report on your real data, add the following URL: `http://region-url/public/reports/latest.esp?reportname=ApplicationOverview`. For example, to add the URL to Internet Explorer, create a dummy favorite. You can then rename it and change its URL. Choose Rename to change the name of the favorite to that of the report. Choose Properties, and then click the Web Document tab to change the URL.



# Running Reports On Demand

---

You can report on the data in your database at any time you desire. During implementation, you created a sample database and a production database (see *Creating the Reporter Databases* in the chapter “Implementation”). Data for the production database comes from the data regions. For information about how to collect this data, see *Implementing Data Collection* in the chapter “Implementation.”

This chapter guides you through the procedure to run a report on the sample data. The procedure to run a report on your real data is similar.

## Generating a Sample Report

Run a report on the sample data as follows:

1. If you are not logged on to the NetMaster region already, log on to it by using its web access URL.
2. Choose Reporting, Sample Reports to display the Sample Reports page, and click the Run Report On Demand tab.
3. Click the Report Name drop-down list box, and select the report you want to run (for example, ApplicationOverview).

The screenshot shows the 'Sample Reports' interface. At the top, there's a 'Report Name' dropdown menu currently showing 'ApplicationOverview'. To its right is a button that says 'List sample runs for all report names'. Below these are four buttons: 'List Report Runs', 'Run Report on Demand', 'List Schedules', and 'Add Schedule'. The 'Run Report on Demand' button is highlighted. Below this is a section titled 'Criteria - Run Report On Demand'. It contains a 'Timeframe' dropdown menu set to 'LAST 7 DAYS' and a text input field for 'Comment'. At the bottom right of this section is a button with a refresh icon and the text 'OK / Redo'.

Enter the criteria above and click 'OK' to proceed.

4. In the Criteria section, select a time frame on which to report and a comment to help you identify the generated report later on.

**Note:** Depending on the report, you might need to specify other criteria.

5. To run the report, click  OK / Redo.

After some processing, a link is returned in the Results section. Review the information in the Results section to find out how to use the link.

6. Click the link to open a new window that either shows the status of the report run or the generated report. You may add the link as a bookmark in your browser so that you can return to it later.

If you have implemented data collection, run reports on your real data to see how well your network is operating.

## Displaying the Generated Report

You can display the generated report by using one of the following methods:

- You can leave the report window open. When report generation is complete, the window displays the report.
- You can add the link to the report as a bookmark and use it later to display the report.
- You can use the List Report Runs tab to list the submitted runs of the report selected in the Report Name drop-down list box. You can click on it to display it. A completed run has the status of ok.



## Scheduling Report Runs

---

The previous chapter describes how you can run reports on demand. However, it is more likely that users will want reporting at regular intervals. They might want to have their reports ready when they arrive at their office in the morning.

This chapter guides you through the procedure to schedule report runs on the sample data. The procedure to schedule report runs on your real data is similar.

Scheduled report runs are checked and performed (if required) at the housekeeping time specified by the REPORTER parameter group in the NetMaster region that acts as the Report Center.

## Schedule Sample Reports

Schedule report runs on the sample data as follows:

1. If you are not logged on to the NetMaster region already, log on to it by using its web access URL.
2. Choose Reporting, Sample Reports to display the Sample Reports page, and click the Add to Schedule tab.
3. Click the Report Name drop-down list box, and select the report you want to schedule (for example, CsmAnalysis).

Enter the criteria above and click 'OK' to proceed.

4. In the Criteria section, complete the following:
  - Select the communications storage manager (CSM) on which you want the report.
  - Select a time frame on which to report.
  - Provide a comment to help you identify the generated reports later on.
  - Specify the number of days for which you want to retain the generated reports.
  - Select the number of report runs to perform, after which the schedule will be deleted.
  - Select the time interval between each report run.

**Note:** Depending on the report, the criteria that you specify might be different.

5. To add the schedule, click  OK / Redo.

## Displaying the Generated Reports

You can use the List Report Runs tab to list the submitted runs of the report selected in the Report Name drop-down list box. You can click a listed report run to display it.

## Maintaining Reporting Schedules

You can review the schedules of a report and delete the schedules that are not required any more as follows:

1. Click the List Schedule tab to list the schedules for the report selected in the Report Name drop-down list box.
2. Check the schedules that you want to delete.
3. Click  Delete Ticked Schedules.

You will be prompted to confirm the deletion. To delete the checked schedules, confirm the deletion.



This chapter enables you to troubleshoot common problems.

## Cursor Is Missing from Comment Text Box

If while trying to run or schedule a report, you are unable to place the cursor in the Comment text box, click outside the box. You will now be able to place the cursor in the box and start typing.

## Database Connection Cannot Be Established

If the following message appears in the NetMaster Java Framework log:

```
Cannot establish database connection. Message: DB2SQLJConnection error in native method: constructor: RRS "IDENTIFY" failed using DB2 system:D...
```

proceed as follows by the REASON code:

- If the REASON code is 00f30012, ensure that the DB2 subsystem has been started.
- If the REASON code is 00f30049, ensure that the statements in the supplied SAMPLIB(SYSGRANT) member have been run (see Creating the Reporter Databases in the chapter “Implementation”).
- If the REASON code is 00f30091, ensure that RRS/MVS has been started.

## Database Connection Cannot Be Established—No Suitable SQLJ/JDBC Driver

If the following message appears in the NetMaster Java Framework log, ensure that you have implemented the REPORTER parameter group in the Report Center region correctly.

Cannot establish database connection. Message: No suitable driver...

Ensure that page 2 of the REPORTER parameter group has been completed correctly in terms of the HFS path for DB2 JDBC classes and the DB2 subsystem location name, for example:

```
.- REPORTER - Reporter Interface -----  
Report Center Setup Continued:  
  
Data Warehouse Specification  
Type of Data Warehouse ..... DB2  
DB2 Specification  
HFS Path for DB2 JDBC Classes zip file  
/usr/lpp/db2/db2610/classes/db2jdbcclasses.zip  
DB2 JDBC URL jdbc:db2os390: ... S390LOC
```

The information should carry through to the **last five** lines of the Adaptor.ini file as follows:

```
WebReportsDB2JDBC.driver=ibm.sql.DB2Driver  
WebReportsDB2JDBC.url=jdbc:db2os390:S390LOC  
WebReportsDB2JDBC.timeout=30  
WebReportsDB2JDBC.retry=5  
WebReportsDB2JDBC.SQLType=DB2
```

## Database Information Cannot Be Retrieved

If the following message appears in the NetMaster Java Framework log, ensure that you have run the statements in the supplied SAMPLIB(SYSGRANT) member (see Creating the Reporter Databases in the chapter “Implementation”).

```
Cannot get database information: DB2JDBCCursor Received Error in Method  
prepare:SQLCODE==> -551 SQLSTATE ==> 42501 Error Tokens ==> <<DB2 6.1 ANSI SQLJ-  
0/JDBC 1.0>> jsname SELECT SYSIBM.SYSxxxxxx  
com.ca.syd.utils.datawarehouse.ExpiryService.009
```

## Key Range Does Not Exist

If the following message appears in the Java Framework log:

Key range for table *database-table-name* does not exist

check the following:

- In the `db2sqljdbc.properties` file, check that:
  - Your DB2 subsystem name is specified correctly
  - The `DISABLETAF=1` parameter has been specified

If a problem is found, correct the file, and then rerun the `db2genJDBC` utility as well as the `DSNTJJCL` job to recreate and rebind the DBRMs (see JDBC and SQLJ in the chapter “Implementation”).

- Check the output of the `CREPROD $n$`  job to ensure that **no** errors were encountered, and that all table spaces, tables and indexes were created correctly.

For information about the `CREPROD $n$`  job, see *Creating the Reporter Databases* in the chapter “Implementation.”

- Ensure that the name of the Java Framework started task, the user ID of the started task, the name of the database schema, and the value of the `JFI` parameter in the `dsnpref.rname.PARMLIB(WRPARMS)` member are all the **same**.

*dsnpref* is the data set prefix used during installation and setup; *rname* is the name of the Report Center region. For more information about installation and setup, see the *Unicenter Mainframe Network Management Installation and Setup Instructions*.

- The DB2 subsystem that contains your NetMaster Reporter databases has been brought down as you have been collecting data. Recycle the Java Framework started task so that new threads can be obtained.

## Logon Is Forced when the Child Window is Closed

You logged on to a NetMaster region through your web browser and opened a child window. You closed the child window. When you next try to do something in the main window, you are forced to log on again.

This problem occurs if you are using Internet Explorer 5.5 or 6.0:

- If you are using Internet Explorer 6.0, install the latest service pack. If there is no service pack, order the fix referred to in Microsoft Knowledge Base article Q315713.
- If you are using Internet Explorer 5.5, upgrade to Internet Explorer 6.0, and install the latest service pack or the fix.

## NetMaster Java Framework Cannot Be Started

The Java Framework started task terminates immediately with no obvious errors. The .../reporter20/adaptor/stderr.txt file contains an error similar to the following:

```
.../reporter20/adaptor/Adaptor.bat 1: FSUM7351 not found
```

Enter the following command in OMVS to verify whether Java 2 Technology Edition is installed (see Implementing Java 2 Technology Edition in the chapter “Implementation”):

```
java -version
```

# Implementing UNIX System Services Authorization

Unicenter NetMaster Reporter uses the HFS and requires its users to have UNIX System Services authorization. In general, users require write access to their home directories.

For more information about how to define UNIX users, see the *UNIX System Services Planning* manual.

## Task 1: Assigning an OMVS UID

Choose an OMVS UID number to associate with the user ID. Your security administrator might have a policy for assigning OMVS UID numbers. If not, use a unique number; for example, 123.

For further information about OMVS UID numbers, see your IBM documentation.

## Task 2: Defining the OMVS Segment

The following sections show you how to define the OMVS segment to your security system for a user ID *uuuuuu* and UID number *nnn*.

Choose the `/u/user-name` home directory to associate with the user ID.

### CA-ACF2 Security Subsystem

Enter:

```
SET PROFILE(USER) DIV(OMVS)
INSERT uuuuuu UID(nnn) HOME(/u/user-name) PROGRAM(/bin/sh)
```

After the segment is secured, enter the following to confirm its contents:

```
SET PROFILE(USER) DIV(OMVS)
LIST uuuuuu
```

## RACF Security Subsystem

Enter:

```
ALU uuuuuu OMVS(UID(nnn) HOME(/u/user-name) PROGRAM(/bin/sh))
```

After the segment is secured, enter the following to confirm its contents:

```
LISTUSER uuuuuu OMVS NORACF
```

## CA-Top Secret Security Subsystem

Enter:

```
TSS ADD(uuuuuu) HOME(/u/user-name) OMVSPGM(/bin/sh) UID(nnn) GROUP(OMVSGRP)
```

After the segment is secured, enter the following to confirm its contents:

```
TSS LIS(uuuuuu) DATA(ALL)
```

## Task 3: Creating the Home Directory

Create the `/u/user-name` home directory. Ensure that the UID has the appropriate access to it.

For example, to set up a directory called `/u/user01` for the user ID 123, you can issue the following commands:

```
mkdir /u/user01
chown 123 /u/user01
chmod +777 /u/user01
```

You can then confirm the owner and access to the directory by using the command:

```
ls -l -d /u/user01
```

# Database Structure

---

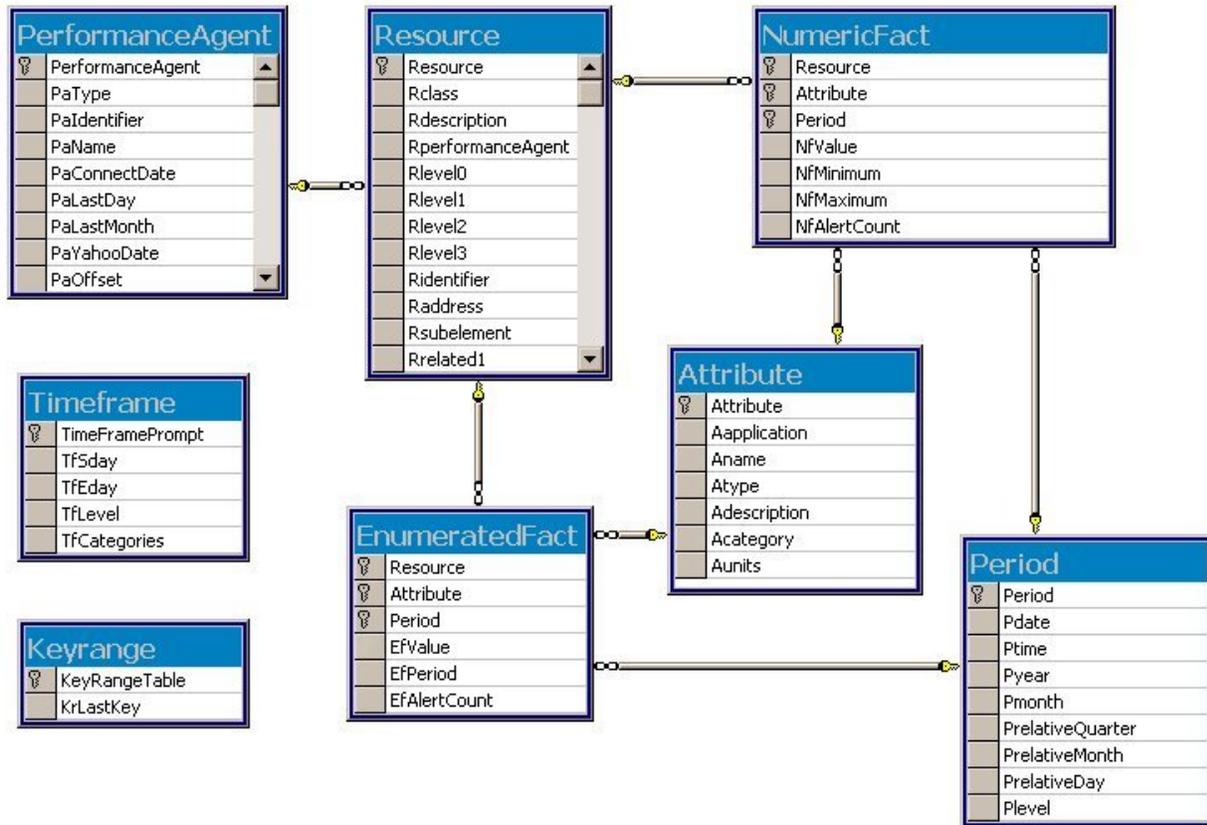
During implementation, you created two databases based on the supplied data model: a production database and a sample database. Unicenter NetMaster Reporter accesses this data in those databases by using a JDBC connection. The implementation of this connection is described in Interfacing with the DB2 Database in the chapter “Implementation.”

## Data Model

The data model uses a classic relational online analytical processing (ROLAP) snowflake schema with non-application specific design with generic facts and a generic resource dimension as the physical implementation. The data model has eight tables as follows:

- PerformanceAgent table stores the details of performance agents sending data to the data warehouse. There is one row per agent.
- Resource table stores the details of an individual monitored resource. The column usage varies with the resource class. There is one row per reported resource.
- NumericFact table stores numeric fact data for counter and gauge attributes.
- Timeframe table is used as an internal work table to present consistent time frame handling across reports and maximize the reusability of time frame filtering on report objects.
- Attribute table contains one row for every reported or monitored attribute.
- Keyrange table is used as an internal work table to prevent the duplicate generation of a dimension table key when more than one NetMaster Java Framework is updating the same dimension table.
- EnumeratedFact table stores character fact data for enumerated attributes such as NETSTATUS and PUSTATUS.
- Period table is used in conjunction with the Timeframe table to provide data retrieval based on the time frame.

The database tables and their relationships are shown in the following diagram. A fact table always joins to the Attribute, Period, and Resource tables, as indicated by the keyed connections. The Resource table further joins to the PerformanceAgent table.



## Resource Table

The use of the columns in the Resource table is dependent on the resource class. Column usage is shown in the following sections.

## Cisco Channel Card Interface

The column usage for a Cisco channel card interface is as follows:

Column	Value
Rclass	IPCIPM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	
Rlevel1	
Rlevel2	
Rlevel3	
Ridentifier	Channel card name
Rsubelement	Null 0 or 1 CMCC channel CLAW total CLAW address TN3270 server IP address Host.ChSysid.CHPID
Raddress	
Rrelated1	
Rrelated2	

## Cisco CLAW

The column usage for a Cisco CLAW is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPCCLAW
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Channel card name
Rlevel1	0 or 1 (channel ID)
Rlevel2	
Rlevel3	
Ridentifier	CLAW IP address
Rsubelement	
Raddress	Path
Rrelated1	OS/390 SYSID
Rrelated2	OS/390 CHPID

## Cisco TN3270 Server

The column usage for a Cisco TN3270 server is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPCPTNS
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Channel card name
Rlevel1	
Rlevel2	
Rlevel3	
Ridentifier	TN3270 card IP address
Rsubelement	
Raddress	
Rrelated1	
Rrelated2	

## Communications Storage Manager

The column usage for a CSM is as follows:

Column	Value
Rclass	IPCSMM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	CSM
Rsubelement	Task name: (ALL), VTAM, stack, or user ID
Raddress	
Rrelated1	
Rrelated2	

## Enterprise Extender

The column usage for an Enterprise Extender is as follows:

Column	Value
Rclass	IPEEM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	EE
Rsubelement	Ignore
Raddress	
Rrelated1	
Rrelated2	

## File Transfer Event

The column usage for a file transfer event is as follows:

<b>Column</b>	<b>Value</b>
Rclass	FTMON
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	User
Rlevel3	File transfer product
Ridentifier	Source address
Rsubelement	Target address
Raddress	
Rrelated1	
Rrelated2	

## Host Channel

The column usage for a host channel is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPCHPM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	System name
Rlevel1	Channel card name
Rlevel2	
Rlevel3	
Ridentifier	CHPID number
Rsubelement	
Raddress	
Rrelated1	
Rrelated2	

## IP Application Address Space

The column usage for an IP application address space is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPASM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	Address space name
Rsubelement	
Raddress	
Rrelated1	
Rrelated2	

## IP Node

The column usage for an IP node is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPNMON
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Subnetwork
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	Node identifier (name or address)
Rsubelement	Null Interface details Card details Adapter details
Raddress	Node address
Rrelated1	
Rrelated2	

## Open Systems Adapter Support Facility

The column usage for an Open Systems Adapter Support Facility (OSA/SF) is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPOSAM
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	OSA device name
Rsubelement	Ignore
Raddress	OS/390 CHPID
Rrelated1	
Rrelated2	

## SNA Node

The column usage for an SNA node is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPSNA
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	VTAM network name
Rlevel1	Major node name
Rlevel2	SNA line name
Rlevel3	PU name
Ridentifier	LU name
Rsubelement	Qualifier value
Raddress	
Rrelated1	
Rrelated2	

## TCP/IP Stack

The column usage for a TCP/IP stack is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPSTMON
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	
Rlevel3	
Ridentifier	Stack name
Rsubelement	Null Application Subnetwork User
Raddress	
Rrelated1	
Rrelated2	

## TCP/IP Stack Local Interface

The column usage for a TCP/IP stack local interface is as follows:

<b>Column</b>	<b>Value</b>
Rclass	IPIFMON
RperformanceAgent	Foreign key to NetMaster region
Rlevel0	Sysplex name
Rlevel1	System name
Rlevel2	Subnetwork
Rlevel3	Link (not device type) Type (user-specified format)
Ridentifier	Full host name
Rsubelement	Link name (not device name)
Raddress	Stack address
Rrelated1	OS/390 CHPID
Rrelated2	

## Data Retention Rules

---

The default data retention rules are in the Adaptor.ini file. For the location of the file, see Distributed Code in the chapter “Concepts.”

You may change the rules. However, you should:

- Use the default retention rules until you are familiar with Reporter and your own requirements.
- Be aware that expired data is deleted. There is no automatic backup or confirmation prompt.
- Back up the database regularly.
- Before you modify the Adaptor.ini file, back it up.

Changes are picked up the next time the NetMaster Java Framework task starts.

The syntax of a retention rule is as follows. You can have more than one retention rule.

**Tip:** You can use the simulate=Y statement to test the retention rules. Deletions will not occur, but you can view the totals in the NetMaster Java Framework log at the end of the Database Expiry Service process to see what would have been deleted. To reinstate deletions, reset the statement to simulate=N.

## Syntax

*RetentionRule**n* = *data-recording-frequency.attribute-name.resource-class.attribute-type.attribute-category.retention-days*

where:

<i>n</i>	Is a sequence number for the rule (for example, 1 and 2).
<i>data-recording-frequency</i>	Is one of HOURLY, DAILY, or MONTHLY. This is the recording level of the rows covered by this rule. An attribute usually has a different retention period for each recording level. For example, hourly TotalConnections data may be kept for one week, daily TotalConnections data may be kept for six months, and monthly TotalConnections data may be kept for two years.
<i>attribute-name</i>	Is a valid attribute name or %, which means that this rule applies to the facts for any attribute that does not appear in a more specific rule. For information about the attributes, see the Attribute.txt file. For its location, see Distributed Code in the chapter “Concepts.”
<i>resource-class</i>	Is a value under the Rclass column in the Resource table. For more information about the Resource table, see Resource Table in the appendix “Database Structure.”
<i>attribute-type</i>	<p>Is one of COUNTER, ENUMERATED, GAUGE, TOTAL, or %, which means that this rule applies to the facts for attributes of any type.</p> <p>Because the attribute name determines its type, specify % if the attribute name is also specified.</p> <p>Because enumerated fact data takes up more storage than numeric fact data (a character string as opposed to a number), you might want to specify different rules for them.</p>
<i>attribute-category</i>	Is %. Rules do not differentiate between attribute categories.
<i>retention-days</i>	<p>Is 0, an integer from 1 through 9998, or 9999 as follows:</p> <ul style="list-style-type: none"><li>■ 0 indicates that the data be deleted immediately.</li><li>■ 1 through 9998 indicates the number of days to retain the data.</li><li>■ 9999 indicates that the data be kept forever.</li></ul>

## Example 1

The following retention rule keeps the daily aggregated ConTotalBytes data for 180 days:

```
RetentionRule1=DAILY.CONTOTALBYTES.IPSTMON.%.%.180
```

## Example 2

The following retention rule keeps the hourly data for any enumerated attribute from the Channel Interface Processor (CIP) monitor for 14 days:

```
RetentionRule2=HOURLY.%.IPCIPM.ENUMERATED.%.14
```

## Example 3

The following retention rule keeps the monthly aggregated TelTotalBytes data forever:

```
RetentionRule3=MONTHLY.TELTOTALBYTES.IPSTMON.%.%.9999
```

## Example 4

The following retention rule deletes the monthly aggregated data for any attributes from the communications storage manager (CSM) monitor:

```
RetentionRule4=MONTHLY.%.IPCSMM.%.%.0
```



# Report Definition Structure

---

This appendix describes the extensible markup language (XML) structure used to define the reports in Unicenter NetMaster Reporter and illustrates how to use it.

## Top Level

At the top level, the report definition file contains an XML definition and a reference to the document type definition (DTD) as follows:

```
<?xml version="1.0" encoding="ebcdic-cp-us"?>
<!DOCTYPE report SYSTEM "Reporter.dtd">
```

These are followed by exactly one <report> tag. The <report> tag contains everything that is specific to this report. There is a heading and a list of section tags.

The <report> tag can also contain an optional application attribute. The attribute is stored in the report's properties file and is used by the web interface to group related reports. This attribute can take any value; but Reporter defines and uses only two: IP (for reports related to Unicenter NetMaster Network Management for TCP/IP) and FT (for reports related to Unicenter NetMaster File Transfer Management). Reports that do not belong to any particular application (such as DataWarehouseStatus reports) do not use this attribute.

An example follows:

```
<?xml version="1.0" encoding="ebcdic-cp-us"?>
<!DOCTYPE report SYSTEM "Reporter.dtd">
<report application="IP">
  ...report contents...
</report>
```

## Report Heading

A report heading has exactly one `<title>` tag, optionally followed by a `<subtitle>` tag and an optional `<text>` tag. The `<title>` tag contains just a string of text. This should be a short name for the report, for example:

```
<title>Stack Analysis</title>
```

The `<subtitle>` tag (if used) contains text and parameters. It can be used to identify the conditions for which the report was run, for example:

```
<subtitle>for Stack = <parm name=Stack></subtitle>
```

The `<text>` tag (if used) can contain a few lines of text that are used to explain what the report is for. Multiple `<text>` tags are allowed, and each becomes a paragraph in the HTML output. An example follows:

```
<text> This report is for people who need to know the TCP/IP traffic details for an individual stack.</text>
```

The `<title>`, `<subtitle>`, and `<text>` tags must appear in that order, and you cannot have more than one of each of the `<title>` and `<subtitle>` tags.

## Report Sections

Each section consists of a section heading and one or more lumps of data. Each lump of data is the result of executing a structured query language (SQL) query (or, occasionally, several queries). A lump of data can be presented as a chart, a table, or a statement.

### Section Headings

A section heading contains a `<title>` tag and can optionally contain `<text>` tags. As with report headings, these tags contain the name of the section and some explanatory text.

## Section Content

As mentioned, each lump of data can be presented as a chart, table, or statement.

### Charts

A chart has an optional `<title>` tag, a `<query>` tag, and an optional `<footnote>` tag. The chart content is described in more detail in [Charts](#) in this appendix.

### Tables

A table contains an optional `<title>` tag that is just text, a `<query>` tag that defines the data to be displayed, and an optional `<footnote>` tag to pass on handy hints to the user. By default, the table is just a copy of the result returned by executing the query. Alternatively, the table can be turned into a crosstab by including the `summarise=crosstab` attribute in the query (see [Crosstabs](#) in this chapter for details).

An example follows:

```
<table>
  <title>Utilization of CHPID</title>
  <query>
    select pintervalname as Date, nfMinimum as Minimum,
           nfvalue as Average, nfMaximum as Maximum
    from numericfact, resource, period, attribute
    where attribute.attribute = numericfact.attribute
    and resource.resource = numericfact.resource
    and period.period = numericfact.period
    and aname = 'ChpidUtilization'
    and period.pDate='2001-12-31'
    and period.plevel = 'Hourly'
    and rclass='IpOsaM'
    and resource.raddress = '<parm name="OsaChpid"/>'
  </query>
  <footnote>See the "detail" report for more info.</footnote>
</table>
```

### Statements

A statement contains a mixture of text and query results, and is presented as an HTML paragraph, for example:

```
<statement>
An interesting number in the database is <query> select nfValue from numericfact
</query>. An interesting enumeratedFact is <query> select efValue from
enumeratedFact </query>.
</statement>
```

The query can return as much data as you like, but only the first datum (that is, the item in the first column of the first row) is used.

## Charts

Basic types are pie, bar, and line charts. Bar charts and line charts can show multiple sets of data. Bars and lines can be combined on one chart.

The data for a pie chart is provided by embedding a `<query>` tag inside the `<chart>` tag. The result of the query for a pie chart must have at least two columns. The first should be character data and is used to label the pie slices; the second should be numeric and is used to provide the values to be plotted.

An example follows:

```
<chart style="pie">
  <query>
    select aName, nfValue from attribute, numericfact
    where attribute.aName = numericfact.aName
  </query>
</chart>
```

The data for a bar or line chart is provided by one or possibly two `<query>` tags. In the result from a query, the first column is always used as X-axis values and the second and subsequent columns are each treated as a set of data to be plotted. For most charts, it is expected that one query is enough. The left-hand Y-axis is always labeled with values that describe the data in the first (or only) `<query>` tag. If there is a second `<query>` tag, the right-hand Y-axis will be labeled to describe it.

## Pie Charts

The `<chart>` tag for a pie chart is as follows:

```
<chart style="pie">
```

The result used to build the chart must have at least two columns. The first column is used for labels and the second column is used for the numerical data that is plotted. Third and subsequent columns are ignored. There is no way to plot more than one set of data on a pie chart.

---

## Bar Charts

The <chart> tag for a bar chart has two variations as follows:

```
<chart style="bar">  
<chart style="stackedbar">
```

Bar charts can plot one or more sets of data against a set of X-axis values. The first column of the result forms the X-axis values. The second and subsequent columns are the sets of data to be plotted. If there is more than one set of data, a legend is included to identify them.

If a <chart style="bar"> tag contains multiple sets of data, they are plotted side by side with the first set of data on the left.

If a <chart style="stackedbar"> tag contains multiple sets of data, they are stacked vertically with the first set of data at the bottom. A stacked bar chart with only one set of data does not really make a lot of sense, but it does no harm.

## Line Charts

The <chart> tag for a line chart is as follows:

```
<chart style="line">
```

Line charts can plot one or more sets of data against a set of X-axis values. The first column of the result forms the X-axis values. The second and subsequent columns are the sets of data to be plotted. If there is more than one set of data, a legend is included to identify them.

## Combined Bar and Line Chart

The <chart> tag for a combined bar and line chart is as follows:

```
<chart style="linebar">
```

A combined bar and line chart combines a line and a set of bars on one chart. For 3-D charts, the line is plotted in front of the bar. This type of chart can only show two sets of data – you cannot have multiple lines or multiple sets of bars.

## X-axis Labels

By default, only the first and last points on the X-axis are labeled. This can be overridden by specifying an "xlabels" attribute in the chart tag as follows:

```
<chart style="anything-other-than-pie" xlabels="all">
```

Valid values are all, none, and ends.

## Queries

The `<query>` tag contains text that is passed as an SQL query to the database. The text can contain parameter values to control what is selected. The text is always converted to upper case before being sent to the database.

An example follows:

```
<query>
  select * from period where pRelativeday > <parm name="startRelativeDay" />
</query>
```

## Crosstabs

The result from a query can be converted from a simple table to a crosstab by specifying the `summarise=crosstab` attribute in the `<query>` tag. This attribute causes the values in the first column of the result to be used to label rows in the output, values from second column of the result to be used to label the columns of the output, and values from the third column of the result to be used as the actual data. Any other columns in the result are ignored. The usual case is that the first column contains dates, the second column contains attribute names, and the third column contains numeric facts.

An example follows.

```
<query summarise="crosstab">
  select pIntervalname, aName, nfValue from period, attribute, numericfact
  where numericfact.period = period.period
  and numericfact.attribute = attribute.attribute
  and aName in ( 'BytesIn', 'BytesOut' )
</query>
```

The query might return the following set of data:

2001-12-01	BYTESIN	123
2001-12-01	BYTESOUT	456
2001-12-02	BYTESIN	789
2001-12-02	BYTESOUT	12

This set of data would be converted to:

	BYTESIN	BYTESOUT
2001-12-01	123	456
2001-12-02	789	12

## Parameters

A report definition can have a corresponding list of required parameters, which is stored elsewhere in *prefix/reporter20/prompts/report-name.txt*. There must be a corresponding `<prompt>` tag for each parameter. Where parameters are related to each other, they can be grouped by using the `group` attribute in the `<prompt>` tag. This attribute causes Reporter to check that the values chosen by the user are compatible with each other.

Some examples follow:

```
<prompt name="System" group="SystemNameGroup">
<prompt name="Sysplex" group="SystemNameGroup">
<prompt name="Timeframe">
```

The first prompt is called the report's criteria and is used to help identify one instance of a report from another.

Within a group, the sequence of the prompts is important—it must match what is in the prompts file.

When a report is run, Reporter provides a value for each of these parameters. The parameter names and the parameter values are text strings.

The value of a parameter can be substituted into a `<query>` tag or into the report's `<subtitle>` tag by including a `<parm>` tag within the text of the query or subtitle. The `<parm>` tag requires a 'name' attribute, but has no other content.

Parameter names are case-sensitive. So `<parm name="TimeFrame" />` and `<parm name="timeframe" />` are not the same.

An example follows:

```
<query>
  select * from period where pLevel = <parm name="periodLevel" />
</query>
```

The prompts file should contain an entry for each distinct name used in any `<parm>` tag. When a report is run, every parameter must have a value (even if only an empty string); otherwise, the report generator will reject the request.

## Time Frames

Many reports operate over a selected time frame. Some special processing is included to support this. If the report's parameter list contains a parameter named Timeframe, Reporter creates some extra time frame related parameters and gives them values that correspond to the value of the Timeframe parameter. The extra, automatically created, parameters are StartRelativeDay, EndRelativeDay, StartDate, EndDate, and PeriodLevel.

For example, if today is 31 December 2000 and a report needs a Timeframe parameter with the value LAST 7 DAYS, then at the start of processing the report, the following parameters are created:

```
StartRelativeDay=11316  
EndRelativeDay=11322  
StartDate=2000-12-24  
EndDate=2000-12-30  
PeriodLevel=DAILY
```

## Document Type Definition

Here is a copy of the DTD that controls the accepted syntax:

```
<?xml version="1.0" encoding="ebcdic-cp-us"?>
<!ELEMENT report (prompt*, title, subtitle?, text*, section+)>
<!ATTLIST report
  application CDATA #IMPLIED
>
<!ELEMENT prompt EMPTY>
<!ATTLIST prompt
  name CDATA #REQUIRED
  group CDATA #IMPLIED
>
<!ELEMENT section (title, text*, (chart | statement | table)+)>
<!ELEMENT chart (title?, query+, footnote?)>
<!ATTLIST chart
  style (pie | bar | line | linebar | stackedbar) "bar"
  xlabel (all | ends | none) "ends"
>
<!ELEMENT statement (#PCDATA | query | reference | title)*>
<!ELEMENT table (title?, query, footnote?)>
<!ELEMENT parm EMPTY>
<!ATTLIST parm
  name CDATA #REQUIRED
>
<!ELEMENT query (#PCDATA | parm | column)* >
<!ATTLIST query
  summarise (crosstab | none) "none"
  name CDATA #IMPLIED
>
<!ELEMENT column EMPTY>
<!ATTLIST column
  display CDATA #IMPLIED
  field CDATA #REQUIRED
>
<!ELEMENT reference EMPTY>
<!ATTLIST reference
  name CDATA #REQUIRED
  column CDATA #REQUIRED
>
<!ELEMENT subtitle (#PCDATA | parm)*>
<!ELEMENT text (#PCDATA)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT footnote (#PCDATA)>
```

The special characters that control repetition of objects in a DTD are as follows:

Character	Repetition of Objects
*	Zero or more are required
+	One or more are required
?	Zero or one is allowed
or none	Exactly one



# Migrating IPTREND Data to the Database

If your NetMaster for TCP/IP region has been collecting IPTREND data, you can migrate the data to the NetMaster Reporter database. Before you do your migration, you should have completed implementation as described in the chapter “Implementation.”

The migration process converts the IPTREND data and sends the converted data to the database. The IPTREND data does not itself contain information about the sysplex, system, and TCP/IP stack for the region that collects the data. NetMaster Reporter requires this information. To correct this, the conversion process will get the information from the region in which the conversion takes place and add it to the converted data. If the IPTREND data was collected by NetMaster for TCP/IP 6.2, you can perform the conversion in the region that collects the data. If the IPTREND data was collected by a NetMaster for TCP/IP region earlier than Version 6.2, then you should perform the conversion in a Version 6.2 region that is in the same sysplex and on the same system, and that uses the same TCP/IP stack as the original region that collects the data.

**Tip:** If your NetMaster for TCP/IP 6.2 region has been collecting IPTREND data, you can disable the collection after you have set up the region to feed data to the Reporter database. For information about how to disable the collection of IPTREND data, see [Disabling the IPTREND Data Set](#) in this appendix.

## Determining Whether the IPTREND Data Should Be Migrated

Migration of the IPTREND data is optional. Before migrating the data, you should consider the following:

- If you have been monitoring stack trends over a long period, migrating your IPTREND data allows you to retain your stack workload statistics.
- Use NetMaster Reporter with the sample data. This enables you to determine if migration will be useful.
- Only part of the IPTREND data is converted:
  - Stack monitor (\$IPSTMON) data is converted.
  - Data from the address space performance, the communication storage manager (CSM), and the stack local interface workload performance monitors is not converted.
  - IP node (\$IPNMON) and channel card (\$IPCIPM) device data conversion is optional. This data is of limited use because it is not retained in IPTREND for long periods. If you do convert device data, convert it separately from the stack data to reduce resource requirements.
  - Data from 2216, local interface, Open Systems Adapter (OSA), and Enterprise Extender devices is not converted.
  - Enumerated attributes such as Netstatus and Pustatus are not converted.
- IPTREND time frames are modified in the conversion process:
  - Weekly records are aggregated into monthly rows.
  - Daily records are converted directly into daily rows.
  - Four hourly records are not converted.
- The conversion process may take several hours. The time taken is dependent on:
  - The size of the IPTREND data set
  - The load on the region in which the conversion takes place
  - Whether you are converting stack data, device data, host data, or all types of data
  - The processing speed, memory size, and disk characteristics of your system
  - The queue depth of the data warehouse server, and your use of the QDEPTH parameter in the TRENDCONV command

## Conversion Issues

If you decide to convert the IPTREND data, be aware that:

- If IPTREND data is collected while the region is also sending data to the Reporter database, dates will overlap. If a fact exists already, subsequent attempts to add the same fact will fail. This does not affect normal processing.
- Converting the same IPTREND data set twice causes all consequent database update attempts to fail because the data records already exist. This does not effect normal processing but is strongly discouraged. It wastes CPU resources and may cause data sent directly to the database to be queued unnecessarily.
- Device data is retained in the database only for a specified period. Ensure that the retention period is long enough to prevent the converted IPTREND data from expiring soon after it is converted. For information about data retention, see Database Expiry Service in the chapter “Concepts.” In general, it is rarely worth converting device or host data.

## Preparing Your VFS File

Converted IPTREND data is stored in your Virtual File Services (VFS) data set before it is sent to the database. You must increase the size of your VFS data set if your IPTREND data set has more than 20,000 records. To increase the size of your VFS data set, proceed as follows:

1. Bring the region in which you want to perform the migration down, and define a temporary VFS data set in the same size as your original VFS data set.
2. Use the REPRO command to copy data in the original VFS data set into the temporary VFS data set.
3. Delete your original VFS data set, and redefine it so that it can hold 75% of the number of records in the IPTREND data set. For example, if your IPTREND data set has 20,000 records, define the new VFS data set to 15,000.
4. Use the REPRO command to copy the data in the temporary VFS data set back into the enlarged VFS data set.
5. Restart the region, and proceed with the migration as described in [Migrating Your IPTREND Data Set](#) in this appendix.
6. When the migration has completed, you can reverse this procedure to restore your VFS data set to its original size.

## Migrating Your IPTREND Data Set

Migrate your IPTREND data set as follows:

1. Enter the **/DWS** panel path shortcut to display the Data Warehouse Services : Server List panel.
2. Ensure that the REPORTER\$DWS server is:
  - Defined and active
  - Configured for all data classes in the \$IP application ID
3. Enter the **=O** panel path to display the Operator Console Services (OCS) panel.
4. At the => command prompt, enter **TRENDCONV HELP**.
5. Enter the **/LOG** panel path shortcut to display the log to view the help on the TRENDCONV command.
6. Follow the instructions in the help to perform the migration.

## Disabling the IPTREND Data Set

Disable the IPTREND data set so that it no longer collects data as follows:

1. Enter the **/ICS** panel path shortcut to list the region initialization and customization parameter groups.
2. Enter **U** beside the IPFILES parameter group to display the TCP/IP File Specifications window.
3. Delete the value in the IPTREND Trend Dataset field.
4. Press F6 (Action), and then press F3 (File).

**Note:** Disabling the IPTREND data set does not invalidate the TREND reporting level in the TCP/IP resource monitor definitions. Any value other than NONE causes performance data to be collected.

## Frequently Asked Questions

**Q:** Is it safe to run TRENDCONV again if it does not complete?

**A:** Yes, the rows that were added the first time will be ignored the second time. Error messages about primary key constraints are written to the NetMaster Java Framework log.

**Q:** Can TRENDCONV be run separately for stack and device data?

**A:** Yes, you can.

**Q:** After TRENDCONV has been run, can it be run again on a newer IPTREND data set?

**A:** Yes, all eligible records are converted. However, most of the records will not be written to the database because they already exist on it. It is better to start NetMaster Reporter's data collection before migrating the IPTREND data. This ensures that the data collected is contiguous with the converted data.



# DB2 Administration

---

This appendix describes the DB2 administration considerations for the Unicenter NetMaster Reporter database.

## Improving the Response Time for the DB2 Queries

If report generation is slow, the most probable cause is DB2 performance. The DB2 query response time can be affected adversely by environmental factors such as:

- Database size
- Database tuning
- Other concurrent DB2 activities
- System capacity and loading

To insure the most efficient usage in CPU time and I/O, the RUNSTATS online utility should be run periodically. Typically, your DB2 administration team should ensure that RUNSTATS is scheduled for the DB2 objects. This is especially true as the volumes within the tables grow after creation. From this perspective, it might be prudent to run the utility daily for a time until there is a representative volume of data within the DB2 tables.

## Reserving Free Space in Table Spaces and Indexes

You should monitor the free space within the table spaces and indexes associated with the DB2 objects closely. Your DB2 administration team should be able to advise if you experience negative performance because of poor organization of data and the lack of free space. If there is a lack of free space, increase the free space parameters, and then reorganize the table space and index. If the free space parameters are adequate, just reorganize the table space and index. These actions result in the following benefits:

- Improved row access because of the clustering of data
- More space for data resulting in fewer overflows
- Less frequent reorganization required because of fewer overflows and fragmentation
- Fewer index page splits and consequently more effective index access to data rows because of more space for index entries

## Analyzing Buffer Pools to Support an Analytical Workload

Because of the nature of the analytical processing of the DB2 objects, you should, if performance problems occur, analyze the buffer pool definition. Ensuring that you have efficient buffer pools allocated to the DB2 objects means:

- Fewer I/O operations resulting in faster access to your data
- Reduced I/O contention with data as well as indexes
- Reduced sorting effort and time for work files for analytical queries.

If performance problems exist, discuss with your DB2 administration team to analyze the typical workloads for Unicenter NetMaster Reporter.

## Backing Up and Recovering

The daily management of the DB2 objects typically falls within the management boundaries already defined within your organization. Back up the table spaces within the production database. The type of backup taken is flexible and can be determined by the site standards defined by your DB2 administration team. Similarly, the frequency of backup should be balanced with the time it takes for forward-log recovery. Growth analysis should be completed by your DB2 administration team.

# Index

## A

---

Adaptor.ini file, 2-5  
Aggregation Service, 2-3  
Attribute table, B-1

## B

---

bar and line charts, combined, D-5  
bar charts, D-5

## C

---

CA-ACF2, A-1  
CA-Top Secret, A-2  
charts  
    bar, D-5  
    bar and line, combined, D-5  
    line, D-5  
    pie, D-4  
    X-axis labels, D-5  
Cisco channel card interface, B-3  
Cisco CLAW, B-4  
Cisco TN3270 server, B-4  
communications storage manager. *See* CSM  
configuration, typical, 2-1  
Control Service, 2-3  
crosstabs, D-6

CSM (communications storage manager), B-5  
cursor, troubleshooting, 7-1

## D

---

data  
    collection, 3-10  
    consolidation, 1-3, 2-1  
    model, B-1  
    retention rules, 2-3, C-1  
    sources, 1-2  
    working, 2-6  
data regions, 2-1, 2-2  
Data Warehouse Service, 2-4  
database  
    access, troubleshooting, 7-2  
    connection, troubleshooting, 7-1, 7-2  
    implementation, 3-8  
Database Expiry Service, 2-3  
database tables  
    Attribute, B-1  
    EnumeratedFact, B-1  
    Keyrange, B-1  
    names, B-1  
    NumericFact, B-1  
    PerformanceAgent, B-1  
    Period, B-1  
    Resource, B-1  
    Timeframe, B-1  
    troubleshooting, 7-3

---

## DB2

- backup, F-2
- buffer pools, F-2
- interface implementation, 3-3
- JDBC, 3-4
- location name, 3-4
- recovery, F-2
- response time, F-1
- SQLJ, 3-4
- table spaces, F-2

## directories

- distributed code, 2-5
- log, 2-6
- prefixes, 2-5
- report output, 2-7
- working data, 2-6

document type definition (DTD). *See* DTD

DTD (document type definition), D-9

---

## E

Enterprise Extender, B-5

EnumeratedFact table, B-1

---

## F

file transfer event, B-6

## files, 2-5

- distributed code, 2-5
- log, 2-6
- report output, 2-7
- working data, 2-6

---

## H

help, 4-2

HFSSECURITY parameter group, 3-7

home directory, A-2

host channel, B-6

housekeeping time, 6-1

---

## I

## implementation

- data collection, 3-10
- databases, 3-8
- DB2 interface, 3-3
- Java 2 Technology Edition, 3-3
- knowledge requirements, 3-1
- Report Center, 3-9
- UNIX System Services authorization, 3-5
- verification, 3-10
- web interface port, 3-9

IP application address space, B-7

IP node, B-8

IPTREND data, E-1

---

## J

Java 2 Technology Edition implementation, 3-3

JDBC, 3-4

---

## K

Keyrange table, B-1

---

## L

line charts, D-5

logon, troubleshooting, 7-4

---

## M

migration, 3-7, E-1

multisystem considerations, 3-8

---

## N

---

NetMaster Java Framework, 2-1, 2-3  
  Aggregation Service, 2-3  
  Control Service, 2-3  
  Data Warehouse Service, 2-4  
  Database Expiry Service, 2-3  
  Prompt Generator Service, 2-4  
  Report Expiry Service, 2-4  
  Report Generator Service, 2-4  
  Scheduler Service, 2-4  
  start, 3-10  
    troubleshooting, 7-4  
  Timeframe Service, 2-5  
  UNIX System Services authorization, 3-5

NumericFact table, B-1

## O

---

OMVS  
  segment, A-1  
  UID, A-1

on demand reports  
  display, 5-3  
  kept period, 4-3  
  runs, 5-2

Open Systems Adapter Support Facility. *See* OSA/SF

OSA/SF (Open Systems Adapter Support Facility), B-9

## P

---

parameter groups  
  HFSSECURITY, 3-7  
  REPORTER, 3-4, 3-9, 3-10, 6-1  
  SOCKETS, 3-9

PerformanceAgent table, B-1

Period table, B-1

pie charts, D-4

Prompt Generator Service, 2-4

## R

---

RACF, A-2

Report Center, 2-1, 2-2  
  implementation, 3-9  
  UNIX System Services authorization, 3-5

report definitions, 2-5

Report Expiry Service, 2-4

Report Generator Service, 2-4

REPORTER parameter group, 3-4, 3-9, 3-10, 6-1

reports  
  access, 2-2, 4-1  
  display, 4-2, 5-3, 6-2  
  links, 4-3  
  location, 3-1  
  on demand, 5-2  
  on demand runs, 1-2  
  predefined, 1-2  
  scheduled, 6-2  
  scheduled runs, 1-2  
  XML structure, D-1

Resource table, B-1  
  Cisco channel card interface, B-3  
  Cisco CLAW, B-4  
  Cisco TN3270 server, B-4  
  CSM, B-5  
  Enterprise Extender, B-5  
  file transfer event, B-6  
  host channel, B-6  
  IP application address space, B-7  
  IP node, B-8  
  OSA/SF, B-9  
  SNA node, B-10  
  TCP/IP stack, B-11  
  TCP/IP stack local interface, B-12

## S

---

scheduled reports  
  display, 6-2  
  housekeeping time, 6-1  
  kept period, 4-3  
  maintenance, 6-3  
  runs, 6-2

Scheduler Service, 2-4

SNA node, B-10

---

SOCKETS parameter group, 3-9

SQLJ, 3-4

## T

---

TCP/IP stack, B-11

TCP/IP stack local interface, B-12

test, function, 3-10

time frames, D-8

Timeframe Service, 2-5

Timeframe table, B-1

timer, 2-3

## U

---

UNIX System Services authorization, 3-5

- CA-ACF2, A-1

- CA-Top Secret, A-2

- home directory, A-2

- NetMaster Java Framework, 3-5

- OMVS segment, A-1

- OMVS UID, A-1

- RACF, A-2

- Report Center, 3-5

- users, 3-7

## W

---

web access URL, 3-9

web interface port implementation, 3-9

## X

---

XML structure

- charts, D-3, D-4

- heading, D-2

- parameters, D-7

- queries, D-6

- sections, D-2

- statements, D-3

- tables, D-3

- top level, D-1