DB2 Enterprise Control Center for TME 10 V1.2

IBM

User's Guide

DB2 Enterprise Control Center for TME 10 V1.2

IBM

User's Guide

> **Note!**
>
> Before using this information and the product it supports, please read the general information under "Notices" on page vii.

Contents

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling (1) the exchange of information between independently created programs and other programs (including this one) and (2) the mutual use of the information that has been exchanged, should contact:

IBM Corporation
555 Bailey Avenue, W92/H3
PO Box 49023
San Jose, CA
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX
DATABASE 2
DataJoiner
DataPropagator
DB2
Distributed Relational Database Architecture
DRDA
IBM
MVS/ESA
OS/2
OS/390
RACF
SystemView

The following terms are trademarks of Tivoli Systems, Inc., an IBM Company, in the United States or other countries or both:

vii

Tivoli
Tivoli/Enterprise Console
Tivoli Management Environment
Tivoli Mangement Framework
Tivoli Management Platform
Tivoli/Sentry
TME
TME 10
TME 10 Distributed Monitoring
TME 10 Enterprise Console

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

About This Book

With DB2 Enterprise Control Center (DB2 ECC), IBM offers a powerful new way to manage DB2 common servers and DB2 Universal Databases residing on an AIX, NT, or Solaris-based platform. This book is intended to help you install and use DB2 ECC to improve database and systems management across your enterprise.

Who Should Read This Book

This book is written for database and system administrators who are involved in installing, setting up the environment, managing DB2 servers and databases, and solving problems in a DB2 ECC environment.

You should have knowledge of the AIX, Sun Solaris, or Windows NT operating system and the Tivoli Management Environment.

You should also know the basic functions of the relational database management systems (RDBMSs) that will be used and managed by DB2 ECC.

Prerequisites

The *TME 10 Framework User's Guide* explains how to plan for, set up, and manage the Tivoli Management Environment (TME 10 environment) with the TME 10 desktop and the TME 10 command line interface. You should be familiar with Tivoli concepts before you use DB2 ECC.

How to Use This Book

This book is designed to be used in conjunction with Tivoli documentation. Some of the tasks and topics in this book are explained in more detail in Tivoli books. This book is divided into seven chapters:
Read this chapter before you install and use DB2 ECC.

- "Chapter 1. Introduction to DB2 ECC" on page 1, provides an introduction to the DB2 ECC product. It explains the capabilities and basic components of DB2 ECC and how DB2 ECC fits into the Tivoli Management Environment.

Read these chapters as you perform the tasks to get up and running with DB2 ECC.

- "Chapter 2. Installing DB2 ECC" on page 11, explains the tasks required before you install or upgrade and provides instructions for installing or upgrading DB2 ECC.

- "Chapter 3. Setting Up Your DB2 ECC Environment" on page 23, provides instructions for setting up your DB2 ECC environment after installation to get you up and running.

Read these chapters as you perform the corresponding DB2 ECC functions. These chapters do not need to be read in any particular order.

- "Chapter 4. Working with DB2 Managed Resources" on page 31, provides instructions for creating and working with DB2 managed resources (instances, databases, partitions, and partition groups).

- "Chapter 5. Working with the DB2 ECC Task Libraries" on page 49, provides instructions for working with the task libraries provided with DB2 ECC.

- "Chapter 6. Working with DB2 ECC Monitoring Collections" on page 61, provides information and instructions on setting up and activating monitors and managing your monitoring environment.
Read this chapter if you run into problems installing or using DB2 ECC.

- "Chapter 7. Troubleshooting" on page 85, provides tips and hints useful if DB2 ECC is not working properly.

The book also contains several appendixes, a bibliography, a glossary, and an index.

Terminology

You can find definitions of DB2 ECC terms in the text of this book. When the book introduces new terms, it italicizes and defines them. You can also find definitions in "Elements in the DB2 ECC Environment: Terminology" on page 1, which contains high-level concepts, and in "Terms and Abbreviations" on page 133, which contains definitions of DB2 ECC terms and technical terms that pertain to DB2 ECC and Tivoli.

Table 1 provides a brief list of common DB2 ECC terms and shortened names used in this book.

Table 1. DB2 Enterprise Control Center Basic Terminology

Term	Definition
DB2	IBM DATABASE 2
DB2 Enterprise-Extended Edition	IBM DB2 Universal Database Enterprise-Extended Edition, Version 5
DB2 Enterprise-Extended Edition for AIX	IBM DB2 Universal Database Enterprise-Extended Edition for AIX, Version 5
DB2 ECC	DB2 Enterprise Control Center
DB2 for AIX	IBM DATABASE 2 for AIX, Version 2
DB2 for common servers	IBM DATABASE 2 for common servers, Version 2
DB2 for NT	IBM DATABASE 2 for Windows NT, Version 2
DB2 for OS/390	IBM DB2 for MVS/ESA Version 4 or IBM DB2 for OS/390 Version 5
DB2 UDB	IBM DB2 Universal Database, Version 5
DB2 UDB for AIX	IBM DB2 Universal Database for AIX, Version 5
DB2 UDB for NT	IBM DB2 Universal Database for Windows NT, Version 5
DB2 UDB for Solaris	IBM DB2 Universal Database for Solaris, Version 5
Sentry	Tivoli/Sentry or TME 10 Distributed Monitoring
IBM Replication	IBM DataPropagator Relational or IBM Replication
NT	Microsoft Windows NT operating system
Solaris	Sun Microsystems Solaris operating system for UNIX workstations
TME	Tivoli Management Environment
TMR	Tivoli Management Region

Typographic Conventions

This book uses the following typographical conventions:

Bold

Used to indicate:

- AIX, NT, or Solaris commands and keywords
- Menus, push buttons, check boxes, list boxes, and other field names in the windows

Italics

Used to indicate:

- General emphasis
- The introduction of a new term
- Titles of books

`Monospace`

Used to indicate:

- Information that you are instructed to type at a command prompt or window
- Examples of specific data values
- Examples of text similar to what might be displayed by the system
- Examples of system messages

Using This Book Online

This book is available in PDF format on the installation media. You can display the PDF file, ECCU1200.PDF, with the Adobe Acrobat Reader 3.01. DB2 ECC provides the Adobe Acrobat Reader on the installation media in three formats: AIX, Solaris, and Windows NT. To install and use the Adobe Acrobat Reader, see "Installing the Online Books and Reader" on page 19.

The ECCU1200.PDF file can be printed on any type of printer.

Chapter 1. Introduction to DB2 ECC

This chapter provides an introduction to DB2 Enterprise Control Center for TME 10 V1.2 (DB2 ECC), a summary of the versions and platforms of DB2 that DB2 ECC supports, a description of the basic elements of DB2 ECC, and an explanation of how DB2 ECC fits into the Tivoli framework.

What Is DB2 ECC?

DB2 ECC allows you to manage DB2 for common servers databases and DB2 UDB databases across your enterprise from a single TME 10 desktop. A database administrator can use DB2 ECC to efficiently manage DB2 for AIX, NT, and Solaris databases in a distributed environment of any size. DB2 ECC manages groups of DB2 databases in a secure, consistent, and scalable manner by making use of the Tivoli systems management framework. The Tivoli Management Environment (TME) 10 desktop provides a consistent, graphical user interface to databases and management functions for IBM's DB2 for AIX, NT, and Solaris relational database management systems (RDBMSs).

DB2 ECC is an effective tool for database systems management when you need to:
- Manage many (a few to hundreds of) DB2 servers from a single location
- Manage DB2 servers in a large, geographically dispersed environment
- Manage many DB2 servers organized in groups based on business needs

Supported Versions of DB2

Table 2 shows the versions and platforms of DB2 that DB2 ECC V1.2 supports.

Table 2. DB2 Versions and Platforms Supported by DB2 ECC V1.2

Version	Platform		
	AIX	**Solaris**	**Windows NT**
DB2 for common servers	✔		✔
DB2 Universal Database	✔	✔	✔
DB2 Enterprise-Extended Edition	✔		

Elements in the DB2 ECC Environment: Terminology

A typical DB2 ECC environment requires the integration of many products and elements. This environment introduces some new concepts. This section explains some of the DB2 ECC and Tivoli concepts and the terminology used in this book and product.

TME 10 Desktop

The *TME 10 desktop* is a user interface that provides point-and-click access to TME features and components. The desktop provides a central control point for you to organize, manage, and delegate system management tasks.

Elements in the DB2 ECC Environment: Terminology

Figure 1 shows a typical TME 10 administrator's desktop.
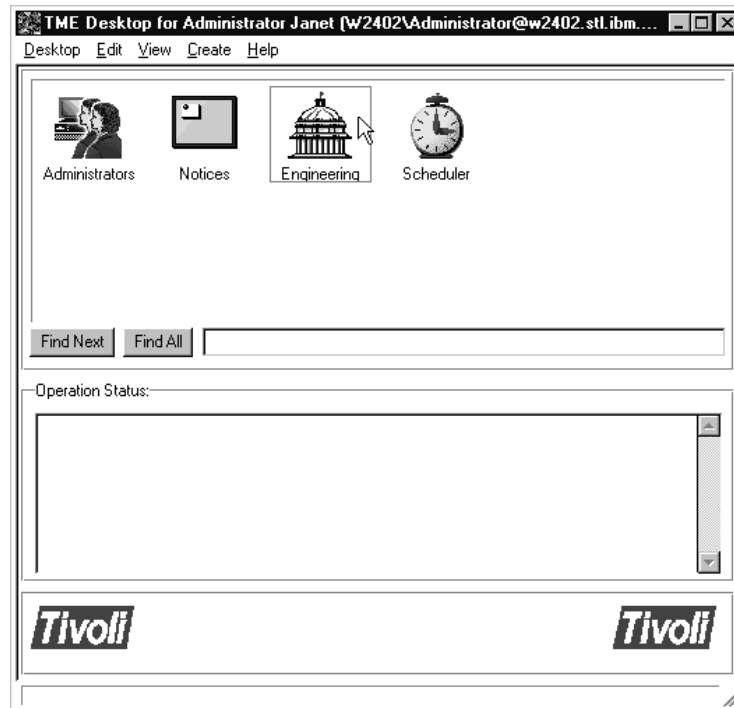
Figure 1. A TME 10 Desktop for a TME 10 Administrator

Tivoli Management Regions

A *Tivoli Management Region (TMR)* is a TME server and the set of managed nodes that it serves. Depending on the size and operational requirements of your organization, you might have more than one TMR.

Policies and Policy Regions

A *policy* is a rule that you put into effect for a system. You use policies to customize TME to fit your needs. The TME enforces the policy when management operations are performed by administrators. For example, you can implement a policy that determines on which managed nodes a particular task or job can run.

TME maintains and enforces policies within *policy regions*. A policy region is a collection of resources that share one or more common policies. Using policy regions, you can construct a model of the organization through which system management tasks and operations are performed. Policy regions can contain any set of managed resources that you want.

Figure 2 on page 3 shows an example of a typical policy region and the managed resources that it contains.
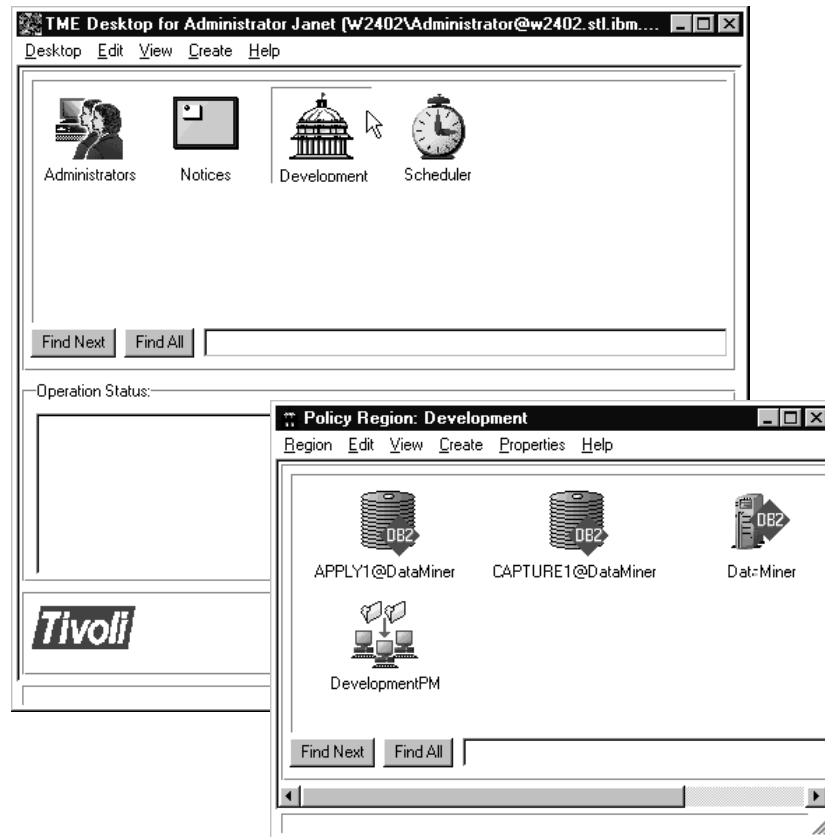
Figure 2. A Policy Region and the Managed Resources That It Contains

DB2 ECC Default Policy Region

When you install DB2 ECC, the DB2 ECC default policy region (DB2_ECC-DefaultPolicyRegion) is created in your TMR. This policy region contains tools that you need to manage your DB2 managed resources. These tools include task libraries and Sentry profiles with DB2 ECC default monitors to support DB2 monitoring and task automation.

Figure 3 on page 4 shows the DB2 ECC default policy region and its contents.

Elements in the DB2 ECC Environment: Terminology
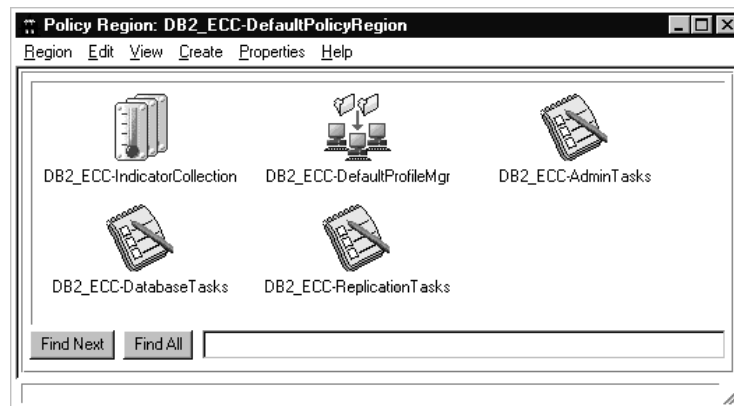
Figure 3. The DB2 ECC Default Policy Region

The DB2 ECC default policy region contains the following components:

DB2 ECC default profile manager
> The central subscription point to which each DB2 managed resource is to be subscribed. The DB2 ECC default profile manager (DB2_ECC-DefaultProfileMgr) is automatically subscribed to all of the jobs in the DB2 ECC task libraries and contains three default profiles (where you select which monitors to use). The three default profiles are:

> **DB2_ECC-DatabaseMonProfile**
> > The profile that contains the DB2 database default monitors.

> **DB2_ECC-DBMSMonProfile**
> > The profile that contains the DB2 nonpartitioned database server default monitors.

> **DB2_ECC-DBPartMonProfile**
> > The profile that contains the DB2 partition default monitors.

> **DB2_ECC-GenericMonProfile**
> > The profile that contains the DB2 ECC administration and operating system default monitors.

DB2_ECC-IndicatorCollection
> The indicator collection that provides status information about the DB2 managed resources defined in the DB2 ECC default policy region. This status information can be information captured by the monitors that you set up.

DB2_ECC-AdminTasks
> The task library that contains monitoring and administration tasks, such as starting and stopping the DB2 ECC monitoring agent and broadcasting a message to all TME 10 desktops.

DB2_ECC-DatabaseTasks
> The task library that contains a collection of DB2 utilities that perform functions such as backing up a database and reorganizing a table.

DB2_ECC-ReplicationTasks
> The task library that contains a set of replication tasks, such as starting and stopping the IBM Replication Capture and Apply programs.

DB2 Managed Resources and Endpoints

A Tivoli *managed resource* represents a system or network resource that you manage with TME 10. A managed resource is a specific resource that has a default policy defined in the policy region. An *endpoint* is a managed resource that is the target for distribution of a profile, or the resource on which a task or job is to be run.

DB2 ECC provides four types of managed resources:

- *DB2 instances*
- *DB2 databases*
- *DB2 partitions*
- *DB2 partition groups*

When you create a DB2 managed resource object, the DB2 instance, database, partition, or partition group becomes a managed resource of DB2 ECC. This arrangement allows you to perform DB2 ECC management tasks, such as running tasks and monitoring, on the DB2 managed resource.

You must define a managed resource object for each DB2 instance, database, partition, and partition group in your environment. For example, if your network contains 20 DB2 instances that you want to manage, you need to create 20 DB2 instance objects within policy regions.

The following figure shows the icons for DB2 instance, database, partition, and partition group objects.

DB2 Instance   DB2 Database   DB2 Partition   DB2 Partition Group

Profiles and Profile Managers

A *profile* is a collection of application-specific information. The information in a profile is specific to the particular profile type. For example, a user profile might contain information such as the user name, login name, user ID, user group, and home directory for each user that is defined in the profile.

Several profiles are required to describe the entire configuration of an endpoint. The TME provides *profile managers* to organize groups of profiles. Profile managers control the distribution of profiles to subscribers across an entire network or across a specific portion of a network.

Figure 4 on page 6 shows a profile manager, the profiles that it contains, and its subscribers.
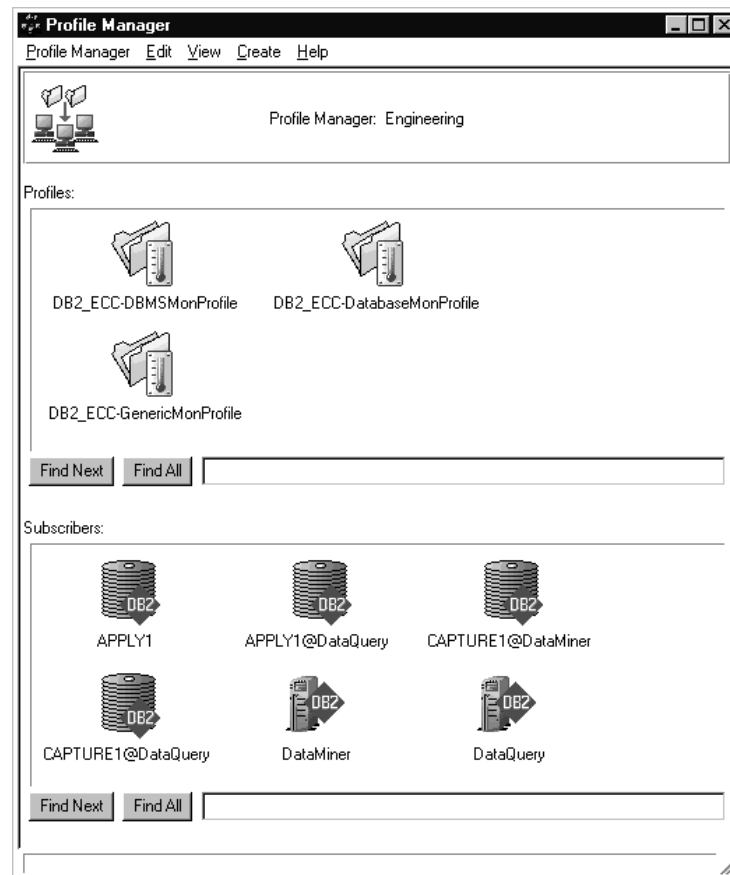
Figure 4. A Profile Manager, the Profiles That It Contains, and Its Subscribers

Administrators

A *TME 10 administrator* is an administrator who has been established as a TME 10 administrator. TME 10 administrators can perform tasks and manage policy regions in one or more TMRs. A TME 10 administrator in the DB2 ECC environment is an administrator who has been granted the privileges to perform DB2 database management tasks.

The initial TME 10 administrator, or *root* administrator, is added when TME 10 is installed. After TME 10 is installed, you can define and give authorization roles to other nonroot administrators. Based on these roles, administrators can perform assigned database and system management tasks. This ability to assign different levels of authority gives senior administrators complete control over who can perform certain operations on different sets of resources.

Figure 5 on page 7 shows the Administrators window with icons for some sample TME 10 administrators.
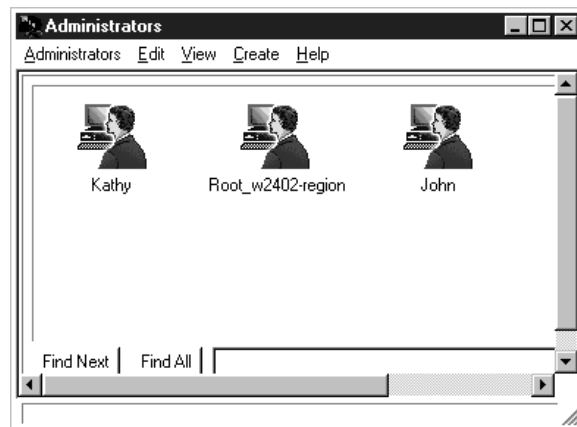
Figure 5. The Administrators Window with Icons for Some Sample Administrators

Notices and Notice Groups

The TME provides a notification facility that tracks system administration activity. The notification facility informs administrators of system management operations and reports which administrator performed a particular operation. The notification facility is especially useful in large enterprises that have a number of administrators because it can serve as an audit trail to determine who did what in the system.

The notification facility uses *notices* and *notice groups* to track system management activities. A notice is a message concerning some operation or change in the distributed system. Notices are sent to application or operation-specific notice groups. A notice group stores and distributes messages pertaining to specific TME functions. For example, the TME Administration notice group receives notices from such operations as creating a new administrator or changing the set of managed resources supported by a policy region.

DB2 ECC provides two notice groups:

DB2 ECC Log
The DB2 ECC Log notice group logs errors and user information posted from DB2 ECC actions.

DB2 Sentry
The DB2 Sentry notice group logs DB2 monitoring source output posted from DB2 ECC monitors. DB2 ECC allows you to send DB2 monitoring source output to the DB2 Sentry notice group as well as to other Sentry notice groups.

The following figure shows the icon for notices.

Task Libraries

A *task library* is a collection of tasks and jobs that can be run on one or more managed resources in a network. DB2 ECC provides about 60 tasks in three task libraries. The task libraries are:

- Monitoring and Administration Tasks
- DB2 Database Tasks
- Data Replication Tasks

The following figure shows the icons for the DB2 ECC task libraries.

DB2_ECC-DatabaseTasks   DB2_ECC-AdminTasks   DB2_ECC-ReplicationTasks

You can run a task in a DB2 ECC task library on a DB2 managed resource immediately or schedule the task as a job to be run at a later time. You can also define a monitor to invoke a task to run.

Monitoring Collections

A *monitoring collection* is a group of related monitors that you use to monitor different aspects of your managed resources.

The DB2 ECC monitoring collections provide over 375 monitoring sources that help you use Tivoli/Sentry to manage distributed DB2 resources (instances, databases, partitions, and partition groups). Each collection is a group of monitoring sources that help you manage a different aspect of your DB2 installations. For example, the DB2 table monitoring collection contains monitoring sources that monitor table row activity. In general, the DB2 ECC monitoring collections are based on the various DB2 monitor groups. For information on the DB2 monitor groups, see the *DB2 System Monitor Guide and Reference* for the version of DB2 that you are using.

The eleven DB2 ECC monitoring collections are:

- DB2 agents and applications
- DB2 basic
- DB2 buffer pool I/O
- DB2 configuration
- DB2 data replication relational
- DB2 locks and deadlocks
- DB2 query
- DB2 SNMP
- DB2 sort work
- DB2 SQL statement activity
- DB2 table

Indicator Collections

Enterprise environments often have a lot of resources to monitor. To keep track of all of them, you might need to check every individual Sentry profile for the state of its monitors. To avoid this problem, Tivoli/Sentry provides indicator collections.

An *indicator collection* is a window that has an icon for each of the profiles affiliated with it. When a monitor reaches one of the available response levels (normal, warning, severe, critical) the indicator collection changes the profile icon according to the level reached. There is no icon for the *always* response level because this level does not indicate an abnormal state. A profile icon shows only the highest response level of all of the monitors in the Sentry profile that it represents. The following figure shows the corresponding profile icons for each response level.

TME 10 Scheduler

The TME 10 *scheduler* is a service provided by Tivoli that you can use to schedule jobs to be run at a later time or more than once. A *job* is a task that is executed on specific managed resources. When you create a job, you select an existing task to be executed and define the execution information required to run the task. The scheduler does not define jobs, but does allow you to schedule previously created jobs. For example, you can schedule a task library job or a profile distribution.

The following figure shows the icon for the scheduler.

DB2 ECC in the Tivoli Framework

TME 10 is an object-oriented framework that helps you create an abstract model of your working environment. The Tivoli modeling elements, such as policy regions and profile managers, allow you to structure the DB2 management domains in terms of responsibility and to use the DB2 resources by project teams or departments.

In DB2 ECC, the DB2 managed resources can be organized hierarchically. For example, you can have a policy region named Sales that contains the subregions Accounts Receivable and Accounts Payable. The Accounts Receivable policy region might contain two DB2 instances; for example, Northern Area and Southern Area.

Within DB2 ECC policy regions, you can create objects (DB2 managed resources) that represent each of the DB2 resources that you choose to manage. All of the tasks and monitors provided by DB2 ECC are designed to run on the DB2

managed resources. The details concerning things such as the location, the kind of server, and the version of the DB2 managed resource are handled by Tivoli and DB2 ECC.

DB2 ECC, like many other Tivoli applications, is a profile-based application. In a profile-based application, the subscribing endpoints and profile managers are the targets of an action. Each DB2 managed resource belongs to a policy region and can be associated with a profile manager. The profile manager can be subscribed to by many objects, including DB2 managed resources and other profile managers. The profile managers are designed to be a central subscription point. Therefore, each DB2 managed resource that you create in a policy region can be subscribed to an associated profile manager. For example, the Northern Area and Southern Area DB2 instances might both subscribe to the Accounts Payable profile manager.

A task that you run on the DB2 ECC default profile manager can be run on all of the DB2 managed resources defined in your TMR. If you are managing a large number of DB2 databases on workstations scattered throughout your company (for example, in three different countries and seven different time zones) this capability provides you with powerful, centralized control.

Figure 6 on page 11 shows two DB2 databases within a policy region that are subscribed to a profile manager. The profile manager is subscribed to the DB2 ECC default profile manager.
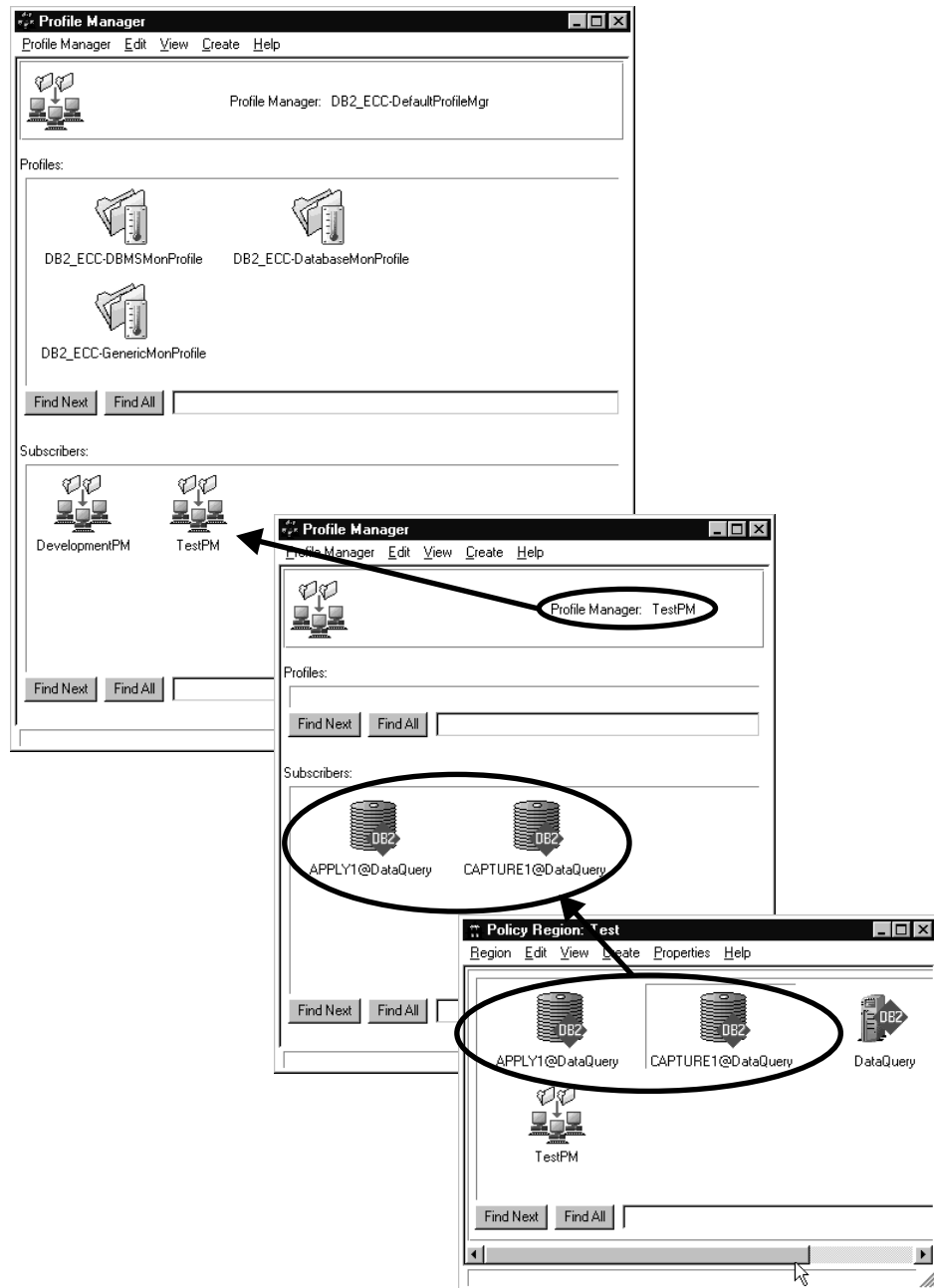
Figure 6. The Relationship between DB2 Databases, Profile Managers, and the DB2 ECC Default Profile Manager

Chapter 2. Installing DB2 ECC

This chapter provides you with information on preparing for a new installation of DB2 ECC V1.2, installing DB2 ECC V1.2 as a new installation or an upgrade of an existing Version 1.1 installation, and installing the online books and reader.

To completely install and set up a new installation of DB2 ECC, you need to:

1. Perform prerequisite tasks, as described in "Preparing for a New DB2 ECC Installation" on page 12.

2. Install DB2 ECC, as described in "Installing DB2 ECC Version 1.2" on page 15.

3. Optional: Perform post-installation tasks, as described in "Post-Installation Tasks" on page 18.

4. Optional: Install the online books and reader, as described in "Installing the Online Books and Reader" on page 19.

5. Set up your DB2 ECC environment, as described in "Chapter 3. Setting Up Your DB2 ECC Environment" on page 23.

To upgrade a system on which DB2 ECC V1.1 is already installed, you need to:

1. Perform prerequisite tasks, as described in "Preparing to Upgrade a Version 1.1 Installation" on page 14.

2. Install DB2 ECC, as described in "Installing DB2 ECC Version 1.2" on page 15.

3. Optional: Perform post-installation tasks, as described in "Post-Installation Tasks" on page 18.

4. Optional: Install the online books, as described in "Installing the Online Books and Reader" on page 19.

Preparing for a New DB2 ECC Installation

Before you can install DB2 ECC, your environment must meet specific hardware and software prerequisites, and you must create a db2ecc operating system user ID. This section explains these prerequisites.

Hardware Requirements

To install DB2 ECC, you must have available disk space for the DB2 ECC files and for object data added to the Tivoli object database during installation. Table 3 lists the available disk space requirements for installing DB2 ECC.

Table 3. Available Disk Space Requirements for DB2 ECC

Software	Required Disk Space
DB2 ECC on an AIX TMR server	15 MB
DB2 ECC on an AIX TMR client	10 MB
DB2 ECC on a Solaris TMR server	18 MB
DB2 ECC on a Solaris TMR client	13 MB
DB2 ECC on a Windows NT TMR server	17 MB
DB2 ECC on a Windows NT TMR client	13 MB

Software Requirements

Before you install the DB2 Enterprise Control Center, ensure that the following software is on your system:

• One of the following Tivoli Management Environment versions:
  – TME 10 Version 3.1.2
  – TME 10 Version 3.2
• One of the following Sentry or Distributed Monitoring versions:

     – Sentry 3.0.2 with patch 3.0.2-SEN-0010

     – Distributed Monitoring Release 3.5 with patch 3.5-SEN-0001

Preparing a db2ecc Operating System (OS) User ID

You must define a db2ecc OS user ID on each node in a TMR where DB2 managed resources will be managed and monitored.

The db2ecc OS user ID is used by:

- The administrative functions of the DB2 managed resource objects
- Tasks that are defined in the DB2 ECC task libraries
- The DB2 ECC monitoring server executable
- Default Sentry profiles (created during installation) to run the default monitors
- User-specified Sentry profiles that contain DB2 ECC monitors

The db2ecc OS user ID must be defined as a login name of the Tivoli root administrator of the management region where you plan to use DB2 ECC. Use the **Edit Logins** pop-up menu to open the Set Login Names window to add db2ecc as a TME 10 administrator login name. For more information on defining the db2ecc OS user ID as a login name, see "Setting Up Existing Administrators for DB2 ECC" on page 27.

The db2ecc OS user ID has specific requirements on each platform.

For AIX or Solaris:

To prepare a db2ecc OS user ID for an AIX or Solaris platform:

1. Define a db2ecc OS user ID on each node in the TMR where DB2 managed resources will be managed.
2. Ensure that the primary group of the db2ecc OS user ID is the SYSADM group of the DB2 instances that you plan to manage. All DB2 instances that you plan to manage across the TMR must be members of the same SYSADM group.
3. If you share executable files between nodes in your TMR using NFS remote mount, ensure that the integer values used to represent the db2ecc OS user ID and its primary group match among all nodes that share executable files in your TMR. If you do not set your environment in this manner, the monitoring system will not work correctly.

For Windows NT:

To prepare a db2ecc OS user ID for a Windows NT platform:

1. Define a db2ecc OS user ID on each node in the TMR where DB2 managed resources will be managed.
2. Ensure that the db2ecc OS user ID is a member of the Administrators and Tivoli_Admin_Privileges groups.
3. Ensure that the db2ecc OS user ID has the ″Log on as service″ advanced user right. Use the User Rights Policy window of the User Manager to grant this right.

You have now completed the prerequisite tasks for a new installation of DB2 ECC. To continue with the installation process, go to "Installing DB2 ECC Version 1.2" on page 15 .

Preparing to Upgrade a Version 1.1 Installation

Before you upgrade an existing DB2 ECC Version 1.1 installation, your environment must meet specific hardware and software prerequisites, you must stop the monitoring agent on each of the nodes to be upgraded, and you might want to save any jobs that you created in the DB2 ECC task libraries. This section explains these prerequisites, how to stop the monitoring agent, and how to save jobs in the DB2 ECC task libraries.

Hardware Requirements

To upgrade your DB2 ECC Version 1.1 installation to Version 1.2, you must have additional disk space. Table 4 lists the additional disk space requirements for upgrading DB2 ECC.

Table 4. Additional Disk Space Requirements for DB2 ECC V1.2

Software	Additional Required Disk Space
DB2 ECC on an AIX TMR server	6 MB
DB2 ECC on an AIX TMR client	6 MB
DB2 ECC on a Windows NT TMR server	7 MB
DB2 ECC on a Windows NT TMR client	5 MB

Software Requirements

Before you upgrade the DB2 Enterprise Control Center, ensure that the following software is on your system:

- One of the following Tivoli Management Environment versions:
  - TME 10 Version 3.1.2
  - TME 10 Version 3.2

  Note that support for TME 10 Version 3.2 has been added and that TME 10 Version 3.1.2 replaces TME 10 Version 3.1 with fix patch 0002.
- One of the following Sentry or Distributed Monitoring versions:
  - Sentry 3.0.2 with patch 3.0.2-SEN-0010
  - Distributed Monitoring Release 3.5 with patch 3.5-SEN-0001

  Note that support for Distributed Monitoring Release 3.5 has been added.

Stopping the Monitoring Agent

Before you upgrade your DB2 ECC Version 1.1 installation to Version 1.2, you must stop the monitoring agent on all nodes to be upgraded. To stop the monitoring agent, run the ECC_Stop_Monitoring_Agent task in the DB2_ECC-AdminTasks task library on each managed node to be upgraded.

After the upgrade is completed successfully, run the ECC_Start_Monitoring_Agent task in the DB2_ECC-AdminTasks task library to start the monitoring agent on each of the managed nodes.

Attention: The results returned by any monitor while DB2 ECC is being upgraded should be disregarded.

Saving Jobs in the DB2 ECCTask Libraries

When you upgrade your DB2 ECC Version 1.1 installation to Version 1.2, the DB2 ECC task libraries are completely replaced. Any jobs that you created in the Version 1.1 DB2 ECC task libraries will be lost.

If you want to restore the Version 1.1 jobs, use the **wgetjob** command to list the information necessary to create the jobs before you upgrade DB2 ECC. After you upgrade DB2 ECC, use the TME 10 desktop or the **wcrtjob** command to re-create the jobs. See the *TME 10 Framework Reference Manual* for more information about the **wgetjob** and **wcrtjob** commands.

Installing DB2 ECC Version 1.2

The installation instructions assume that you have TME installed and running; if you do not, see the *TME 10 Framework Planning and Installation Guide*. With Tivoli, you can install DB2 ECC onto any node in a TMR from any other node in the TMR.

DB2 ECC must be installed on the TME 10 server node. When you install DB2 ECC, Tivoli copies binary files and libraries to each node that you select, and it copies database updates to the TME 10 server node. If you share executable files through mounting a shared resource, Tivoli installs only nonshared files as required for each node.

Important: You should back up the Tivoli database before you install DB2 ECC. See the *TME 10 Framework Planning and Installation Guide* for information on backing up the Tivoli database.

To install the DB2 Enterprise Control Center:
1. Start Tivoli, as described in the *TME 10 Framework User's Guide*.
2. If you are upgrading DB2 ECC, remove any jobs you currently have scheduled using the TME 10 desktop or the **wdelsched** command.
3. Select the **Desktop→Install→Install Product** menu choice. The Install Product window opens (Figure 7 on page 16).

Installing DB2 ECC Version 1.2
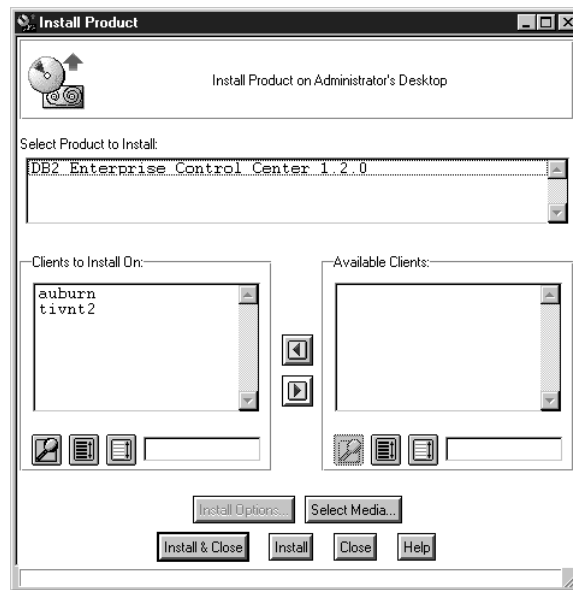
Figure 7. The Install Product Window

If you previously installed a product but the media setting for the installation facility does not point to a directory containing a CONTENTS.LST file, a warning message is displayed. See "Installation Problems" on page 85 for additional information.

4. Ensure that **DB2 Enterprise Control Center 1.2.0** is displayed in the **Select Product to Install** box.

   • If the **Select Product to Install** box contains **DB2 Enterprise Control Center 1.2.0**, go to step 5.

   • If the **Select Product to Install** box does not contain **DB2 Enterprise Control Center 1.2.0**, specify a new path for the DB2 ECC installation package:

      a. Click on the **Select Media** push button. The File Browser window opens.

      b. Select the host on which DB2 ECC installation package is located from the **Hosts** box and type the full path in the **Path Name** field.

      c. Set the new path by clicking on the **Set Path** push button.

      d. Save the new media path by clicking on the **Set Media & Close** push button. The Install Product window reopens, and **DB2 Enterprise Control Center 1.2.0** is displayed in the **Select Product to Install** box.

5. From the **Select Product to Install** box, select **DB2 Enterprise Control Center 1.2.0**. The Install Options window opens (Figure 8 on page 17).
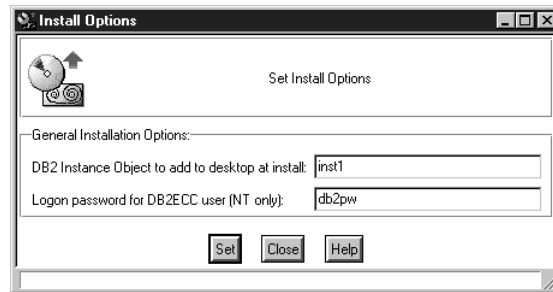
Figure 8. The Install Options Window

6. Optional: If you want to create a DB2 instance object, type a DB2 instance name in the **DB2 Instance object to add to desktop at install** field. For example:

   ```
   inst1
   ```

   The Install Options window lets you enter one DB2 instance name to create a DB2 instance object on the TME 10 server node. This object is added to the DB2 ECC default policy region as part of the installation process.

   If you want to create additional DB2 instances, you can do so after installation from the **Create→DB2Instance** menu option in any policy region that has the DB2 instance managed resource type.

7. For Windows NT: Type the db2ecc password into the **Logon password for DB2ECC user (NT only)** field.

8. Click on the **Set** push button. The installation options information is saved, the window closes, and the Install Product window reopens.

9. Select the clients on which you want to install DB2 ECC by using the ▣ and

   ▣ push buttons. See Figure 7 on page 16.

   All of the available clients in the TMR appear in the **Clients to Install On** box. Remove all of the client names that you do not want to install DB2 ECC on by selecting them and clicking on the ▣ push button. The client names move to

   the **Available Clients** box.

   Reminder: Be sure to select the TME 10 server node as a client for the installation.

   If the DB2 instance object that you want to add or the logon password vary for different sets of clients, you need to install each client-set separately and update the Install Options window each time for each client-set that shares common installation options.

10. Click on the **Install & Close** push button.

    The Install Product window closes, and the Product Install window opens.

    Tivoli performs checks for prerequisites, reports any possible problems, and lists the operations that will be performed during installation.

    Attention: If an error occurs during installation, after you correct the error you can restart the installation by closing the Product Install window and clicking on the **Install** push button again.

11. Click on the **Continue Install** push button.

Installing DB2 ECC Version 1.2

The installation begins, and the Product Install window lists information as the installation proceeds. When the installation is complete, the following message is displayed in the Product Install window: `Finished product installation.` Figure 9 shows the Product Install Window with installation information listed.
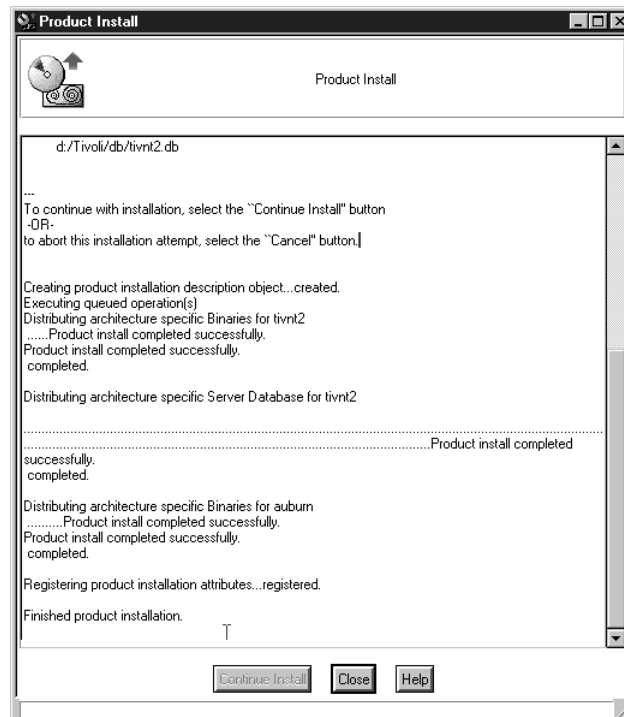
Figure 9. The Product Install Window, Finished Product Installation

12. Click on the **Close** push button.

To see the DB2 ECC objects that were created during installation:

1. From the Desktop Navigator, select **PolicyRegion** from the **Navigate To** box. Available policy regions are displayed in the **Resources** box.

2. Select **DB2_ECC-DefaultPolicyRegion** from the **Resources** box and click on the **Go To** push button. The DB2_ECC-DefaultPolicyRegion window opens. The DB2_ECC-DefaultPolicyRegion window shows the DB2 ECC objects that were created during installation.

Post-Installation Tasks

After the installation is complete, you might need to do some additional tailoring of the TME 10 server or managed nodes if you plan to have one of the following configurations:

- A TME 10 server machine without any DB2 managed resources on it. See "Tailoring the TME 10 Server" on page 19.

- Managed nodes that share DB2 ECC executable files. See "Setting up a Shared Executable Configuration" on page 19.

If you don't plan to have either of these configurations, skip to "Installing the Online Books and Reader".

Tailoring the TME 10 Server

DB2 ECC installs processes that are automatically started by the operating systems. If you don't plan to manage any DB2 managed resources on the TME 10 server machine, you can stop the operating system from automatically starting these processes.

To stop the monitoring processes from starting:
1. Run the ECC_Set_Up_Monitoring_Agent in the DB2_ECC-AdminTasks task library.
2. Select the TME 10 server node and select **Remove and stop** for the action on the task argument window.
3. For Windows NT only: Provide the user ID and password.

Attention: Do not remove the DB2 ECC binary files from the TME 10 server.

Setting up a Shared Executable Configuration

You can use NFS remote mount to share executable files between managed nodes. If you do share executable files through mounting a shared resource, you must set up and start the monitoring agent on the managed nodes that share executable files.

To set up and start the monitoring agent on the managed nodes, run the ECC_ Set_Up_Monitoring_Agent task on each managed node that is sharing the executable files. Select the **Add and start** action on the task argument window.

On AIX and Solaris managed nodes, this task inserts a db2ecc entry into /etc/inittab and starts the monitoring daemon on the managed node. On Windows NT managed nodes, this task installs and starts the DB2 ECC monitoring service on the managed nodes.

Installing the Online Books and Reader

Two DB2 ECC online books are provided on the DB2 ECC CD-ROM with the Adobe Acrobat Reader, which you can use to view the books. Table 5 shows the files and directories for the online books and the Adobe Acrobat Reader.

Table 5. DB2 ECC Online Book Files and Adobe Acrobat Reader Files

Title	Directory and File Name
DB2 Enterprise Control Center for TME 10 V1.2 User's Guide	DOCS\ECCU1200.PDF
DB2 Enterprise Control Center for TME 10 V1.2 Monitoring Collection and Task Library Reference	DOCS\ECCM1200.PDF
Adobe Acrobat Reader for AIX	Acrobat/aix4-r1/acroread_aix_301_tar.Z
Adobe Acrobat Reader for Solaris	Acrobat/solaris2/acroread_solaris_301_tar.Z
Adobe Acrobat Reader for Windows NT	Acrobat\w32-ix86\ar32e301.exe

Installing the Online Books and Reader

You can also download the Adobe Acrobat Reader from the following site:

http://www.adobe.com/prodindex/acrobat/readstep.html

For AIX:

To install the online books and reader:

1. Copy ECCU1200.PDF and ECCM1200.PDF from the DOCS file path on the DB2 ECC CD-ROM to a directory on your workstation.
2. Copy acroread_aix_301_tar.Z from the Acrobat/aix4-r1/ file path on the CD-ROM to a directory on your workstation.
3. Uncompress the acroread_aix_301_tar.Z file by entering the following command:
   ```
   uncompress acroread_aix_301_tar.Z
   ```
4. Untar acroread_aix_301_tar by entering the following command:
   ```
   tar -xvf acroread_aix_301_tar
   ```
5. Read the INSTGUID.TXT file for detailed installation steps.
6. Start the install script by entering the following command:
   ```
   ./INSTALL
   ```
7. Follow the instructions that appear on the screen. (The installation application asks you to read and accept the Electronic End-User License Agreement.)

For Solaris:

To install the online books and reader:

1. Copy ECCU1200.PDF and ECCM1200.PDF from the DOCS file path on the DB2 ECC CD-ROM to a directory on your workstation.
2. Copy acroread_solaris_301_tar.Z from the Acrobat/solaris2/ file path on the CD-ROM to a directory on your workstation.
3. Uncompress the acroread_solaris_301_tar.Z file by entering the following command:
   ```
   uncompress acroread_solaris_301_tar.Z
   ```
4. Untar acroread_solaris_301_tar by entering the following command:
   ```
   tar -xvf acroread_solaris_301_tar
   ```
5. Read the INSTGUID.TXT file for detailed installation steps.
6. Start the install script by entering the following command:
   ```
   ./INSTALL
   ```
7. Follow the instructions that appear on the screen. (The installation application asks you to read and accept the Electronic End-User License Agreement.)

For Windows NT:

To install the online books and reader:

1. Copy ECCU1200.PDF and ECCM1200.PDF from the DOCS file directory on the DB2 ECC CD-ROM to a directory on your computer.
2. Copy ar32e301.exe from the Acrobat\w32-ix86\ file path on the CD-ROM to a directory on your computer.
3. Double-click on the ar32e301.exe file.
4. Follow the instructions on your screen. (The installation application asks you to read and accept the Electronic End-User License Agreement.)

Attention: If there is a failure during the installation of Acrobat Reader 3.01, the installer performs a complete uninstallation. After you close the ″Thank You″ window that appears after the installation completes, do not close the installer application. After a few moments, the installer automatically closes the window.

Installing the Online Books and Reader

Chapter 3. Setting Up Your DB2 ECC Environment

After you install DB2 ECC, you need to set up your DB2 ECC environment so that you can use DB2 ECC to manage your DB2 instances and databases. Before you complete the tasks in this chapter, be sure that you understand Tivoli concepts. See the *TME 10 Framework Planning and Installation Guide* for information on how to plan for and implement Tivoli modeling elements in your enterprise.

After you conceptually plan your environment, you are ready to start setting up your environment using the Tivoli framework. This chapter explains the following tasks that you need to perform to set up your environment:

- Establishing administrators for DB2 ECC
- Creating policy regions
- Moving DB2 ECC objects created during installation

After you complete these tasks, you will be ready to start managing your DB2 instances and databases by performing DB2 ECC tasks, such as adding monitors and scheduling jobs.

Establishing Administrators for DB2 ECC

You can begin setting up your DB2 ECC environment by establishing system administrators to manage resources and perform tasks.

The following sections:

- Explain how to establish new TME 10 users as DB2 ECC administrators.
- Explain how to establish existing TME 10 administrators as DB2 ECC administrators, and how to adjust any DB2 ECC administrator's identifiers, authorizations, and setup for receiving messages.

Establishing First-Time Administrators for DB2 ECC

To establish first-time Tivoli users as administrators for DB2 ECC:

1. Open the Create Administrator window:
   - If the **Administrators** icon is on your desktop, press and hold the right mouse button on this icon and select **Create Administrator** from the pop-up menu. The window opens (Figure 10 on page 24).

Figure 10. The Create Administrator Window

- If the **Administrators** icon isn't on your desktop:
  a. Select **Desktop→Navigator**. The Navigator window opens.
  b. Select the **AdministratorCollection** check box.
  c. In the **Resources** box, select **Administrators**. The Administrators window opens.
  d. On the Administrators window, select **Create→Administrator**. The Create Administrator window opens.

2. In the Create Administrator window, define the administrator to DB2 ECC:
  a. In the **Administrator Name/Icon Label** field, type the administrator's name as you want it displayed beside the administrator's icon.
  b. In the **User Login Name** field, type the administrator's user account name for the platform (AIX, Solaris, or Windows NT) that the administrator will be using.
  c. In the **Group Name** field, type the group account name for the platform (AIX, Solaris, or Windows NT) that the administrator will be using.

3. If you want to authorize the administrator to perform TMR-wide operations:
  a. Click on **Set TMR Roles**. The Set TMR Roles window opens (Figure 11 on page 25).
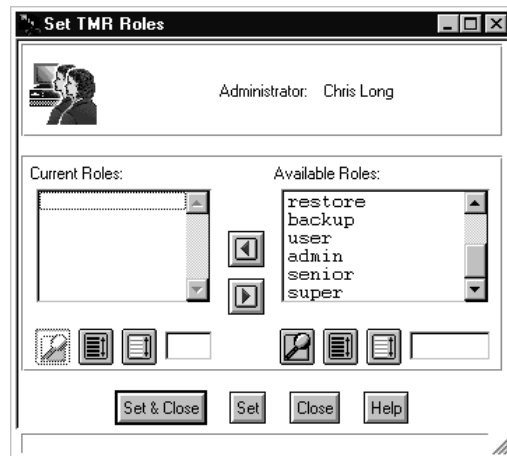
Figure 11. The Set TMR Roles Window

      b. In the **Available Roles** box, select one or more roles for the administrator and then click on the ◁ push button. The selected roles move to the

          Current Roles box.

      c. Click on the **Set & Close** push button.

4. If you want to authorize the administrator to manage individual resources:

      a. In the Create Administrator window, click on **Set Resource Roles**. The Set Resource Roles window opens (Figure 12).
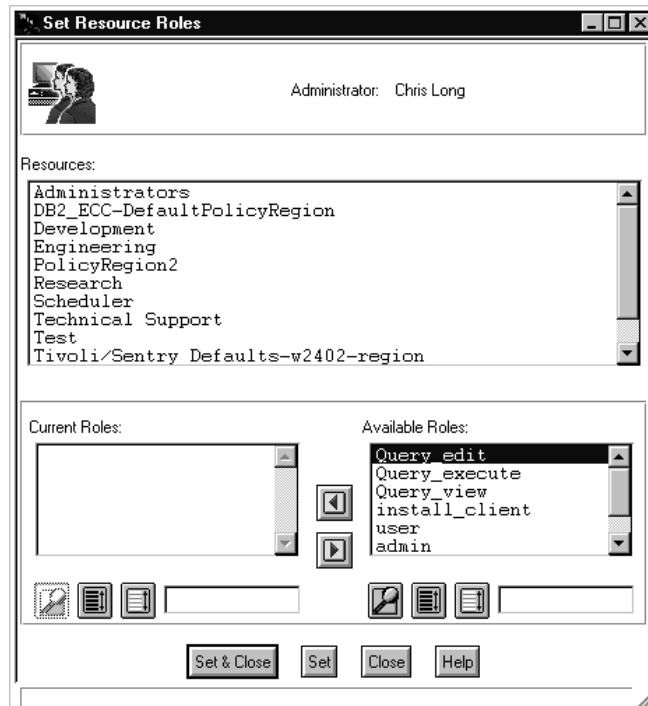
Figure 12. The Set Resource Roles Window

    b. In the **Resources** box, select a resource that you want the administrator to manage.

    c. In the **Available Roles** box, select one or more roles that apply to the selected resource and then click on the ⊡ push button to move the

       selected roles to the **Current Roles** box.

    d. Repeat steps b and c for each additional resource that you want the administrator to manage.

    e. Click on the **Set & Close** push button.

5. Set the ID under which you want the administrator to open the desktop and run commands.

    a. In the Create Administrator window, click on **Set Logins**. The Set Login Names window opens (Figure 13).

Figure 13. The Set Login Names Window

    b. Supply the ID:

- If you want the administrator to open the desktop from any machine in your TMR:

    1) In the **Add Login Name** field, type an unqualified ID; for example:

        `chris`

    2) Press Enter to move the ID to the **Current Login Names** box.

- If you want the administrator to open the desktop from one or more specific machines:

    1) In the **Add Login Name** field, type an ID followed by @ and the name of the machine on which you want the ID to work; for example:

        `chris@tundra`

    2) Press Enter to add the ID to the **Current Login Names** box.

    3) Repeat steps 1) and 2) for each additional machine on which you want the desktop to be available.

    c. Click on the **Set & Close** push button.

6. Subscribe the administrator to one or more notice groups.

a. In the Create Administrator window, click on **Set Notice Groups**. The Set Notice Groups window opens (Figure 14).
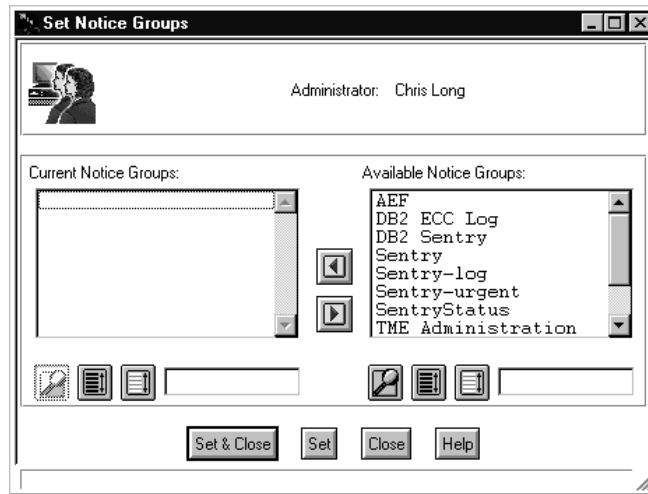
Figure 14. The Set Notice Groups Window

b. In the **Available Notice Groups** box, select the notice groups that you want the administrator to subscribe to and then click on the ▣ push button. The

   selected groups move to the **Current Notice Groups** box.

   Be sure to subscribe the administrator to DB2 ECC Log, so that the administrator can receive informational and error messages about DB2 ECC operations. If you want the administrator to use any of the Sentry profiles supplied by DB2 ECC as defaults, also subscribe the administrator to DB2 Sentry, which provides DB2 ECC monitoring source output. Additional notice groups for the monitoring environment are available; for example, SentryStatus, which reports monitor execution errors.

c. Click on the **Set & Close** push button.

7. On the Create Administrator window, click on **Create and Close**.

8. Drag and drop the icons for the managed resources that the administrator is going to manage onto the icon for the administrator.

Setting Up Existing Administrators for DB2 ECC

To set up existing administrators (that is, users who are already administrators in your TMR when you install DB2 ECC) as administrators for DB2 ECC, use windows discussed in "Establishing First-Time Administrators for DB2 ECC" on page 23. For example, to allow an administrator to create and delete DB2 instances throughout the TMR, use the Set TMR Roles window (Figure 11 on page 25). Or to change the ID under which the administrator opens the desktop, use the Set Login Names window (Figure 13 on page 26).

To set up subscriptions, use the Set Notice Groups window (Figure 14). Be sure to subscribe the administrator to the DB2 ECC Log notice group. If you want the administrator to use the Sentry profiles that DB2 ECC provides as defaults, also subscribe the administrator to the DB2 Sentry notice group.

Creating Policy Regions

A *policy region* is a Tivoli term for a collection of resources that share one or more common policies. To use DB2 ECC to manage your DB2 instances and databases, you need to create policy regions.

You can establish a policy hierarchy that reflects the organization of your company; for example, with top-level policy regions named Sales, Marketing, and Manufacturing. Each of these policy regions might contain subregions to further represent the organization; for example, Accounts Receivable and Accounts Payable under Sales. Different administrators might have different roles over these regions so that they can manage appropriate sets of DB2 instances and databases.

You can create a policy region from the TME 10 desktop or a command line.

Desktop

To create a policy region from the desktop:

1. From the desktop, select **Create→Region**. The Create Policy Region window opens (Figure 15).

Figure 15. The Create Policy Region Window

2. In the **Name** field, type a name for the new policy region. The policy region name can include letters, underscores, dashes, periods, and spaces, and must be unique within the local TMR.
3. Click on the **Create & Close** push button to create the new policy region and return to the desktop.

Command Line

Use the **wcrtpr** command to create a policy region from the command line. The syntax for the command might vary depending on the version of the TME framework that you are using. For more information, see the *TME 10 Framework Reference Manual*.

Example:

To create a policy region named Europe_Operations on administrator Chris' desktop and make DB2Instance and DB2Database managed resource types of the new policy region, enter the following command:

```
wcrtpr -a chris -m DB2Instance -m DB2Database Europe_Operations
```

Moving DB2 ECC Objects Created during Installation

DB2 ECC objects created during installation are added to the DB2 ECC default policy region. For easier accessibility to the DB2 ECC objects, you might want to move them to another policy region. Moving objects to another policy region would allow you to directly access the DB2 ECC objects from your desktop instead of having to go through the **Navigator** pull-down. For example, if you want the DB2 database objects created through database discovery to be created in a particular policy region, move the DB2 instance object created during installation to that region before you run the database discovery task. You can move DB2 instance objects, DB2 database objects, DB2 partition objects, and DB2 partition group objects between policy regions and copy them to administrator desktops.

An object is accessible as a subscriber or an endpoint in a policy region regardless of whether that object resides in the policy region. The resource definition belongs to the TMR, not to a specific policy region.

You can move a DB2 ECC object created during installation to another policy region from the TME 10 desktop or a command line.

Desktop, Drag and Drop

To move a DB2 ECC object created during installation to another policy region:

1. From the desktop, select **Desktop**⇒**Navigator**. The Navigator window opens.
2. From the Navigator window, select the **PolicyRegion** check box. The policy regions that you can use are listed in the **Resources** box.
3. Select **DB2_ECC-DefaultPolicyRegion** and click on the **Go To** push button. The policy region window for the DB2 ECC default policy region opens (Figure 16).
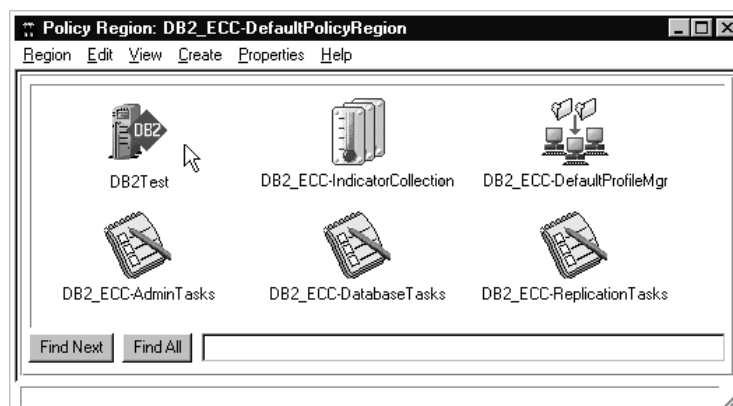
Figure 16. The Policy Region Window for the DB2 ECC Default Policy Region

4. Select the icon of the DB2 ECC object that you want to move.
5. Press and hold the Shift key.
6. Press and hold the left mouse button and drag the icon into a policy region or over a policy region icon.
7. Release the left mouse button and the Shift key. The DB2 ECC object is moved from the original policy region to the destination policy region.

Command Line

Use the **wmv** command to move DB2 ECC objects between collections from the command line. The syntax for the command might vary depending on the version of the TME framework that you are using. For more information, see the *TME 10 Framework Reference Manual*.

Examples:

To move a DB2 instance called DB2_inst1 from the DB2 ECC default policy region to the Sales policy region, enter the following command:

```
wmv /Regions/DB2_ECC-DefaultPolicyRegion/DB2_inst1 /Regions/Sales
```

To move the task library called DB2_ECC-DatabaseTasks to a policy region called Manufacturing, enter the following command:

```
wmv /Library/TaskLibrary/DB2_ECC-DatabaseTasks /Regions/Manufacturing
```

Chapter 4. Working with DB2 Managed Resources

A *managed resource* is a Tivoli term for a system or network resource that you manage with TME 10. DB2 ECC provides four types of managed resources: DB2 instances, DB2 databases, DB2 partitions, and DB2 partition groups.

This chapter provides steps for creating and performing operations on DB2 managed resources. You must first create objects for the resources that you want to manage before you can perform any of the other tasks on the DB2 managed resources. The tasks that you can perform include:

- Creating DB2 instance objects
- Creating DB2 database objects
- Creating DB2 partition objects
- Creating DB2 partition group objects
- Changing members of a partition group
- Viewing DB2 object properties
- Starting and stopping DB2 instances, or nodes for partitions or partition groups
- Running DB2 commands from the DB2 command line processor
- Launching the DB2 Control Center
- Migrating DB2 ECC objects to DB2 UDB

You can also open a window for a DB2 managed resource object that shows the relevant contents of the object by selecting **Open** from the object's pop-up menu.

Creating DB2 Instance Objects in a Policy Region

You can create a DB2 instance object in a policy region using the TME 10 desktop or a command line. A DB2 instance object can represent either a partitioned or nonpartitioned database server. For a partitioned database server, all partitions belonging to the instance must reside in the same TMR. You need only one DB2 instance object to represent a partitioned database server even though the instance resides on multiple managed nodes.

Attention: If you drop and re-create a DB2 instance, you should delete the corresponding DB2 ECC instance object and then re-create it. Before deleting a DB2 instance object, you must unsubscribe the object from all profile managers to which it is subscribed.

Desktop

To create a DB2 instance object from the desktop:

1. From the desktop, double-click on the policy region to which you want to add a DB2 instance object. The Policy Region window opens.
2. Ensure that **DB2Instance** is displayed in the **Create** menu. Figure 17 on page 32 shows a **Create** menu that contains **DB2Instance**.

Creating DB2 Instance Objects
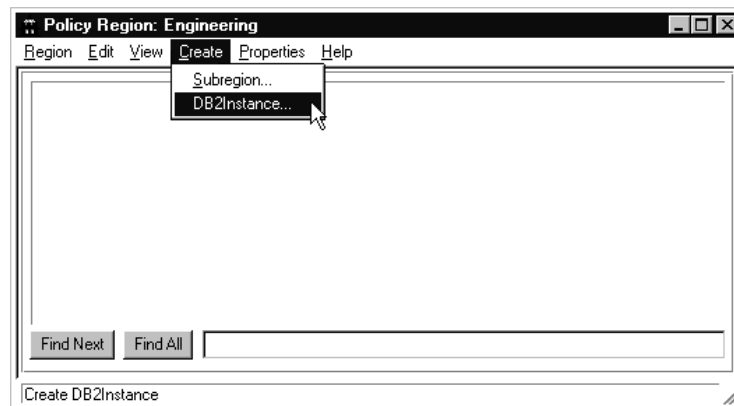
Figure 17. A Policy Region Window with Create→DB2Instance Selected

- If **DB2Instance** is displayed in the **Create** menu, go to step 3.
- If **DB2Instance** is not displayed in the **Create** menu:
  - a. From the Policy Region window, select **Properties→Managed Resources**. The Set Managed Resources window opens (Figure 18).
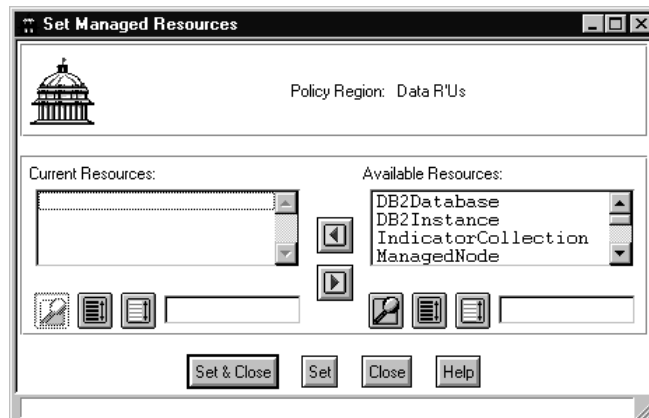
Figure 18. The Set Managed Resources Window

- b. In the **Available Resources** box, select **DB2Instance** and click on the ⬜ push button. **DB2Instance** moves to the **Current Resources** box.
- c. Click on the **Set & Close** push button.
3. From the Policy Region window, select **Create→DB2Instance**. The Create DB2Instance window opens (Figure 19 on page 33).

Figure 19. The Create DB2Instance Window

4. From the **Managed Node** box, select the host name of the machine on which the DB2 instance exists.
5. In the **Instance Name** field, type the name of the DB2 instance on the machine selected in the **Managed Node** box.
6. In the **Label** field, type a unique label for the icon for the DB2 instance specified in the **Instance Name** field.
7. Click on the **Create & Close** push button to create the DB2 instance object.

Command Line

Use the following command to create a DB2 instance object from a command line:

```
wcrdb2inst -l instance_label -n managed_node -i instance -p policy_region
```

where:

-l *instance_label*
> Specifies a unique label for the DB2 instance object.

-n *managed_node*
> Specifies the host name of the machine containing the DB2 instance.

-i *instance*
> Specifies the name of the existing instance on the specified managed node for which you want to create an endpoint.

-p *policy_region*
> Specifies the name of the policy region to which the DB2 instance object will belong.

Example:

To create a DB2 instance object for the instance db2inst1 labeled db2inst1@catalina, on the managed node catalina in the policy region westcoast, enter the following command:

```
wcrdb2inst -l db2inst1@catalina -n catalina -i db2inst1 -p westcoast
```

Creating DB2 Database Objects in a Policy Region

You can create a DB2 database object in a policy region using the TME 10 desktop or a command line. You can also create DB2 database objects using the Discover Databases function. See "Discovering Databases from a DB2 Instance Object" on page 36 for more information.

You must create a DB2 instance object before creating a DB2 database object for a DB2 database on that instance. The DB2 database must be a single-partition database.

Attention: Before deleting a DB2 database object, you must unsubscribe the object from all profile managers to which it is subscribed.

Desktop

To create a DB2 database object from the desktop:

1. From the desktop, double-click on the policy region to which you want to add a DB2 database object. The Policy Region window opens.

2. Ensure that **DB2Database** is displayed in the **Create** menu. Figure 20 shows a **Create** menu that contains **DB2Database**.
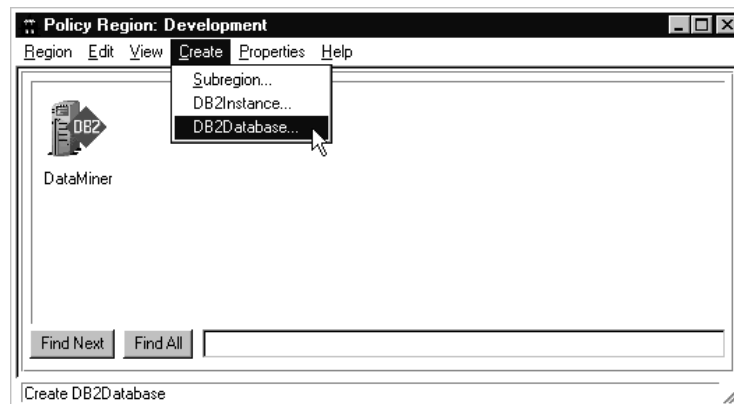
Figure 20. A Policy Region Window with Create⇥DB2Database Selected

- If **DB2Database** is displayed in the **Create** menu, go to step 3.

- If **DB2Database** is not displayed in the **Create** menu:
  a. From the Policy Region window, select **Properties⇥Managed Resources**. The Set Managed Resources window opens (Figure 18 on page 32).

  b. In the **Available Resources** box, select **DB2Database** and click on the ⬅ push button. **DB2Database** moves to the **Current Resources** box.

  c. Click on the **Set & Close** push button.

3. From the Policy Region window, select **Create⇥DB2Database**. The Create DB2Database window opens (Figure 21 on page 35).

Figure 21. The Create DB2Database Window

4. From the **DB2 Instance** box, select the instance name in which the DB2 database exists.
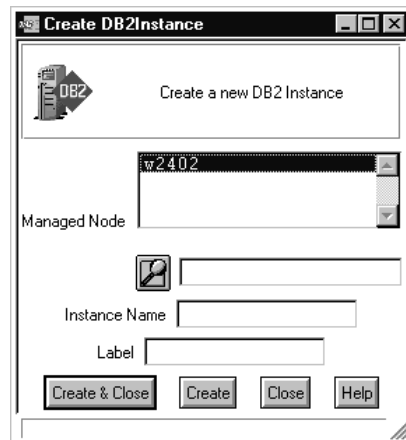5. In the **Database Name** field, type the name of the DB2 database that exists on the instance selected in the **DB2 Instance** box or click on the **Discover** push button to have DB2 ECC help you select an available database name.
6. In the **Label** field, type a unique label for the icon for the DB2 database specified in the **Database Name** field. If you clicked on the **Discover** push button to have DB2 ECC help you select an available database name, the **Label** field is filled in for you.
7. Click on the **Create & Close** push button to create the DB2 database object.

Command Line

Use the following command to create a DB2 database object from a command line:

```
wcrdb2db -l database_label -i instance -d database -p policy_region
```

where:

-l *database_label*
> Specifies a unique label for the DB2 database object.

-i *instance*
> Specifies the name of the existing instance label that contains the DB2 database.

-d *database*
> Specifies the name of the existing database on the specified instance for which you want to create an endpoint.

-p *policy_region*
> Specifies the name of the policy region to which the DB2 database object will belong.

Example:

To create a DB2 database object for a database sample labeled sample@db2inst1, on the instance db2inst1@catalina in the policy region westcoast, enter the following command:

```
wcrdb2db -l sample@db2inst1 -i db2inst1@catalina -d sample -p westcoast
```

Discovering Databases from a DB2 Instance Object

DB2 ECC provides an automated way of creating DB2 database objects in a policy region. This function, called Discover Databases, is available from any nonpartitioned DB2 instance pop-up menu.

The Discover Databases function lists local databases only. For a remote DB2 database object, use the steps described in "Creating DB2 Database Objects in a Policy Region" on page 34.

To discover databases on a DB2 instance:

1. From the desktop, double-click on the policy region that contains the DB2 instance object whose databases you want to discover. The Policy Region window opens.
2. Select **Discover Databases** from the pop-up menu for the DB2 instance object whose databases you want to discover, as shown in Figure 22. The Discover Databases window opens (Figure 23).
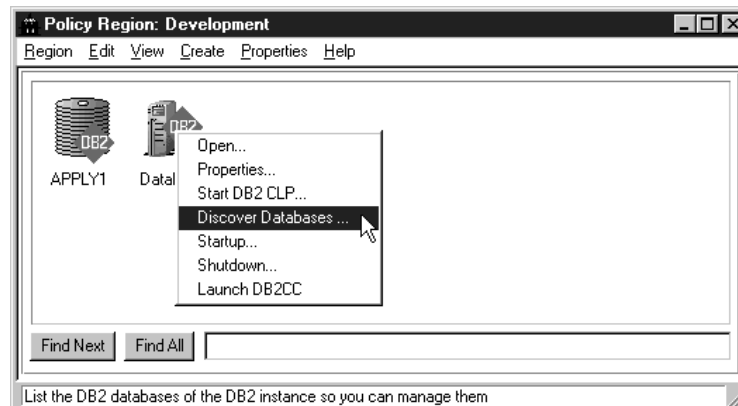
Figure 22. The Policy Region Window with the DB2 Instance Pop-Up Menu
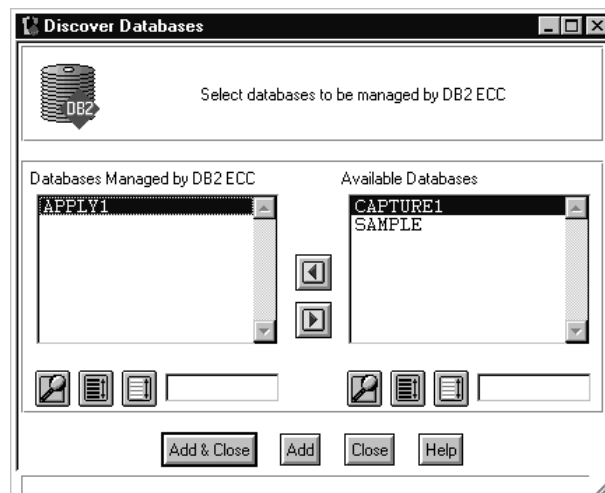
Figure 23. The Discover Databases Window

3. Select a database from the **Available Databases** box and click on the ▣ push

   button to move the database name to the **Databases Managed by DB2 ECC**

   box.

4. Click on the **Add & Close** push button to add the DB2 database object to the
   policy region.

Creating DB2 Partition Objects in a Policy Region

You can create a DB2 partition object in a policy region using the TME 10 desktop
or a command line. There must be a partitioned database server (DB2 instance)
object in your TMR before you can create a DB2 partition object. All partitions
belonging to a partitioned database server must reside in the same TMR.

Attention: Before deleting a DB2 partition object, you must unsubscribe the object
from all profile managers to which it is subscribed.

Desktop

To create a DB2 partition object from the desktop:

1. From the desktop, double-click on the policy region to which you want to add a
   DB2 partition object. The Policy Region window opens.

2. Ensure that **DB2Partition** is displayed in the **Create** menu. Figure 24 shows a
   **Create** menu that contains **DB2Partition**.

Figure 24. A Policy Region Window with Create➔DB2Partition Selected

- If **DB2Partition** is displayed in the **Create** menu, go to step 3 on page 38.

- If **DB2Partition** is not displayed in the **Create** menu:

  a. From the Policy Region window, select **Properties➔Managed Resources**.
     The Set Managed Resources window opens (Figure 25 on page 38).

Creating DB2 Partition Objects

Figure 25. The Set Managed Resources Window

      b. In the **Available Resources** box, select **DB2Partition** and click on the ◁ push button. **DB2Partition** moves to the **Current Resources** box.

      c. Click on the **Set & Close** push button.

3. From the Policy Region window, select **Create→DB2Partition**. The Create DB2Partition window opens (Figure 26).

Figure 26. The Create DB2Partition Window

4. From the **DB2 Instance** box, select the name of the partitioned database server (DB2 instance) from which you want to create a DB2 partition object. The node number, host name, and logical port number of the available partitions for the selected DB2 instance are displayed in the **Node Number** box.

5. From the **Node Number** box, select the node that contains the database for which you want to create a DB2 partition object.

6. In the **Database Name** field, type the name of the DB2 database that exists on the node selected in the **Node Number** box or click on the **Discover** push button to have DB2 ECC help you select an available database name.

7. In the **Label** field, type a unique label for the icon for the DB2 partition selected in the **Node Number** box.

8. Click on the **Create & Close** push button to create the DB2 partition object.

Command Line

Use the following command to create a DB2 partition object from a command line:

```
wcrdb2part -l partition_label -i instance_label -d database -n node_number
  -p policy_region
```

where:

-l *partition_label*
    Specifies a unique label for the DB2 partition object.

-i *instance_label*
    Specifies the label of the instance containing the DB2 partition.

-d *database*
    Specifies the name of the existing partitioned database in the specified instance for which you want to create an object.

-n *node_number*
    Specifies the node number of the existing partitioned database in the specified instance for which you want to create an object.

-p *policy_region*
    Specifies the name of the policy region to which the DB2 partition object will belong.

Example:

To create a DB2 partition object for a partitioned database sample, with the node number 0, in the instance labeled db2pel@catalina in the policy region westcoast, enter the following command:

```
wcrdb2part -l sample_0@db2pel -i db2pel@catalina -d sample -n 0 -p westcoast
```

Creating DB2 Partition Group Objects in a Policy Region

A *DB2 partition group* is a DB2 ECC object that represents a collection of DB2 partitions. It does not represent a DB2 object, and has no relationship to a DB2 node group.

You can create a DB2 partition group object in a policy region using the TME 10 desktop or a command line. There must be a partitioned database server (DB2 instance) object in your TMR before you can create a DB2 partition group object. You should create DB2 partition objects before creating a DB2 partition group object so the DB2 partition group object will not be empty.

Attention: Before deleting a DB2 partition group object, you must unsubscribe the object from all profile managers to which it is subscribed.

Desktop

To create a DB2 partition group object from the desktop:

1. From the desktop, double-click on the policy region to which you want to add a DB2 partition group object. The Policy Region window opens.

Creating DB2 Partition Group Objects

2. Ensure that **DB2PartitionGroup** is displayed in the **Create** menu. Figure 27 shows a **Create** menu that contains **DB2PartitionGroup**.
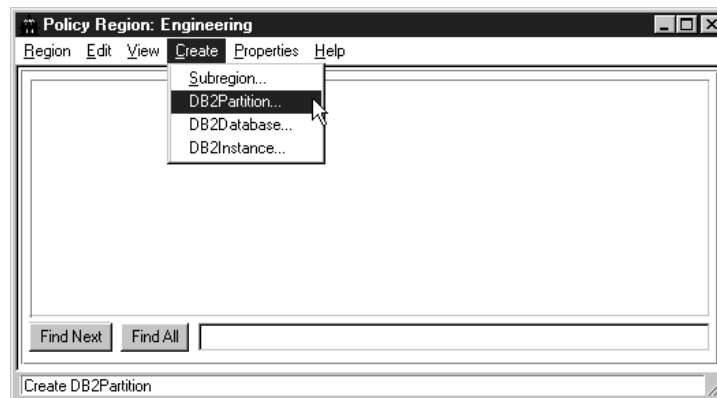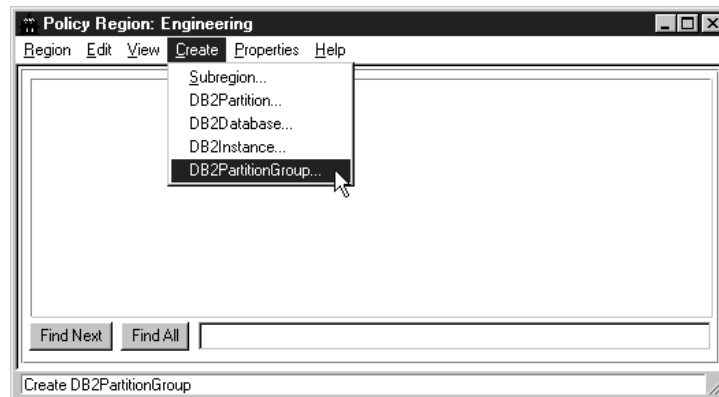

Figure 27. A Policy Region Window with Create→DB2PartitionGroup Selected

- If **DB2PartitionGroup** is displayed in the **Create** menu, go to step 3.
- If **DB2PartitionGroup** is not displayed in the **Create** menu:
  a. From the Policy Region window, select **Properties→Managed Resources**. The Set Managed Resources window opens (Figure 28).
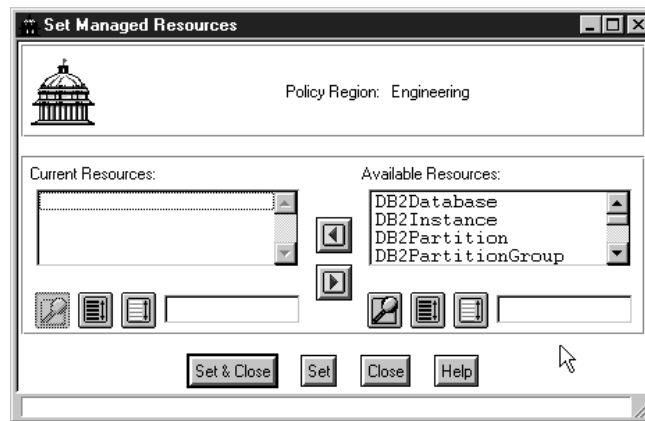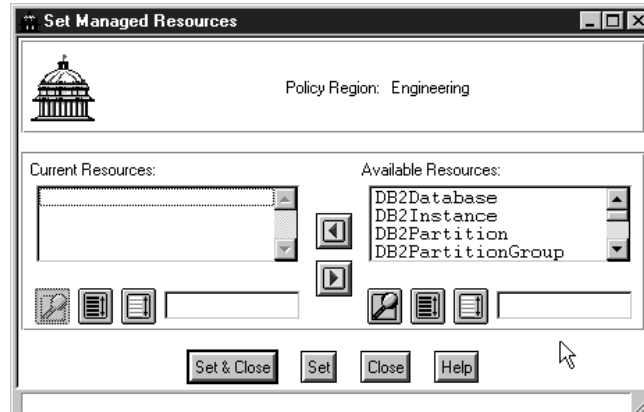

Figure 28. The Set Managed Resources Window

  b. In the **Available Resources** box, select **DB2PartitionGroup** and click on the ◁ push button. **DB2PartitionGroup** moves to the **Current Resources** box.
  c. Click on the **Set & Close** push button.
3. From the Policy Region window, select **Create→DB2PartitionGroup**. The Create DB2PartitionGroup window opens (Figure 29 on page 41).
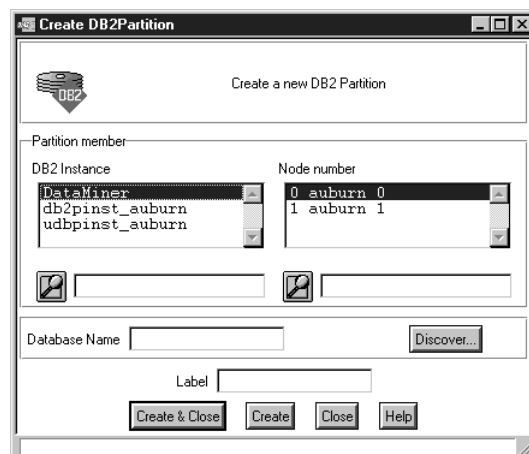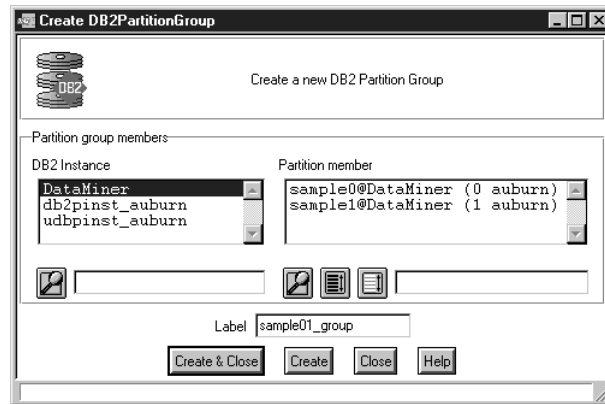
Figure 29. The Create DB2PartitionGroup Window

4. From the **DB2 Instance** box, select the name of the partitioned database server (DB2 instance) from which you want to create a DB2 partition group object. The database partitions for the selected DB2 instance are displayed in the **Partition Member** box.

5. From the **Partition Member** box, select the partitions for which you want to create a DB2 partition group object.

6. In the **Label** field, type a unique label for the icon for the group of DB2 partitions selected in the **Partition Member** box.

7. Click on the **Create & Close** push button to create the DB2 partition group object.

Command Line

Use the following command to create a DB2 partition group object from a command line:

```
wcrdb2partgp -l partition_label -i instance_label -d database
  -n "node_number node_number ..." -p policy_region
```

where:

-l *partition_label*
> Specifies a unique label for the DB2 partition group object.

-i *instance_label*
> Specifies the label of the instance containing the DB2 partition group.

-d *database*
> Specifies the name of the existing partitioned database in the specified instance from which you want to create a DB2 partition group object.

-n *"node_number node_number ..."*
> Specifies the node numbers of the existing partitioned database in the specified instance for which you want to create an object.

-p *policy_region*
> Specifies the name of the policy region to which the DB2 partition group object will belong.

Example:

Creating DB2 Partition Group Objects

To create a DB2 partition group object for a partition group labeled sample_grp@db2pel, with the node numbers 0, 1, and 2, in the instance labeled db2pel@catalina, in the partitioned database sample in the policy region westcoast, enter the following command:

```
wcrdb2partgp -l sample_grp@db2pel -i db2pel@catalina -d sample -n"0 1 2" -p westcoast
```

Changing Members of a Partition Group

From your TME 10 desktop, you can change the members of a DB2 partition group.

To change the members of a DB2 partition group:

1. From the desktop, double-click on the policy region that contains the DB2 partition group object whose members you want to change. The Policy Region window opens.

2. Select **Change** from the pop-up menu for the DB2 partition group object whose members you want to change. The Change DB2PartitionGroup Members window opens (Figure 30).
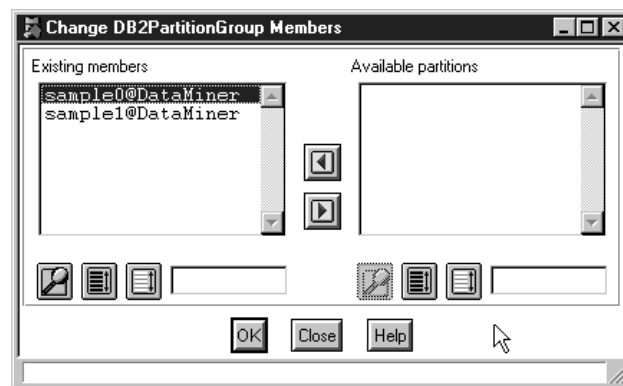
Figure 30. The Change DB2PartitionGroup Members Window

3. Optional: In the **Available Partitions** box, select the partitions that you want to add to this partition group and click on the ▣ push button.

4. Optional: In the **Existing Members** box, select the partitions that you want to remove from this partition group and click on the ▣ push button.

5. Click on **OK**.

Viewing DB2 Object Properties

From your TME 10 desktop, you can view DB2 configuration properties for your DB2 managed resources.

You can view the following properties of a DB2 instance:

- Managed node
- Instance name
- Node type
- Default DB path

- Database manager configuration release level
- CPU speed
- Default client communication
- IPX socket
- Service name

You can view the following properties of a DB2 database:
- Managed node
- Instance name
- Database name
- Database is consistent
- Backup pending
- Rollforward pending
- Database territory
- Database code set

You can view the following properties of a DB2 partition or members of a partition group:
- Managed node
- Instance name
- Database name
- Node number
- Database is consistent
- Backup pending
- Rollforward pending
- Database territory
- Database code set

For details about the configuration properties, see the *DB2 Command Reference* for the version of DB2 that you are using.

To view DB2 managed resource object properties:
1. From the desktop, double-click on the policy region that contains the DB2 managed resource object whose properties you want to view. The Policy Region window opens.
2. Select **Properties** from the pop-up menu for the DB2 managed resource object whose properties you want to view. The DB2 Instance Properties window, DB2 Database Properties window, DB2 Partition Properties window, or DB2 Partition Group Properties window opens. Figure 31 on page 44 shows an example of a DB2 Instance Properties window.

Viewing DB2 Object Properties
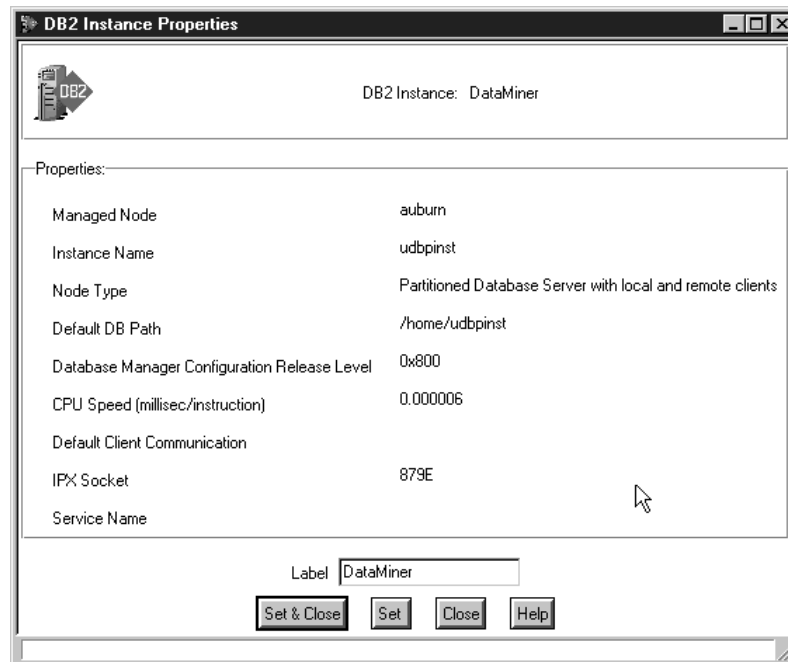
Figure 31. The DB2 Instance Properties Window

From this window, you can view the current values of the properties for the DB2 managed resource object that you selected.

Starting and Stopping Instances, or Nodes for Partitions or Partition Groups

From your TME 10 desktop, you can start and stop remote DB2 instances, and nodes for partitions and partition groups. The Start and Stop functions run the DB2 commands **db2start** and **db2stop** at the managed node where the instance or partitions reside. For DB2 partition and partition group objects, the Start and Stop functions start up and shut down the instance on the node or nodes that the object represents.

To start or stop a DB2 instance, partition, or partition group:

1. From the desktop, double-click on the policy region in which you want to start or stop a DB2 instance, partition, or partition group. The Policy Region window opens.
2. Select **Startup** or **Shutdown** from the pop-up menu for the DB2 managed resource object that you want to start or stop. The startup window or shutdown window for the DB2 managed resource opens. Figure 32 on page 45 shows the Startup DB2 Instance window.

Figure 32. The Startup DB2 Instance Window

3. Click on **Yes** to start or stop the DB2 managed resource.

Running DB2 Commands from the DB2 Command Line Processor

From your TME 10 desktop, you can open a window from which you can run DB2 commands against a DB2 instance object, DB2 database object, or DB2 partition object and see the results.

To run DB2 commands from the DB2 command line processor:

1. From the desktop, double-click on the policy region in which you want to run DB2 commands. The Policy Region window opens.
2. Select **Start DB2 CLP** from the pop-up menu for the DB2 managed resource object against which you want to run DB2 commands. The DB2 Command Line Processor window opens (Figure 33).
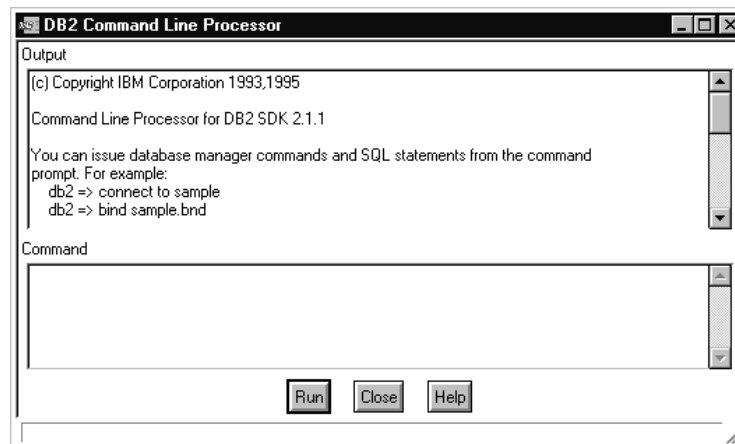
Figure 33. The DB2 Command Line Processor Window

The DB2 Command Line Processor window contains an **Output** box and a **Command** box. The **Output** box contains information on how to use this window and the latest results or information from any commands that you run.

3. Type the DB2 command that you want to run in the **Command** box. For more information on the commands that you can run from the DB2 Command Line Processor window, see the *DB2 Command Reference* for the version of DB2 that you are using.
4. Click on the **Run** push button to execute the DB2 command that you specified. The results are displayed in the **Ouput** box.

   Attention: Only one DB2 command can be run at a time.

5. To run another command, repeat steps 3 and 4 on page 45.

6. Click on **Close** to close the DB2 Command Line Processor window.

Launching the DB2 Control Center

When working with a particular DB2 instance or database in DB2 ECC, you might need to access specific DB2 objects belonging to the DB2 instance or database. For example, if a monitor that you are running indicates that space on a system management table space is nearing full, you can launch the DB2 Control Center against that database and remove unnecessary tables in that table space.

From the TME 10 desktop, you can go from a DB2 instance or database object that resides on a Windows NT managed node to the same object on the DB2 Control Center desktop. This feature gives you access to all the related DB2 objects, such as tables, table spaces, indexes, views, and triggers, that belong to your DB2 ECC object.

From the DB2 Control Center, you can perform many tasks on your DB2 UDB for NT objects, such as creating tables, running statistics, and adding users. See *DB2 Universal Database Administration Getting Started* for more information about the DB2 Control Center.

To launch the DB2 Control Center:

1. From the desktop, double-click on the policy region that contains the DB2 instance or database object from which you want to launch the DB2 Control Center. The Policy Region window opens.

2. Select **Launch DB2CC** from the pop-up menu for the DB2 instance or database object from which you want to launch the DB2 Control Center.

   The DB2 Control Center desktop opens and positions you at the DB2 instance or database object on the object tree that corresponds to the object from which you launched on the TME 10 desktop. From this position, you can navigate down the object tree and perform any tasks that the DB2 Control Center provides.

If you launch the DB2 Control Center from a DB2 instance or database that is not on the local node but is cataloged with DB2, then the launch will be in context, that is the object tree will start at the DB2 object that corresponds to the object from which the DB2 Control Center was launched. Otherwise, the launch will start the object tree from the SYSTEM. You can then add the other node (or system) and catalog the instance or database. See *DB2 Universal Database Administration Getting Started* for more information on configuring the DB2 Control Center.

Migrating DB2 ECC Objects to DB2 UDB

If you are migrating existing DB2 instances and databases from DB2 for common servers to DB2 UDB, you need to perform the following steps to make the change known to DB2 ECC:

1. On the managed node where the DB2 instance is to be migrated, stop the monitoring agent by performing one of the following tasks:
   - For AIX or Solaris: Run the ECC_Set_Up_Montoring_Agent task on the managed node. Select the **Remove and stop** action on the task argument window.

- For Windows NT: Run the ECC_Stop_Monitoring_Agent task on the managed node.

2. If there are any DB2 databases, belonging to that DB2 instance, that you are not planning to migrate to DB2 UDB, remove the DB2 ECC database objects that represent those databases.

3. Migrate the DB2 instance and databases according to the DB2 UDB documentation.

4. Press and hold the right mouse button on the DB2 instance object that you just migrated and select **Properties** from the pop-up menu. The DB2 Instance Properties window opens and the **Database Manager Configuration Release Level** field should now show 0x800 for DB2 UDB.

5. On the managed node where the DB2 instance was migrated, start the monitoring agent by performing one of the following tasks:

   - For AIX or Solaris: Run the ECC_Set_Up_Montoring_Agent task on the managed node. Select the **Add and start** action on the task argument window.

   - For Windows NT: Run the ECC_Start_Monitoring_Agent task on the managed node.

If you created any jobs in a DB2 ECC task library, you might need to redefine the jobs if any task arguments changed from DB2 for common servers to DB2 UDB. For more information on creating jobs, see "Scheduling Tasks in the DB2 ECC Task Libraries" on page 52.

Chapter 5. Working with the DB2 ECC Task Libraries

A *task* is a definition of an operation that needs to be done to the management environment again and again, such as backing up a database. You can use tasks to define a set of complex operations that you can easily run on any machine without consideration to platform type. A *job* is a task that is executed on specific managed resources. Tasks and jobs are stored in *task libraries* so that they can be used again and again. Tasks and jobs are subject to policy and this policy is enforced by the policy region that contains the task library in which the task or job is defined.

DB2 ECC provides three task libraries to support DB2 monitoring and task automation:
- Monitor and Administration Tasks (DB2_ECC-AdminTasks)
- DB2 Database Tasks (DB2_ECC-DatabaseTasks)
- Data Replication Tasks (DB2_ECC-ReplicationTasks)

For a complete list of tasks provided in the DB2 ECC task libraries, see "Appendix C. DB2 ECC Task Libraries" on page 101.

This chapter provides steps for performing the following operations with the DB2 ECC task libraries:
- Running tasks in the DB2 ECC task libraries
- Scheduling tasks in the DB2 ECC task libraries

Running Tasks in the DB2 ECC Task Libraries

You can run a task that exists in a DB2 ECC task library using the TME 10 desktop or a command line.

Desktop

To run a task in a DB2 ECC task library:
1. From the DB2 ECC default policy region, drag the icon for the DB2 ECC task library that contains the task that you want to run to your desktop. Your desktop now contains a reference to that DB2 ECC task library.

   Attention: To remove the DB2 ECC task library icon from your desktop, select the icon and then select **Edit**→**Remove**. Do *not* select **Edit**→**Delete**, which might delete the DB2 ECC task library.
2. Select **Execute Task** from the pop-up menu for the DB2 ECC task library. The Task Library window opens (Figure 34 on page 50).

Running Tasks

Figure 34. The Task Library Window

3. Select the task that you want to run and click on the **Execute Task** push button. The Execute Task window opens (Figure 35).
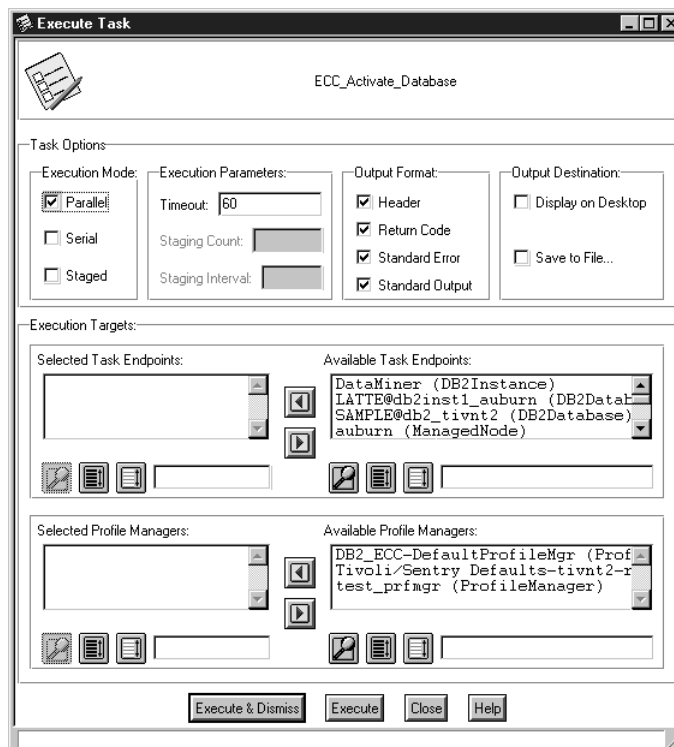
Figure 35. The Execute Task Window

4. Specify the execution mode by selecting one of the following check boxes:

Parallel
If you want the task to run on all targets in parallel.

Serial If you want the task to run synchronously on all targets. The task is executed on the endpoints in alphabetical order.

Staged
If you want the task to run in a staged fashion on sets of managed nodes. If you select the **Staged** check box, you must also specify the

number of endpoints to include in each staged set and the number of seconds for the staging interval between each set's startup time. The task is executed on the endpoints in alphabetical order.

5. In the **Timeout** field, type the timeout value (in seconds) for the task. This value specifies the amount of time that the task output window remains open when the task is run. (The task output window is displayed only if you select the **Display on Desktop** check box in step 7.) The default is 60 (seconds).

   Attention: If the task takes longer than the timeout value to complete, the task output window closes, but the task continues to run.

6. Choose the types of output that you want the task to return by selecting one or more of the following check boxes:

   Header
   Returns a descriptive header for each record.

   Return Code
   Returns the task's return code.

   Standard Output
   Returns the standard output.

   Standard Error
   Returns the standard output error.

7. If you want the task output to be displayed on the desktop, select the **Display on Desktop** check box.

8. If you want to save the task output to a file:

   a. Select the **Save to File** check box. The Destination for Task Output window opens (Figure 36).

Figure 36. The Destination for Task Output Window

   b. In the **On Host** field, type the name of the managed node on which to save the output. The managed node must be a TME client. For example, type `catalina`.

   c. In the **Output File** field, type the absolute path to which the output of the task is to be written. For example, on an AIX system, type `/tmp/mytask.out`.

   d. Click on the **Set & Close** push button.

9. Choose the endpoints on which you want to run the task. You can either:

   • Run the task on specific endpoints.

   Select the endpoints that you want from the **Available Task Endpoints** box and click on the ▣ push button. The selected endpoints move to the

   Selected Task Endpoints box.

   • Run the task on all subscribers of the specified profile managers.

Select the profile managers that you want from the **Available Profile Managers** box and click on the ▣ push button. The selected profile

managers move to the **Selected Profile Managers** box.

10. Click on the **Execute & Dismiss** push button to run the task and return to the Task Library window.

Command Line

Use the **wruntask** command to run a task from a task library. The syntax for the command might vary depending on the version of the TME framework that you are using. For more information, see the *TME 10 Framework Reference Manual*.

Example:

To start the DB2 ECC monitoring agent on the host named bandit, enter the following command:

```
wruntask -t ECC_Start_Monitoring_Agent -l DB2_ECC-AdminTasks -h bandit
```

Scheduling Tasks in the DB2 ECC Task Libraries

DB2 ECC utilizes the TME 10 scheduler to allow you to schedule tasks in a DB2 ECC task library to run at a later time or on a regular basis. To schedule a task, you first create a job for the task and then schedule the job.

You must have the proper TMR role to execute the job that you're scheduling. If you do not have the proper TMR role, the job will fail when it is executed. For more information on the TMR roles required for each DB2 ECC task, see "Appendix B. DB2 ECC Administrator Roles" on page 97.

Desktop

To schedule a task in a DB2 ECC task library, you need to:
1. Create a job in the DB2 ECC task library that contains the task that you want to schedule.
2. Schedule the job using the TME 10 scheduler.

These steps are discussed in detail in this section.

Step 1: Create a Job in the DB2 ECC Task Library

To create a job in a task library that contains the task that you want to schedule:
1. From the DB2 ECC default policy region, drag the icon for the DB2 ECC task library that you want to your desktop. Your desktop now contains a reference to that DB2 ECC task library.

    Attention: To remove the DB2 ECC task library icon from your desktop, select the icon and then select **Edit**→**Remove**. Do *not* select **Edit**→**Delete**, which might delete the DB2 ECC task library.
2. Press and hold the right mouse button on the icon for the DB2 ECC task library and select **Create Job** from the pop-up menu. The Create Job window opens (Figure 37 on page 53).
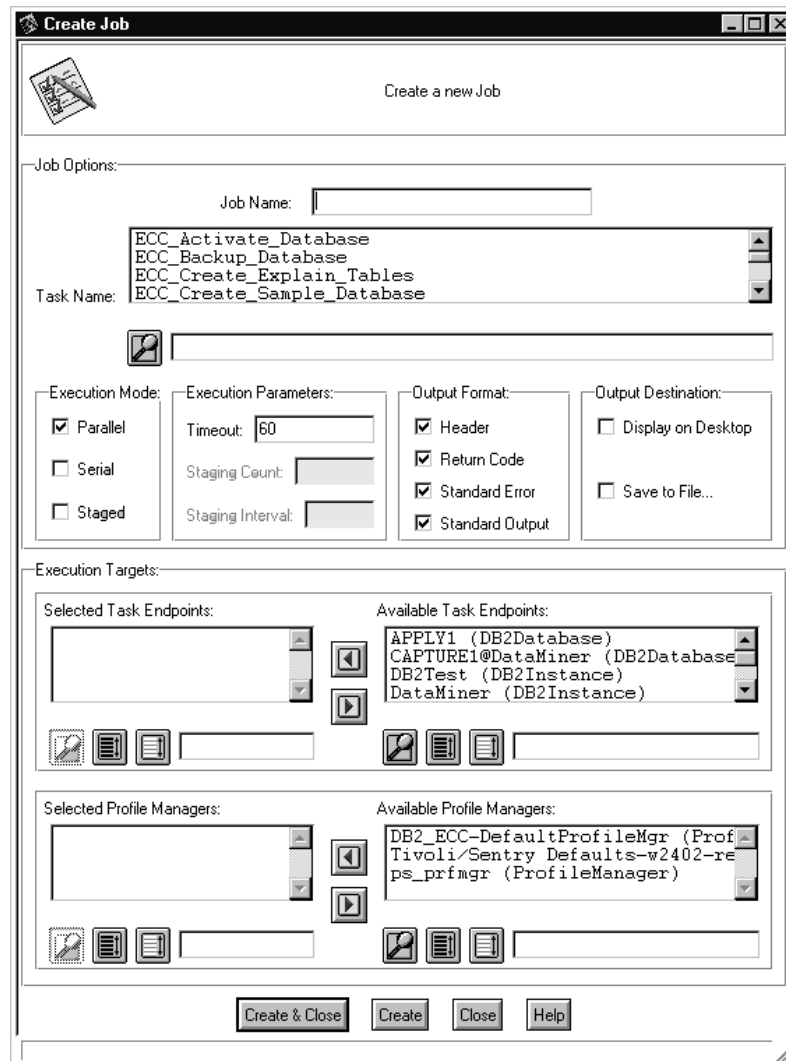
Figure 37. The Create Job Window

3. In the **Job Name** field, type a name for the job. The job name can include letters, numbers, underscores, dashes, periods, and blanks.

4. From the **Task Name** box, select the task that the job will execute.

5. Specify the execution mode by selecting one of the following check boxes:

   Parallel
   > If you want the job to run on all targets in parallel.

   Serial If you want the job to run synchronously on all targets. The job is executed on the endpoints in alphabetical order.

   Staged
   > If you want the job to run in a staged fashion on sets of managed nodes. If you select the **Staged** check box, you must also specify the number of endpoints to include in each staged set and the number of seconds for the staging interval between each set's startup time. The job is executed on the endpoints in alphabetical order.

6. In the **Timeout** field, type the timeout value (in seconds) for the job. This value specifies the amount of time that the job output window remains open when the

job is run. (The job output window is displayed only if you select the **Display on Desktop** check box in step 8.) The default is 60 (seconds).

Attention: If the job takes longer than the timeout value to complete, the job output window closes, but the job continues to run.

7. Choose the types of output that you want the job to return by selecting one or more of the following check boxes:

Header
> Returns a descriptive header for each record.

Return Code
> Returns the job's return code.

Standard Output
> Returns the standard output.

Standard Error
> Returns the standard output error.

8. If you want the job output to be displayed on the desktop, select the **Display on Desktop** check box.

9. If you want to save the job output to a file:
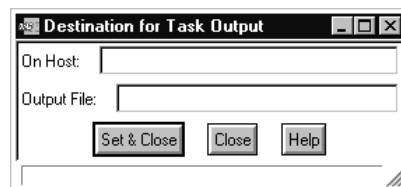
   a. Click on the **Save to File** check box. The Destination for Task Output window opens.

   b. In the **On Host** field, type the name of the managed node on which to save the output. The managed node must be a TME client. For example, type catalina.

   c. In the **Output File** field, type the absolute path to which the output of the job is to be written. For example, on an AIX system, type /tmp/myjob.out.

   d. Click on the **Set & Close** push button.

10. Choose the endpoints on which you want to run the job. You can either:

   • Run the job on specific endpoints.

     Select the endpoints that you want from the **Available Task Endpoints** box and click on the ▣ push button. The selected endpoints move to the

     Selected Task Endpoints box.

   • Run the job on all subscribers of the specified profile managers.

     Select the profile managers that you want from the **Available Profile Managers** box and click on the ▣ push button. The selected profile

     managers move to the **Selected Profile Managers** box.

11. Click on the **Create & Close** push button to create the job and return to the window for the task library. An icon for the job that you created appears in the window (Figure 38 on page 55).
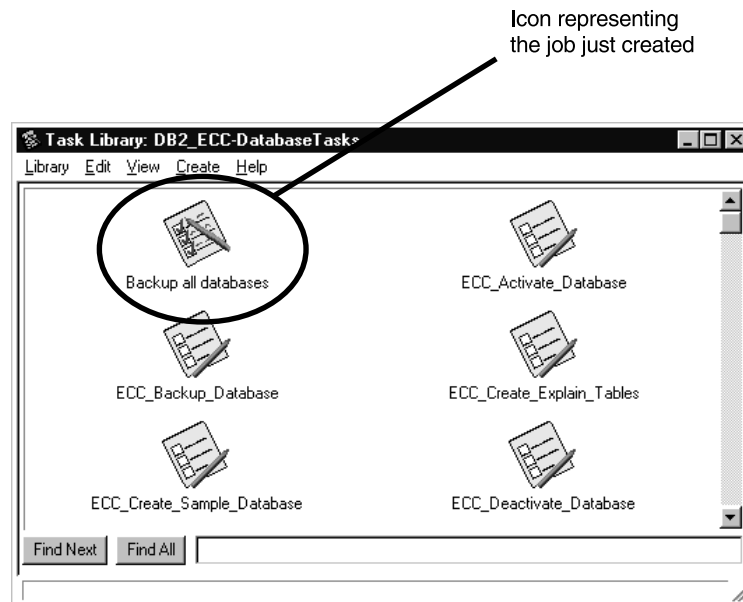
Figure 38. The Task Library: DB2_ECC-DatabaseTasks Window with Job Icon

Step 2: Schedule the Job Using the TME 10 Scheduler

To schedule a job:

1. From the window for the task library, drag the job icon that you just created onto the TME 10 scheduler.

    • If the task does not require any arguments, the Add Scheduled Job window opens. Go to step 3 on page 57.

    • If the task requires that certain arguments be specified, a task window opens. Figure 39 on page 56 shows an example of a task window for the back up database task.
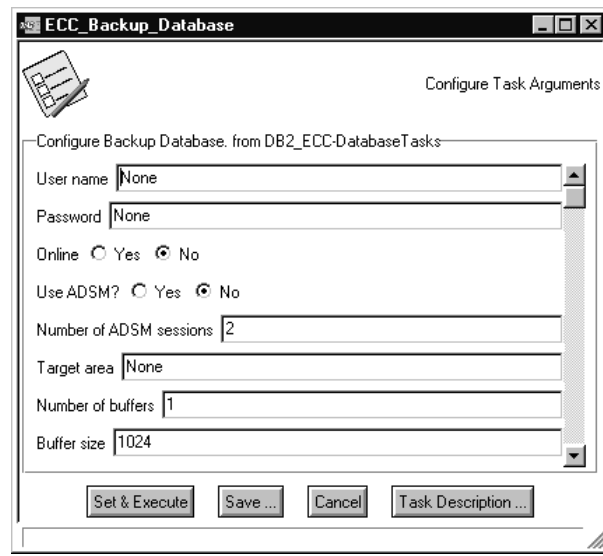
Figure 39. An Example of a Task Window. This window is for the back up database task.

2. If a task window opens, specify the necessary arguments and click on the **Set & Execute** push button to set the task arguments and open the Add Scheduled Job window (Figure 40 on page 57).

Figure 40. The Add Scheduled Job Window

3. In the **Job Label** field of the Add Scheduled Job window, type a label for the icon for the job. The job label can include letters, numbers, underscores, dashes, periods, and blanks. If you do not specify a label, the job name is used.

4. In the **Schedule Job For** group box:

   a. Specify the date in the **Month**, **Day**, and **Year** fields.

   b. Specify the time using the **Hour** and **Minute** boxes and the **AM** and **PM** radio buttons.

5. If the job will be run more than once, specify the repeat options in the **Repeat The Job** group box.

   a. Select one of the following repeat options:

      • If you want to repeat the job indefinitely, select the **Repeat the job indefinitely** check box.

      • If you want to repeat the job a finite number of times, select the **Repeat the job** check box and specify the number of times.

   b. Specify the interval between start times for the job.

6. Specify the action that you want to occur when the job completes in the **When Job Complete** group box. Select one or more of the following check boxes:

Post Tivoli Notice
Specifies that a Tivoli notice be posted. Click on the **Available Groups** push button to open the Available Groups window to specify the notice groups to which the notice will be posted.

Post Status Dialog on Desktop
Specifies that the status message be displayed on the desktop. Type the message that you want to display in the field.

Send email to
Specifies that an e-mail message be sent when the job completes. Type the e-mail address in the field.

Log to File
Specifies that the status be written to a file when the job completes. You can specify the file either with the **Host** and **File** fields or by clicking on the **File Browser** push button and selecting a file.

7. If you want to set any retry, cancel, or restriction options for the job:

   a. Click on the **Set Retry/Cancel/Restriction Options** push button. The Set Retry/Cancel Options window opens (Figure 41).
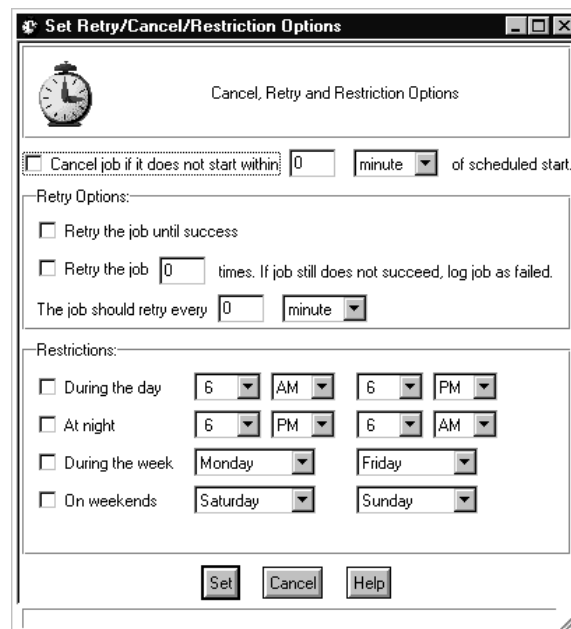
Figure 41. The Set Retry/Cancel Options Window

   b. Optional: Select the **Cancel job** check box and specify an amount of time to wait before canceling a job.

   c. Optional: Specify the retry options:

      1) Select one of the following check boxes:

         Retry the job until success
         To retry the job until success.

         Retry the job
         To retry the job a specified number of times. Type the number of times that you want in the field.

        2) Use the **The job should retry every** fields to specify how often to retry the job.

    d. Optional: Specify the available run times by selecting the appropriate check boxes in the **Restrictions** group box and filling in the appropriate fields. The run times that you specify in these fields are the times that the job can run.

    e. Click on the **Set** push button to set your options and return to the Add Scheduled Job window.

8. Click on the **Schedule Job & Close** push button to schedule the job and return to the desktop.

Command Line

Use the **wcrtjob** command to create a job in a task library from a command line. Use the **wschedjob** command to schedule a job that exists in a DB2 ECC task library from a command line. Only jobs that exist in a task library can be scheduled from a command line. The syntax for the commands might vary depending on the version of the TME framework that you are using. For more information, see the *TME 10 Framework Reference Manual*.

Example:

To schedule a job that contains the ECC_Invoke_Stored_Procedure task from the DB2_ECC-DatabaseTasks task library to run at 9:55 on 10/10/97, label the job job1, log the status to the /tmp/job1.log file, and send an e-mail to slainte.stl.ibm.com when the job completes, enter the following command:

```
wschedjob -t '10/10/97 09:55' -L DB2_ECC-DatabaseTasks -n Job_InvokeProc
  -l job1 -f /tmp/job1.log -h slainte.stl.ibm.com
```

Scheduling Tasks

Chapter 6. Working with DB2 ECC Monitoring Collections

This chapter introduces the DB2 ECC monitors and tells you how to set them up and activate them. The topics covered are:

- DB2 ECC monitors: their nature; how they are organized; how they can be configured; how they are stored and activated
- DB2 ECC default monitors: where they are stored; how to activate them quickly
- How to store, define, and activate DB2 ECC monitors that you configure
- How to determine what DB2 ECC monitors are operating on a given resource
- How to arrange for information captured by DB2 ECC monitors to be correlated with information from other sources in your Tivoli Management Region (TMR)

Monitors and Monitoring Collections

This section describes the nature of DB2 ECC monitors, explains how they are organized and what their purposes are, summarizes how you can configure them, and discusses how they are stored and activated.

In the TME 10 environment, a *monitor* is a mechanism that typically captures and returns information about a resource or application within the environment. For example, most DB2 ECC monitors capture and return information about DB2 instances and databases—information such as whether an instance is up or down, or whether a database is active, available, or in some other state. Other DB2 ECC monitors capture and return information about applications that use DB2 instances and databases—applications such as DPROPR and user-supplied programs.

Monitors are organized into groups called *monitoring collections*. For example, the DB2 ECC monitors "table: Reorg needed" and "table: Overflow accesses" belong to a monitoring collection for statistics on tables and packages. The DB2 ECC monitoring collections are:

- DB2 agents and applications (DB2_Agent_Monitors)
- DB2 basic monitoring (DB2_Basic_Monitors)
- DB2 buffer pool I/O monitoring (DB2_Bufio_Monitors)
- DB2 configuration (DB2_Config_Monitors)
- DB2 data replication relational (DB2_DataReplication_Monitors)
- DB2 locks and deadlocks (DB2_Lock_Monitors)
- DB2 query (DB2_Query_Monitors)
- DB2 SNMP (DB2_SNMP_Monitors)
- DB2 sort work (DB2_Sort_Monitors)
- DB2 SQL statement activity (DB2_Stmt_Monitors)
- DB2 table (DB2_Table_Monitors)

For a brief description of each monitoring collection and each monitor, see "Appendix D. DB2 ECC Monitoring Collections and Default Monitors" on page 105. For a full description and guidelines on using the monitors, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

You can configure DB2 ECC monitors to run on specified schedules. For example, you could configure the monitor "table: Reorg needed" so that it checks a particular table once a week.

Monitors and Monitoring Collections

You can also configure the monitors to perform actions (called *response actions*) whenever they run, as well as when certain criteria (called *thresholds*) are met. For example, you could configure the "table: Reorg needed" monitor so that both:

- Each time it checks a table, you're notified as to whether the table needs reorganizing.
- If the need exists, a DB2 ECC task is run to reorganize the table.

Like all monitors in the TME 10 environment, the DB2 ECC monitors are managed by the Tivoli/Sentry or TME 10 Distributed Monitoring applications (called *Sentry* in this book). When you define a monitor, you store it in a Sentry profile.

Before you run a monitor against a managed resource, you must subscribe that resource to the profile manager that contains the Sentry profile in which the monitor is stored. When you're ready to run the monitor, you distribute the profile to the resource. Depending on the monitors that it contains, you can distribute a Sentry profile to one or more of five kinds of resources—DB2 instances, DB2 databases, DB2 partitions, DB2 partition groups, and managed nodes. These resources are the ultimate destination of the distribution and are therefore called *endpoints.*

Before distributing a Sentry profile to an endpoint, you can distribute it to a profile manager other than the one that it's in. When it's received into this other profile manager, you or other administrators can edit the monitors in the profile or add more monitors to it.

For more information about profiles, profile managers, subscription, and distribution, see "Elements in the DB2 ECC Environment: Terminology" on page 1 and "Terms and Abbreviations" on page 133. For more information about Sentry, see the *Tivoli/Sentry User's Guide* or the *TME 10 Distributed Monitoring User's Guide.*

Firing off DB2 ECC Default Monitors

Schedules, response actions, and other configuration parameters are predefined for a starter set of default DB2 ECC monitors. You can use these monitors to obtain information about your DB2 environment as soon as you install DB2 ECC. This section explains how the default monitors are stored and how you can start firing them off.

The default monitors are separated into four Sentry profiles supplied as defaults by DB2 ECC. These profiles are:

DB2_ECC-DBMSMonProfile
Contains monitors for DB2 nonpartitioned database servers.

DB2_ECC-DatabaseMonProfile
Contains monitors for DB2 databases.

DB2_ECC-DBPartMonProfile
Contains monitors for DB2 partitions.

DB2_ECC-GenericMonProfile
Contains monitors for managed nodes.

The default profiles are stored in DB2 ECC's default profile manager, DB2_ECC-DefaultProfileMgr.

For a list of the default monitors, see "DB2 ECC Default Monitors" on page 121.

You can fire off the default monitors using the TME 10 desktop or a command line.

Desktop

To start firing off the default monitors from the desktop:

1. Add subscribers to DB2_ECC-DefaultProfileMgr.

   a. Double-click on the icon for DB2_ECC-DefaultProfileMgr. The window for DB2_ECC-DefaultProfileMgr opens (Figure 42).
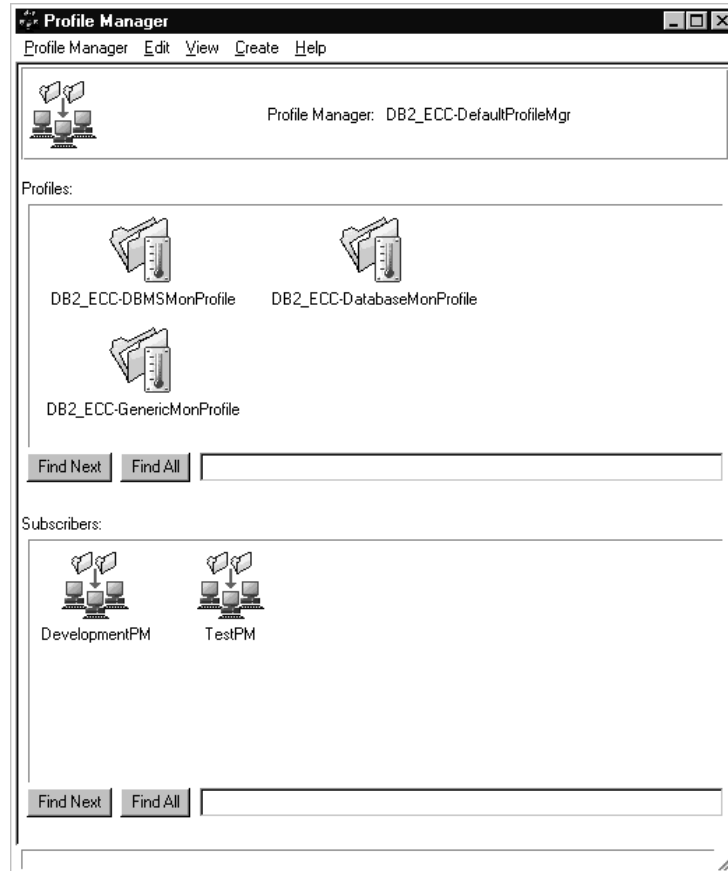
Figure 42. The Profile Manager Window for DB2_ECC-DefaultProfileMgr

   b. Select **Profile Manager→Subscribers**. The Subscribers window opens (Figure 43 on page 64).

Firing Off DB2 ECC Default Monitors
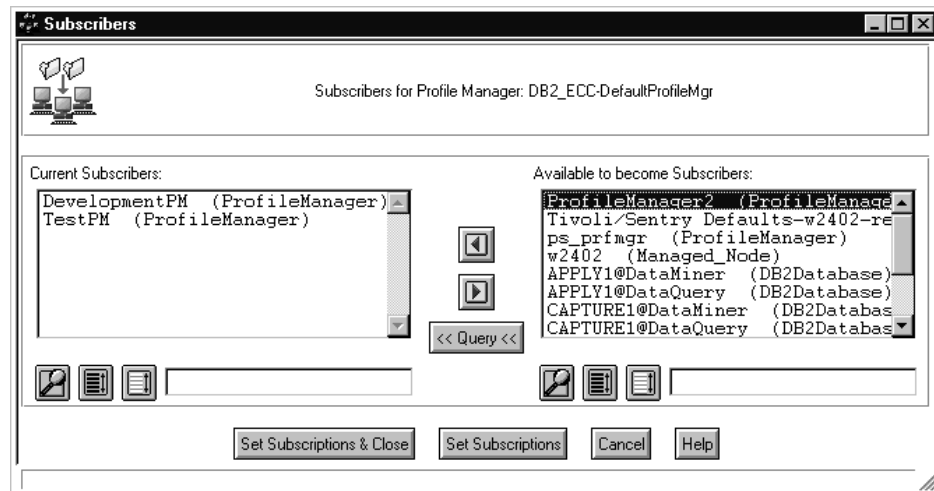
Figure 43. The Subscribers Window

    c. In the **Available to become Subscribers** box, select the resource or resources that you want to monitor and click on the ▣ push button. The selected resource or resources move to the **Current Subscribers** box.

    d. Click on **Set Subscriptions & Close**. The icon of each selected resource is displayed in the window for DB2_ECC-DefaultProfileMgr.

2. Drag and drop the icon for each default profile onto the icon for the resource that you want the monitors in that profile to monitor. Sentry distributes the profiles to the resources, and the default monitors start running as scheduled. For more information on the types of endpoints on which you can drop each default profile, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

3. Optional: To find out the status of the monitors in each default profile:

    a. On your desktop, select **Desktop→Navigator**. The Navigator window opens.

    b. On the Navigator window, select the **PolicyRegion** check box. The policy regions that you can use are listed in the **Resources** box.

    c. Select **DB2_ECC-DefaultPolicyRegion** and click on the **Go To** push button. The policy region window for the DB2 ECC default policy region opens.

    d. Click on the icon for the DB2 ECC default indicator collection (DB2_ECC-IndicatorCollection). The Sentry Indicator Collection window opens, showing an indicator icon for each default profile. The indicator icons look like thermometers. The "temperature" on each indicator icon reflects the highest response level that is being reached in the profile with which the icon is associated. For information about response levels, see "Step 5: Specify Response Levels, the Condition for Triggering a Response, and the Nature of the Response" on page 72.

Command Line

To fire off default monitors from the command line:

1. Use the **wsub** command to add subscribers.

    Example:

To add the subscribers DevelopmentPM and TestPM to the DB2_ECC-DefaultProfileMgr, enter the following command:

```
wsub @DB2_ECC-DefaultProfileMgr @ProfileManager:DevelopmentPM
  @ProfileManager:TestPM
```

2. Use the **wdistrib** command to distribute the profiles.

   Example:

   To distribute the default profile DB2_ECC-DBMSMonProfile to the DB2 instance endpoints inst1 and inst2, enter the following command:

```
wdistrib @SentryProfile:DB2_ECC-DBMSMonProfile @DB2Instance:inst1
  @DB2Instance:inst2
```

The syntax might vary for the commands in this section depending on the version of the TME framework that you are using. See the *TME 10 Framework Reference Manual* for more information about the syntax of the commands.

Setting Up DB2 ECC Monitors in Your Own Sentry Profiles

This section explains how to set up DB2 ECC monitors in Sentry profiles that you create. Each of the following steps in this task is described in detail in this section:

1. Set up a policy region to contain the resources that you want.
2. Add a profile manager, a Sentry profile and, optionally, an indicator collection.
3. Add subscribers to the profile manager.
4. Select a monitoring source; specify arguments as needed.
5. Specify response levels, the condition for triggering a response, and the nature of the response.
6. Schedule the monitor.

At this point, you can repeat steps 4, 5, and 6 to define additional monitors.

7. **Important for a Sentry profile with any DB2 ECC monitors:** Set a default user ID.
8. Save and distribute the Sentry profile.

The steps in this section form a suggested scenario; you might want to depart from it as you gain experience in setting up monitors. For example, you could set subscribers after defining your monitors, rather than before.

The following section explains how to perform these steps from the desktop. You can also perform the steps from the command line, as described in "Command Line" on page 80.

Desktop

Step 1: Set Up a Policy Region to Contain the Resources That You Want

To set up a policy region to contain the managed resources that you'll use:

1. Select or create the policy region:
   - If you want to use an existing policy region:
     a. On your desktop, select **Desktop→Navigator**. The Navigator window opens.
     b. On the Navigator window, select the **PolicyRegion** check box. The policy regions that you can use are listed in the **Resources** box.

    c. Select the policy region that you want.

- If you want to create a policy region, follow the steps in "Creating Policy Regions" on page 28.

2. Specify the types of resources that you want to work with:

    a. If you created the policy region, open its window by clicking on its icon on your desktop. (If you're using an existing policy region that you accessed from the Navigator window, its window is still open.)

    b. On the policy region's window, select **Properties**→**Managed Resources**. The Set Managed Resources window opens (Figure 44).
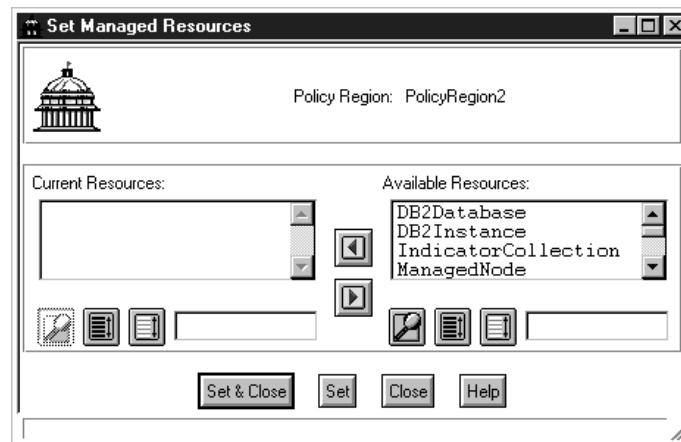
Figure 44. The Set Managed Resources Window

    c. If the **Current Resources** box doesn't list the types of managed resources that you want to work with, select these types in the **Available Resources** box. In DB2 ECC, the types of resources used in monitoring on AIX and NT platforms are:

- **ProfileManager** (required)
- **SentryProfile** (required)
- **IndicatorCollection** (optional; select this type of resource if you want to graphically represent response levels that have been reached or exceeded)

    d. Click on the ⊡ push button. The selected types of resources move to the

       Current Resources box.

    e. Click on the **Set and Close** push button.

Step 2: Add a Profile Manager, a Sentry Profile, and an Indicator Collection

You begin this step with the Policy Region window still open.

To add a profile manager and Sentry profile to the policy region, and to create an indicator collection for the Sentry profile:

1. Create the profile manager:

a. On the Policy Region window, select **Create**→**Profile Manager**. The Create Profile Manager window opens (Figure 45).
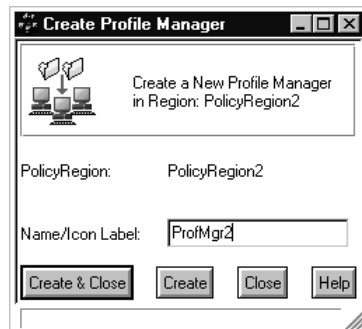
Figure 45. The Create Profile Manager Window

b. In the **Name/Icon Label** field, type a unique label for the profile manager's icon.

c. Click on the **Create & Close** push button. The profile manager's icon is displayed in the Policy Region window.

2. Create the Sentry profile:

a. Double-click on the profile manager's icon. The Profile Manager window opens.

b. On the Profile Manager window, select **Create**→**Profile**. The Create Profile window opens (Figure 46).
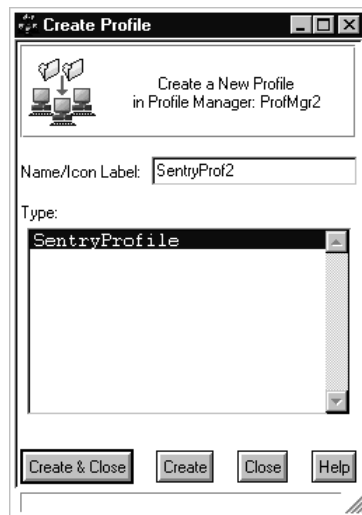
Figure 46. The Create Profile Window

c. In the **Name/Icon** field of the Create Profile window, type a unique label for the Sentry profile's icon.

d. In the **Type** box of the Create Profile window, select **SentryProfile**.

e. Click on **Create & Close**. The new Sentry profile's icon is displayed in the Profile Manager window.

3. Optional: Add an indicator collection for your Sentry profile:

Setting Up Your Own Monitors

    a. On the window for the policy region, select **Create→IndicatorCollection**.
       The Create Sentry Indicator Collection window opens (Figure 47).
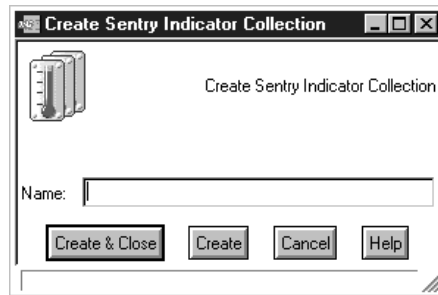
Figure 47. The Create Sentry Indicator Collection Window

    b. In the **Name** field, type a name for the indicator collection. The name must differ from that of any other managed resource in your TMR.

    c. Click on the **Create & Close** push button. An icon representing the indicator collection that you created is displayed in the Policy Region window.

    d. On the Sentry Profile Properties window, select **Sentry→Select Indicator Collection**. The Select Indicator Collection window opens (Figure 48).
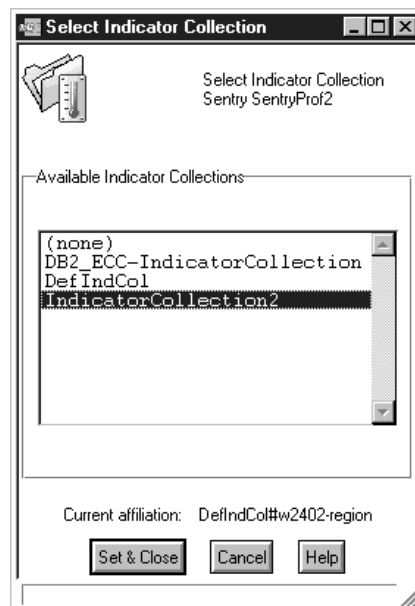
Figure 48. The Select Indicator Collection Window

    e. In the **Available Indicator Collections** box, select the indicator collection that you created.

    f. Click on the **Set & Close** push button. Your indicator collection is now associated with your Sentry profile.

  4. Close the window for the policy region that you created.

Step 3: Add Subscribers to the Profile Manager

You begin this step with the Profile Manager window still open.

To add subscribers to the profile manager:

1. On the Profile Manager window, select **Profile Manager→Subscribers**. The Subscribers window opens (Figure 49).
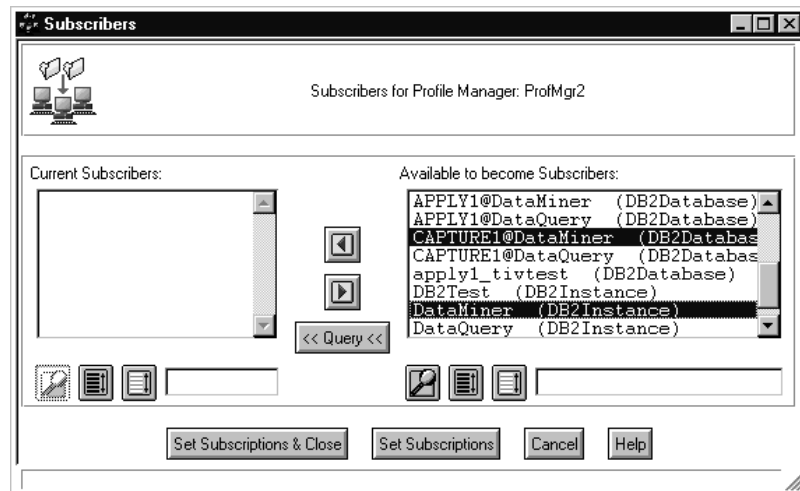
Figure 49. The Subscribers Window

2. In the **Available to become Subscribers** box, select:
   - Any DB2 instances, DB2 databases, DB2 partitions, DB2 partition groups, or managed nodes that you plan to monitor
   - Any profile managers to which you want to distribute the Sentry profile that you created
3. Click on the ◀ push button. The selected subscribers move to the **Current Subscribers** box.

4. Click on the **Set Subscriptions & Close** push button. The new subscribers' icons are displayed in the Profile Manager window.

Step 4: Select a Monitoring Source; Specify Arguments as Needed

You begin this step with the Profile Manager window still open.

To select a monitoring source and specify arguments:

1. In the Profile Manager window, select the Sentry profile's icon. The Sentry Profile Properties window opens (Figure 50 on page 70).

Setting Up Your Own Monitors
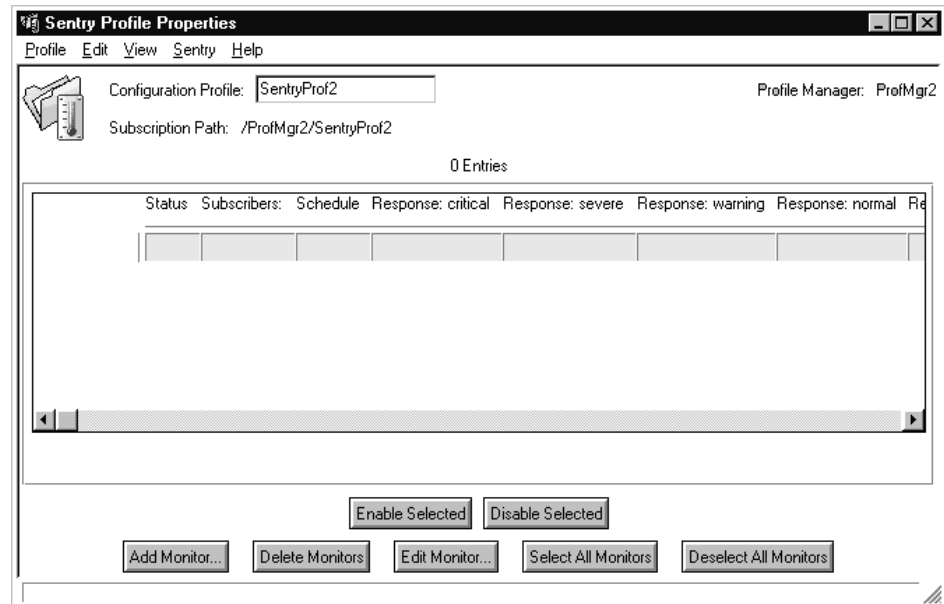
Figure 50. The Sentry Profile Properties Window

2. Close the Profile Manager window.
3. Click on the **Add Monitor** push button. The Add Monitor to Tivoli/Sentry Profile window opens (Figure 51).
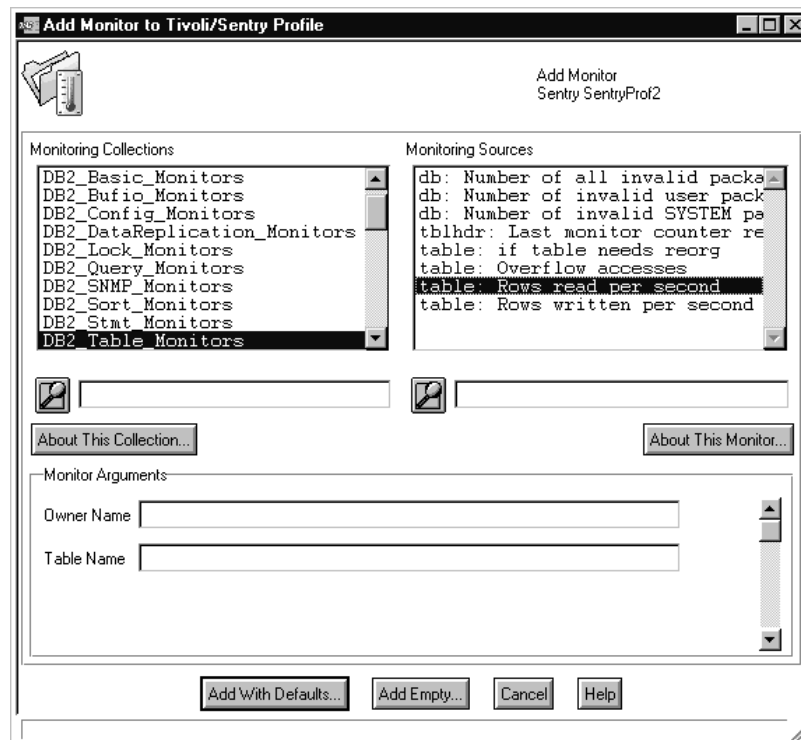
Figure 51. The Add Monitor to Tivoli/Sentry Profile Window

4. In the **Monitoring Collections** box, select a monitoring collection that contains a monitoring source that you want to implement as a monitor.

5. In the **Monitoring Sources** box, select the monitoring source.

   Table 6 lists the monitoring source prefixes and the type of endpoint for each prefix.

Table 6. Monitoring Sources Prefixes and Endpoints

Prefix	*Monitoring Source Endpoints					
	DB	DB Part.	DB Part. Grp.	Inst. (N)	Inst. (P)	Man. Node
conn:	✔	✔				
db:	✔	✔				
dbms:	✔	✔		✔		
dbp:		✔				
dbpg:			✔			
tbsp:	✔	✔				
table:	✔	✔				
user:	✔	✔				
None						✔

*Monitoring Source Endpoints
DB= database, DB Part. = DB2 Partition, DB Part. Grp. = DB2 Partition Group, Inst. (N) = Instance (nonpartitioned database server), Inst. (P) = Instance (partitioned database server), Man. Node = Managed Node

Tip: You might want to put all of the monitoring sources that are to be distributed to the same endpoint in one profile because the distribute action is per profile.

6. If the **Monitor Arguments** box displays fields for arguments that identify the object that the monitor will query, type the arguments in these fields.

   Note: The arguments are case sensitive: Owner names and table names are uppercase, instance names are lower case, and application IDs are mixed case.

7. Open the Edit Sentry Monitor window (Figure 52 on page 72) by clicking on one of the following push buttons:

   Add With Defaults
   Click on this push button if you selected a monitoring source that was implemented as a default monitor. The Edit Sentry Monitor window shows the monitor's default configuration. You can retain this configuration, or you can change it in "Step 5: Specify Response Levels, the Condition for Triggering a Response, and the Nature of the Response" on page 72 and "Step 6: Schedule the Monitor" on page 74.

   Add Empty
   Click on this push button if you selected a monitoring source for a monitor that you want to define.
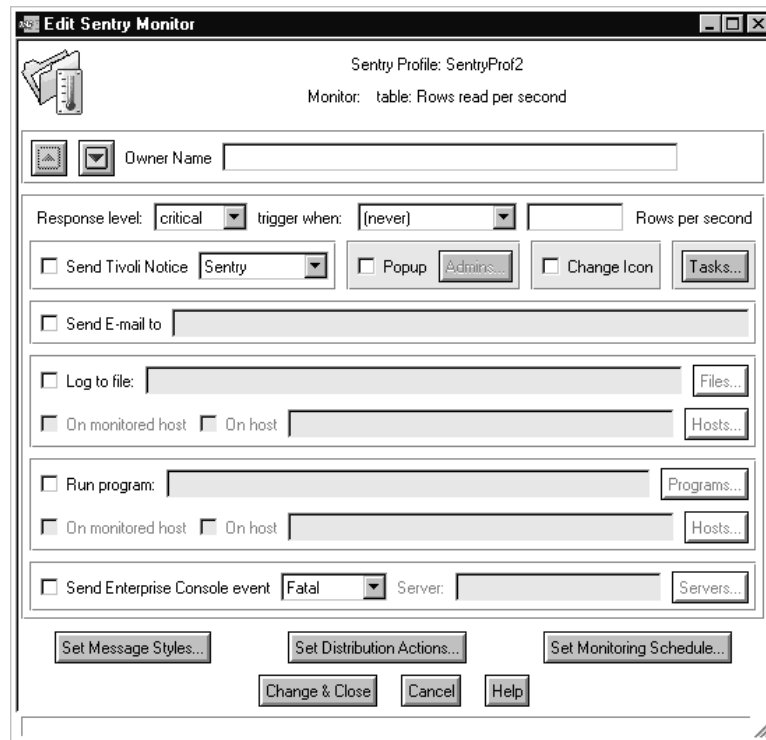
Figure 52. The Edit Sentry Monitor Window

Step 5: Specify Response Levels, the Condition for Triggering a Response, and the Nature of the Response

Sentry can trigger one to five responses to values returned by a monitor. For example, Sentry can trigger one sort of response when a returned value indicates that a critical problem has occurred, and another sort when a returned value indicates that conditions are normal. The occasions on which a response is possible—occasions such as a critical problem and normal conditions—are called *response levels*.

Some response levels are reached only when a returned value meets a threshold that you define to Sentry. At all response levels, Sentry can trigger a response consisting of one to eight actions; for example, sending a notice to a notice group and running a task in a task library.

You begin this step with the Sentry Profile Properties and Edit Sentry Monitor windows still open.

To specify response levels, the condition for triggering a response, and the nature of the response:

1. In the **Response level** box, click on the down arrow to display the available response levels and select one. The response levels are:

   critical

   > The returned value meets a threshold that indicates that an exceedingly serious problem has arisen.

severe

> The returned value meets a threshold that indicates that a serious, but not critical, problem has arisen.

warning

> The returned value meets a threshold that indicates that a problem important enough to warrant an alert has arisen.

normal

> The returned value fails to meet thresholds established for the critical, severe, or warning response levels. If no thresholds have been established for these levels, then every returned value triggers a response action.

always

> A response is triggered whenever the monitor returns a value, even if the value meets the threshold set for another response level.

2. If you selected a response level of critical, severe, or warning, specify the condition for triggering a response at that level:

    a. In the **trigger when** box, click on the down arrow to display the available threshold options, and select one. Most options require a value that you specify in step 2b.

    b. If the threshold option requires a value, type this value in the field to the right of the **trigger when** box.

3. Specify one or more actions to be taken when the condition that you just specified is met. The actions that you can choose are:

Send Tivoli Notice

> A notice is sent to a specified notice group. To receive the notice, you must be subscribed to the group.

Popup

> An alarm window pops up on the desktops of either all administrators (the default) or of the administrators that you specify.

Change Icon

> The icon in an indicator collection changes its state to reflect the response level.

Task   A task specified in any task library is performed.

> **Attention:** If you select **Task** as a response action for your monitor, the task that you want triggered must be stored in a task library that's installed at the same managed resource at which the monitor operates. DB2 ECC task libraries, however, are installed on the TMR server. Because the DB2 ECC monitors might not operate on the TMR server, but on managed resources on a client, the **Task** option might not be operable for them. Therefore, you must use the **Run program** option to trigger a DB2 ECC task.

Send E-mail to

> E-mail is sent to one or more addresses that you specify. If you specify multiple addresses, separate them by commas.

Log to file

> The information that the monitor returns is written to a log file. If the file doesn't exist, Sentry creates it. You must specify an existing directory for the file.

Run program

A program or script is launched. This response is executed under the same user ID and group ID as specified in the Sentry profile.

To specify that you want to trigger a DB2 ECC task as a response action:

a. Select the **Run program** check box.

b. In the field to the right of the **Run program** check box, type:

```
wecctlib eccTaskName argument1 argument2...
```

where:

wecctlib

Is a required command.

eccTaskName

Is the name of the task. For the names of the DB2 ECC tasks, see "Appendix C. DB2 ECC Task Libraries" on page 101 .

argument1 argument2 . . .

Are the values of the arguments for the task. For these values, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

Send Enterprise Console event

The information that the monitor returns is sent in the form of an event to a TME 10 Enterprise Console (TEC) server. You can specify this option only if both:

• The TEC is installed in your TME 10 environment.

• The DB2 ECC monitoring collections are registered with the TEC.

For more information, see "Registering DB2 ECC Monitoring Collections with the TME 10 Enterprise Console" on page 81.

Step 6: Schedule the Monitor

You can schedule the monitor to capture information at specific times. As a default, Tivoli sets all monitors to run once an hour. You can override this default in any of these ways:

• By setting a new default schedule for all monitors in a Sentry profile.

• By setting a different schedule for each monitor in the profile.

• By setting a default schedule for the profile and individual schedules for specific monitors in the profile.

Schedules for monitors override profile-wide schedules. Therefore, if you set a default schedule for profile X, and individual schedules for monitors A and B in profile X, monitors A and B will run according to these individual schedules, and all other monitors in profile X will run according to the default schedule.

If you want to use Tivoli's hourly default, skip to "Step 7: Set a Default User ID For a Sentry Profile with Any DB2 ECC Monitors" on page 77. Otherwise, proceed with this step.

You begin this step with the Profile Manager, Sentry Profile Properties, and Edit Sentry Monitor windows still open.

To schedule the monitor:

1. Open the Set Monitoring Schedule window (Figure 53):

   • If you plan to set a schedule for this monitor, click on the **Set Monitoring Schedule** push button in the Edit Sentry Monitor window.

   • If you plan to set a default schedule for the monitor's profile, select **Edit→Set Default Schedule** in the Sentry Profile Properties window.
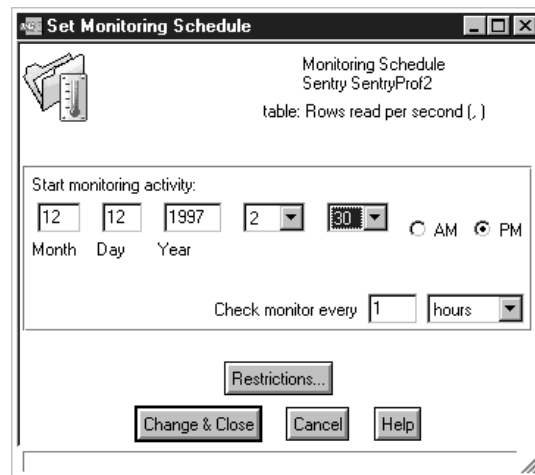
Figure 53. The Set Monitoring Schedule Window

2. Indicate when you want the monitor to start capturing information:

   • If you want the monitor to start when it is distributed, do not specify any values in the **Start monitoring activity** group box.

   • If you want the monitor to start at a specific point in time:

      a. In the **Month**, **Day**, and **Year** fields, specify the date on which you want the monitor to start.

      b. In the boxes beside the **Year** field, specify the hour and the minute on which you want the monitor to start. Then click on either the **AM** radio button or the **PM** radio button.

3. Specify that the monitor is to check the object that it's monitoring *n* times per *time span*, where:

   n   Is an integer that you type in the **Check monitor every** field.

   time span
            Is a value (**minutes**, **hours**, **days**, **weeks**, or **months**) that you specify to the right of the field.

4. Accept the schedule or restrict it:

   • If you're satisfied with the schedule, click on the **Set & Close** push button.

   • If you want to restrict it to certain hours or days:

      a. Click on the **Restrictions** push button. The Monitoring Schedule Restrictions window opens (Figure 54 on page 76).
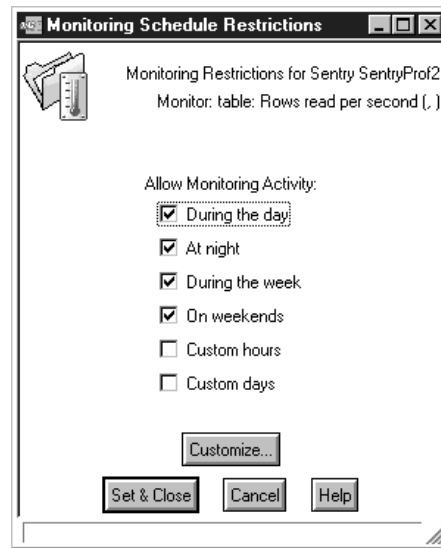
Figure 54. The Monitoring Schedule Restrictions Window

    b. If you want the monitor to run throughout the day or night during hours that Sentry determines:

      1) Optional: Find out what hours Sentry has set for daytime operation and nighttime operation:

        a) Click on the **Customize** push button. The Customize Monitoring Schedule Restrictions window opens (Figure 55 on page 77).

        b) In the Customize Monitoring Schedule Restrictions window, note the hours in the **AM** and **PM** boxes beside the **Daytime hours** and **Nighttime hours** labels. These hours mark the time spans set for the **During the day** and **At night** options on the Monitoring Schedule Restrictions window.

        c) Click on **Cancel** to close the Customize Monitoring Schedule Restrictions window.

      2) If you want the monitor to run during hours set by Sentry, select one or both of the following check boxes:

        – **During the day**

        – **At night**

    c. If you want the monitor to run on days set by Sentry, select one or both of the following check boxes:

      – **During the week**

      – **Weekends**

5. Accept the schedule or customize it:

    • If you're satisfied with the restricted schedule, click on the **Set & Close** push button.

    • If you want to customize the schedule:

      a. Select a check box for a time span:

        – If you want the monitor to run from a specific hour a.m. through a specific hour p.m., select the **During the day** check box.

        – If you want the monitor to run from a specific hour p.m. through a specific hour a.m., select the **At Night** check box.

- – If you want the monitor to run from one to four weekdays, select the **During the week** check box.
- – If you want the monitor to run either Saturday or Sunday, select the **On weekends** check box.
- – If you want the monitor to run for one or more hours a.m., or one or more hours p.m., select the **Custom hours** box.
- – If you want the monitor to run during the week and weekend, select the **Custom days** box.

    b. Customize the schedule:

       1) Click on the **Customize** push button. The Customize Monitoring Schedule Restrictions window opens.
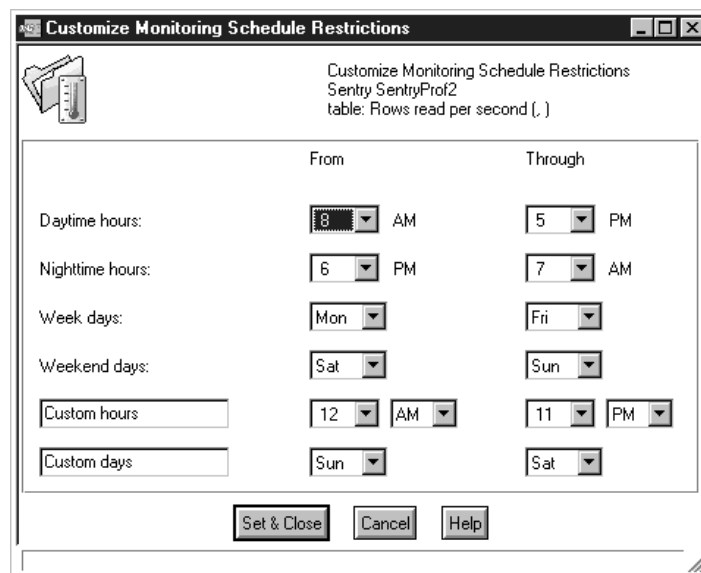
Figure 55. The Customize Monitoring Schedule Restrictions Window

       2) In the boxes in the **From** and **Through** columns, specify the hours or days that you want. For example, suppose that in step 5 on page 76 you selected the **During the day** check box in preparation for scheduling the monitor to run from 3 a.m. through 2 p.m. In the Customize Monitoring Schedule Restrictions window, you would select **3** and **2** in the **AM** and **PM** boxes for the **Daytime hours** field.

    c. Click on the **Set & Close** push button.

    d. In the Monitoring Schedule Restrictions window, click on the **Set & Close** push button.

6. In the Edit Sentry Monitor window, click on the **Change & Close** push button. Both this window and the Add Monitor to Tivoli/Sentry Profile window close.

Step 7: Set a Default User ID For a Sentry Profile with Any DB2 ECC Monitors

If you selected a monitoring source in any collection other than DB2_SNMP_Monitors, you *must* set the OS user ID, db2ecc, as a default for your Sentry profile. For information about db2ecc, see "Preparing a db2ecc Operating System (OS) User ID" on page 13.

Setting Up Your Own Monitors

You begin this step with the Sentry Profile Properties window still open.

To set the default user ID:

1. On the Sentry Profile Properties window, select **Edit**➔**Set User & Group ID**. The Edit Default Policies window opens (Figure 56).
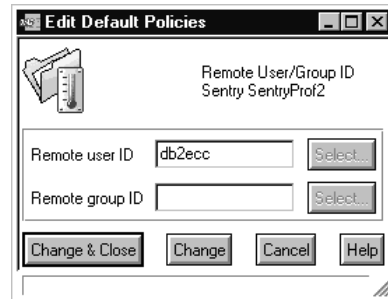
Figure 56. The Edit Default Policies Window

2. In the **Remote user ID** field, type:

   db2ecc

3. If the **Remote group ID** field is blank, leave it as is; if it contains a value, erase this value.

4. Click on **Change & Close**.

Step 8: Save and Distribute the Sentry Profile

You begin this step with the Sentry Profile Properties window still open.

To save and distribute the Sentry profile:

1. In the Sentry Profile Properties window, select **Profile**➔**Save**.

2. Select **Profile**➔**Distribute**. The Distribute Profile window opens (Figure 57 on page 79).
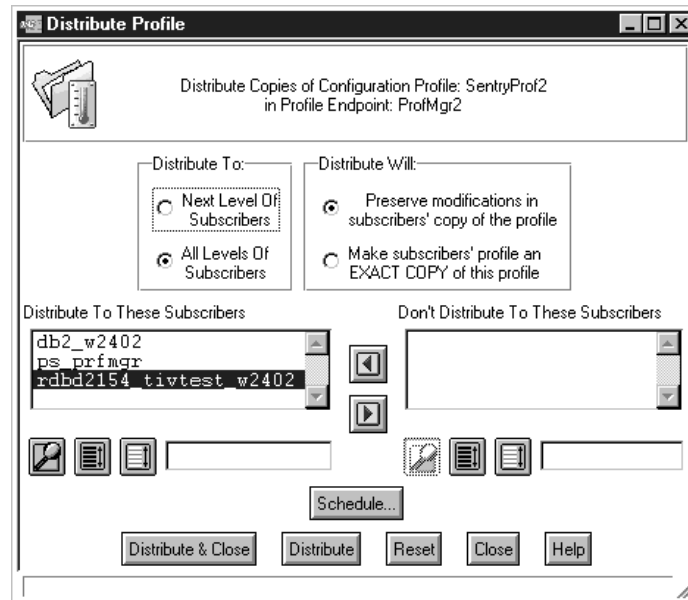
Figure 57. The Distribute Profile Window

3. In the **Distribute To** group box, select a radio button to specify which subscribers are to receive the profile:

   Next Level of Subscribers
   Distribute the profile only to the subscribers listed in the **Distribute To These Subscribers** box.

   All Levels of Subscribers
   Select this radio button if either of the following conditions exist:
   - One or more profile managers subscribe to the profile manager that contains the profile, and you want to distribute the profile to all subscribers in the distribution hierarchy.
   - A managed node is the only subscriber of the profile manager that contains the profile.

4. If the distribution hierarchy includes any profile managers at lower levels, and if any of these profile managers contain a copy of the profile that you're distributing, select one of these options in the **Distribute Will** group box:

   Preserve modifications in subscribers' copy of the profile
   Retains any differences that exist between the profile that you're distributing and the copies stored in the subscribing profile managers.

   Make subscribers' profile an EXACT COPY of this profile
   Causes your profile to overwrite the copies in the subscribing profile managers.

5. In the **Distribute to These Subscribers** box, select the subscribers that you do *not* want to distribute the Sentry profile to and then click on the ▣ push button.

   The selected subscribers move to the **Don't Distribute To These Subscribers**

   box.

6. Distribute the profile or set a time to distribute it:
   - To distribute the profile, click on **Distribute & Close**.

Setting Up Your Own Monitors

   • To set a time to distribute the profile:

      a. Click on the **Schedule** push button. The Add Scheduled Job window opens.

      b. Use the Add Scheduled Job window (Figure 40 on page 57) to schedule the distribution, and to specify that Sentry try to distribute the profile again if one or more attempts fail.

      c. On the Distribute Profile window, click on the **Distribute & Close** push button.

7. In the Sentry Profile Properties window, select **Profile→Close**.

Command Line

To set up your own DB2 ECC monitors from the command line:

1. Use the **wsetpr** command to specify the types of resources that you want to work with in your policy region.

   Example:

   To specify that the policy region TestRegion can contain the ProfileManager and DB2Instance resource types, enter the following command:

   ```
   wsetpr ProfileManager DB2Instance @PolicyRegion:TestRegion
   ```

2. Use the **wcrtprfmgr** command to create a profile manager.

   Example:

   To create a profile manager called ProfMgr2 in the policy region TestRegion, enter the following command:

   ```
   wcrtprfmgr @PolicyRegion:TestRegion ProfMgr2
   ```

3. Use the **wcrtprf** command to create a Sentry profile.

   Example:

   To create a Sentry profile called SentryProf2 in the ProfMgr2 profile manager, enter the following command:

   ```
   wcrtprf @ProfileManager:ProfMgr2 SentryProfile SentryProf2
   ```

4. Use the **wcrtsntcoll** command and the **wsetcoll** command to add an indicator collection for your Sentry profile.

   Example:

   To add an indicator collection called IndicatorCollection2 in the policy region TestRegion, enter the following command:

   ```
   wcrtsntcoll @PolicyRegion:TestRegion IndicatorCollection2
   ```

   To set the affiliation of the TestProf Sentry profile to the IndicatorCollection2 indicator collection, enter the following command:

   ```
   wsetcoll IndicatorCollection2 TestProf
   ```

5. Use the **wsub** command to add subscribers to the profile manager.

   Example:

   To add subscribers CAPTURE1@DataMiner (a DB2 database) and DataMiner (a DB2 instance) to the profile manager ProfMgr2, enter the following command:

   ```
   wsub @ProfileManager:ProfMgr2 @DB2Database:CAPTURE1@DataMiner
     @DB2Instance:DataMiner
   ```

6. Use the **waddmon** command to select a monitoring source, specify the arguments, specify the response to the monitor, and schedule the monitor.

   Example:

To add the monitor "table: Rows read per second" to the profile TestProf, log the response to the file mon_out.log, and schedule the monitor to run every five minutes, enter the following command:

```
waddmon DB2_Table_Monitors tableReorgChkOne -t "5 minutes" -d -c always
    -f "mon_out.log" TestProf
```

7. Use the **wsetsntid** command to set the default user ID for a Sentry profile with any non-SNMP monitors.

   Example:

   To set the OS user ID, db2ecc, for the TestProf Sentry profile with a default group ID of nobody, enter the following command:

   ```
   wsetsntid db2ecc '' TestProf
   ```

8. Use the **wdistrib** command to distribute the Sentry profile.

   Example:

   To distribute the default profile DB2_ECC-DBMSMonProfile to a DB2 instance endpoint with the label udbsinst@bandit, enter the following command:

   ```
   wdistrib @SentryProfile:DB2_ECC-DBMSMonProfile @DB2Instance:udbsinst@bandit
   ```

The syntax might vary for the commands in this section depending on the version of the TME framework that you are using. See the *TME 10 Framework Reference Manual* for more information about the syntax of the commands.

Determining Which Monitors Are Running against an Endpoint

From the desktop, you can quickly determine which monitors are running against a specific endpoint (managed node, DB2 instance, or DB2 database):

1. Locate and double-click on the icon for the endpoint. The window for that endpoint opens.
2. On the endpoint's window, look for an icon that represents a Sentry profile. If you don't find such an icon, it means that no monitors are running against the endpoint.
3. If you do find an icon for a Sentry profile, double-click on it. The Sentry Profile Properties window opens, listing the monitors that are running against the endpoint.

Registering DB2 ECC Monitoring Collections with the TME 10 Enterprise Console

If your TME 10 environment includes a TME 10 Enterprise Console (TEC), the TEC can make it easier to understand and act on information collected by your monitors. The TEC is a central receiving area for messages from various sources throughout your network—sources such as monitors, logfiles, and command lines. By bringing these messages together at the TEC, you improve your chances of detecting the real causes of problems.

You can configure the TEC to trigger preventative and corrective actions when it receives messages about potential or actual problems.

In TEC terminology, messages sent to the TEC are called *events*. Before your monitors can send events to the TEC, you must import predefined event classes from DB2 ECC to the TEC rule base. The classes are stored in Basic Recorder of Objects in C (BAROC) files; the DB2 ECC BAROC files are stored in the directory $BINDIR/TME/DB2ECC/monitoring/collections.

Registering with the TME 10 Enterprise Console

Sentry2_0_Base is the event class from which the DB2 ECC event classes are derived. DB2 ECC provides the DB2_Monitors class so that you can group all of the DB2 ECC events by specifying one event filter. Each monitoring collection and monitor also has its own event class.

The DB2 ECC event classes are organized in the following hierarchy:

Sentry2_0_Base
>The class for all Sentry events

>**DB2_Monitors**
>>The class for all DB2 events

>>**DB2_Agent_Monitors**
>>>The class for events from monitors in the DB2 agents and applications monitoring collection

>>**DB2_Basic_Monitors**
>>>The class for events from monitors in the DB2 basic monitoring collection

>>**DB2_Bufio_Monitors**
>>>The class for events from monitors in the DB2 buffer pool I/O monitoring collection

>>**DB2_Conf_Monitors**
>>>The class for events from monitors in the DB2 configuration monitoring collection

>>**DB2_DataRep_Monitors**
>>>The class for events from monitors in the DB2 data replication relational monitoring collection

>>**DB2_Lock_Monitors**
>>>The class for events from monitors in the DB2 locks and deadlocks monitoring collection

>>**DB2_Query_Monitors**
>>>The class for events from monitors in the DB2 query monitoring collection

>>**DB2_Sort_Monitors**
>>>The class for events from monitors in the DB2 sort work monitoring collection

>>**DB2_Stmt_Monitors**
>>>The class for events from monitors in the DB2 SQL statement activity monitoring collection

>>**DB2_Table_Monitors**
>>>The class for events from monitors in the DB2 table monitoring collection

>>**DB2_SNMP_Monitors**
>>>The class for events from monitors in the DB2 SNMP monitoring collection

Use the following script to import the DB2 ECC BAROC files to the TEC:

```
db2ecc_config_evtsvr.sh Rulebase CloneRulebase RulebasePath
```

where:

Rulebase
> Specifies the target rule base containing the DB2 ECC event classes. The default is DB2ECC.

CloneRulebase
> Specifies the rule base to be cloned. The default is the Default rule base.

RulebasePath
> Specifies the directory path of the target rule base to be created. You should choose a path other than $BINDIR/TME/DB2ECC.

The db2ecc_config_evtsvr.sh script defines the DB2 ECC event classes to the TEC and creates the DB2 ECC Event Group that you can assign to administrators to monitor DB2 ECC events. After you run the db2ecc_config_evtsvr.sh script, you can view the output in the /tmp/db2ecc_config_evtsvr.log file.

For more information, see the *Tivoli/Sentry User's Guide* or the *TME 10 Distributed Monitoring User's Guide*.

Chapter 7. Troubleshooting

This chapter provides explanations and remedies for problems that you might encounter when you install DB2 ECC, run DB2 ECC tasks, and use DB2 ECC monitors.

Installation Problems

Symptom: *When you select* **Desktop***➔***Install***➔***Install Product** *to open the Install Product window, you receive this error message:*

```
Something is wrong with the current installation media settings. The media
host must be connected, and the media directory needs to have a file
(CONTENTS.LST) which contains a list of patches present. Please select a
media directory. Also, note that you can use the ''Select Media...'' button
on the ''Product Install'' dialog, to change/inspect the current media
setting.
```

This message indicates that the current media setting isn't pointing to a directory containing a CONTENTS.LST file. On the window that contains the message, select **OK**. The Set Path to Tivoli Install Media window opens. Set the path for the install media to point to the DB2 ECC installation image as described in step 4 on page 16 .

Symptom: *You receive an error message saying that there was a media packet failure. The error log's path and file name are not displayed on the Install Product window.*

This symptom means that the Tivoli Management Region (TMR) server couldn't process an installation packet file. One reason for this symptom can be that the server didn't have enough space in the /tmp directory to contain the temporary files that Tivoli creates when DB2 ECC is installed. To resolve this problem, allocate more space to the /tmp directory or free additional space on the disk or file system.

Symptom: *The installation process fails and there is a message in the error log. For example:*

```
Fri Nov 7 16:05:20 1997 (18): 'iom_open' failed with code '67': ''
```

This message means that the Tivoli Management Region (TMR) server didn't have enough space in the /tmp directory to contain the temporary files that Tivoli creates when DB2 ECC is installed. To resolve this problem, allocate more space to the /tmp directory or free additional space on the disk or file system.

Installation Problems

Symptom: *You receive an error message saying that there was a media packet failure. In addition, the error log's path, file name, and last few entries are displayed in the Install Product window. As the following example of this display indicates, the entries clearly identify the problem or problems that occurred.*

```
Failure: the last few lines of the error log
(e:\Tivoli\shasta.db\tmp/DB2ECC_V1R2_DBafter.error):
+ ø 2 -ne 0 !
+ echo Cannot start monitoring service for db2ecc with password db2eccpw
+ echo Verify password and "Log on as a service" user right for db2ecc
+ do_cleanup
+ ø 1718480917.1.327#TMF_ManagedNode::Managed_Node# = 1718480917.2.7#TMF_
 Node::Managed_Node# !
```

The displayed log entries indicate that Tivoli detected one or more problems when it tried to install the DB file packet. For example, a db2ecc user might not be defined on a system, or a password for NT might be incorrect. Check the error log file for additional entries.

Symptom: *During the installation process, you receive an error. For example:*

```
Something went wrong:
Media Packet: "/tmp/scratch_dir/package/FILE3.PKT"
Error Line 187: Expected '=' assignment.
FATAL.
```

This message might indicate that you specified more than one DB2 instance object to create during the DB2 ECC installation. To resolve this problem:

1. Select **Cancel** to return to the Install Product window.
2. Select **Install Options** and enter the name of only one DB2 instance object to create on each managed node on which you install DB2 ECC.
3. Start the installation again.

Symptom: *During the installation process, a message appears in the Install Product window. For example:*

```
+do_fixuptlib ecc_tlib1 + do_loadtasklib /tivolifs/Tivoli/aix4-
r1/TME/DB2ECC/tasklib/ecc_tlib1.sed An instance named "beryl" of resource
"ManagedNode" was not found.
```

This message might indicate that the host name of the server is set to the full TCP/IP name. To resolve this problem, try setting the host name to the short TCP/IP name and rerun the installation.

This message might also indicate that the host name is different from the name that the Tivoli environment recognizes as the managed node. To find out how Tivoli named the managed node, type the following command after setting up the Tivoli environment:

```
wlookup -r ManagedNode -a
```

The response might look something like this:

```
Beryl       1821498885.2.7#TMF_ManagedNode::Managed_Node#
```

This response means that the TCP/IP host name of the machine is beryl, but the Tivoli managed node was created with the name Beryl. To resolve this problem, you can issue the following command to register beryl as a managed node:

```
wregister -i -r ManagedNode beryl 1821498885.2.7
```

and try the installation again. After the installation completes, you can issue the following command to unregister the entry:

```
wregister -u -r ManagedNode beryl
```

Symptom: *During the installation of DB2 ECC on an NT managed node, the installation process hangs and the error log shows an "awk" command as the last action to be issued.*

Use the pview.exe NT tool to examine the currently running processes. If the awk process is hung, use pview to kill the process. To resolve this problem, you might need to install the Tivoli patch 3.1-TMP-0015. Contact Tivoli support to request a copy of the patch. Follow the instructions in the readme file for the patch to ensure that the necessary files are updated on your NT system.

Symptom: *During the upgrading of DB2 ECC V1.1 to V1.2 on an NT managed node, you receive this error message:*

```
ecdb2ms5.EXE - Entry Point Not Found
The procedure entry point sqlbstpq@16 could not be located in the dynamic
link library DB2APP.dll
```

You can dismiss the message window and allow the upgrade to continue until completion. When the upgrade completes successfully, run the ECC_Stop_Monitoring_Agent and ECC_Start_Monitoring_Agent tasks to stop and restart the monitoring agent on each of the nodes.

Symptom: *During the upgrading of DB2 ECC V1.1 to V1.2 on an NT managed node, you receive an error message indicating that the specified service has been marked for deletion and the monitoring service for DB2 ECC cannot be started. In addition, the error log's path, file name, and last few entries are displayed in the Install Product window. As the following example of this display indicates, the entries clearly identify the problem that occurred.*

```
Failure: the last few lines of the error log
c:\Tivoli\fingy.stl.ibm.com.db\tmp/DB2ECC_V1R2_BIN_after.error): The specified
service has been marked for deletion.
+ ( 2 -ne 0 )
+ echo Cannot start monitoring service for db2ecc with password db2ecc
+ echo Verify password and "Log on as a service" user right for db2ecc
+ do_cleanup
```

```
+ rm -f c:/Tivoli/bin/w32-ix86/.installed/DB2ECC_V1R2_BIN

+ return 0
+ exit 12
```

This message and error log information indicate that the installation script could not start the DB2 ECC monitoring service because a delete request is pending on that NT machine. You should allow the upgrading of DB2 ECC to complete on all other nodes selected to be upgraded. Then reboot the NT machine with the problem and attempt the upgrade on that NT node again.

Administration Problems

Symptom: *When you are creating a DB2 instance object on an AIX or Solaris managed node from the desktop, you receive an error code of -1390.*

Make sure that the instance name is a valid instance and that the permission on the instance home directory is set to 755.

Symptom: *When you are creating a DB2 partition object, you receive the following error message:*

```
A communications error occurred: destination dispatcher unavailable
```

This message indicates that there is a DB2 instance defined on a managed node where the object dispatcher is not running. To resolve this problem, bring up the object dispatcher on the managed node and try to create the DB2 partition object again.

Task Library Problems

Symptom: *The TMR server is installed on an NT system. When you try to run the ECC_Apply_Start task or the ECC_Capture_Start task, you receive this error message:*

```
The name specified is not recognized as an internal or external command,
operable program or batch file.
```

Make sure that the Apply and Capture installation directories are included in the NT system environment variable path.

Symptom: *A job for a DB2 ECC task completes without returning a value.*

The reason might be that the job took longer than 60 seconds to complete. Sixty seconds is the default time for a Tivoli task to complete. This default is not long enough for certain tasks, such as ECC_Create_Sample_Database. To prevent or

resolve this problem, increase the value in the **Timeout** field of the Create Job window. For more information, see "Scheduling Tasks in the DB2 ECC Task Libraries" on page 52.

Symptom: *When you run a task from a Solaris managed node, the task does not execute and you receive an error message. For example:*

```
Tue Oct 14 10:01:06 1997(26): command exited with signal 0, core=FALSE
```

To resolve this problem, make sure that your LANG environment variable is set to C before starting the TME 10 desktop. If it is set to en_US (as it should be for an AIX managed node), tasks will not execute.

Symptom: *When you a run a task on an AIX or Solaris managed node, you receive an error message. For example:*

```
crispy (Managed Node): The task failed to execute. crispy (Managed Node):
Thu Oct 16 17:45:22 1997 (18): 'getgrnam' failed with code '2':'sysadm'
```

This message indicates that on the managed node crispy, the task is expected to execute as user ID db2ecc and group sysadm, but the db2ecc ID doesn't belong to the sysadm group or the sysadm group does not exist. To resolve this problem, you can do one of the following things:

- Change the db2ecc ID on the managed node so that its primary group is sysadm.
- If you are going to run DB2 ECC tasks on this managed node only and not on any other managed node:
  1. Find out what the primary group of db2ecc is on this managed node (for example, admgroup).
  2. Use the following **ecwset** commands in the $BINDIR/TME/DB2ECC/task_lib directory to set all DB2 ECC tasks to run as user db2ecc and group admgroup:
     ```
     ecwset1.sh db2ecc admgroup
     ecwset2.sh db2ecc admgroup
     ecwset3.sh db2ecc admgroup
     ```

Symptom: *Tasks fail to run from a monitor.*

A possible cause is that you did not select an appropriate type of endpoint for the given task or you did not specify the correct arguments. To help you diagnose the problem, direct the standard and error output from the task to a file with the following command:
```
wecctlib task_name arg1 .. argn 2>&1 >> file_name
```

Monitoring Problems

Symptom*: Monitors do not return values.*

If a Sentry profile's monitors do not return values after you distribute the profile, check the SentryStatus notice group for a message that explains why. If you don't find such a message, you might need to refresh the Sentry engine with the **wstopeng** command:

```
wstopeng managed_node
```

For more information, see the *Tivoli/Sentry User's Guide*, the *TME 10 Distributed Monitoring User's Guide*, or the wstopeng man page entry.

Refreshing the Sentry engine doesn't work if the Tivoli object dispatcher (oserv) isn't running on the managed node at which the profile that you distributed is supposed to be implemented. If oserv isn't running, the communication server ecdb2ipc generates a CORE file. To remedy this situation:

1. Enter the command:

   ```
   odadmin odlist
   ```

   Tivoli returns the oserv identifier.

2. Restart oserv by entering the command:

   ```
   odadmin start oserv_identifier
   ```

3. Terminate and restart the ecdb2ipc process on the managed node by running the following DB2 ECC tasks:

   a. Run the ECC_Stop_Monitoring_Agent task in the DB2_ECC-AdminTasks library.

   b. Run the ECC_Start_Monitoring_Agent task in the DB2_ECC-AdminTasks library.

   The endpoint for both tasks is the managed node where Sentry is not returning values.

Symptom*: Monitors running on a Solaris managed node do not return values.*

If a Sentry profile's monitors do not return values after you distribute the profile to a Solaris managed node or to a DB2 object on a Solaris managed node, the monitoring server might have stopped running. When the monitoring server stops running on a Solaris managed node, you must manually start it. To manually start the monitoring server, do one of the following things:

- Run the ECC_Start_Monitoring_Agent task in the DB2_ECC-AdminTasks library. The endpoint for this task is the Solaris managed node.

- As root, type:

```
nohup $BINDIR/TME/DB2ECC/monitoring/scripts/ecdb2ipc.sh > /dev/null 2>&1 &
```

Symptom*: When a monitor is being executed on an AIX or Solaris managed node, you receive the following error message:*

```
Fatal Exception: ISystemCallFail
Location ../src/UnixConnection.cpp
Description : ECMG003 : Error reading message Permission denied
```

A possible cause is that the client does not have the correct authorization, that is the monitor was not submitted by the db2ecc user ID. For more information, see "Step 7: Set a Default User ID For a Sentry Profile with Any DB2 ECC Monitors" on page 77 .

Symptom: *When a monitor is being executed on an NT managed node, you receive one of the following error messages:*

```
Access to service control manager was denied.
```

```
Cannot open Client Event to send monitor request to DB2 ECC Monitoring
Service. Verify that you have the correct authorization to run the monitor.
```

A possible cause is that the client does not have the correct authorization, that is the monitor was not submitted by the db2ecc user ID. For more information, see "Step 7: Set a Default User ID For a Sentry Profile with Any DB2 ECC Monitors" on page 77 .

Symptom: *The "DB2 database state" monitor returns a value of 0 even if allowable values are 1-5.*

A value of 0 indicates that this monitor did not execute in the allowable time. This value could mean that the SNMP process (snmpd) is not operational, that the query took longer than the time allowed for execution, or that too many DB2 databases are defined on the managed node on which the monitor operates.

Symptom: *Monitor returns a value of -999.*

This value indicates that the monitor did not return a valid value. Review any additional comments returned by Sentry.

Symptom: *Monitor returns a value of -999 and you receive an error message. For example:*

```
Invalid Endpoint Type
Please verify that you have distributed this monitor to the correct endpoint
Endpoint Class distributed = DB2Instance
Endpoint ID distributed = 1047464517.1.900
```

Verify that you have specified a valid endpoint for the monitor. (See Table 6 on page 71 for a list of monitoring source prefixes and valid endpoint types). If you

have specified a valid endpoint, another possible cause for this message is that the monitoring agent has lost its connection to the Tivoli database. To resolve this problem, perform the following steps on the managed node:

1. Run the ECC_Stop_Monitoring_Agent task in the DB2_ECC-AdminTasks task library to stop the monitoring agent on the managed node.
2. Run the ECC_Start_Monitoring_Agent task in the DB2_ECC-AdminTasks task library to restart the monitoring agent on the managed node.

If the problem persists, restart the object dispatcher on the TME 10 server.

Symptom: *The "DB2 database state" monitor returns a -999 error or an incorrect value on OS/2.*

DB2 database names must be unique across OS/2 instances on the same drive. This symptom means that the DB2 SNMP subagent returned insufficient information needed by the "DB2 database state" monitor to uniquely identify a database with the same name defined on different instances on one OS/2 system. This problem is due to a known limitation of the DB2 SNMP subagent.

If your database names are not unique for all instances installed on the same drive, make them unique.

If your database name is unique, enter the following arguments when you define a "DB2 database state" monitor:

1. In the **instance name** field, type the drive letter where the DB2 instance is installed. For example, if the instance is installed on drive d, type d.
2. In the **database name** field, type the database's unique name.

Symptom: *Numeric monitor returns a value of 32767 or -32767.*

This value indicates that the monitor did not receive valid data from DB2. If the monitor normally returns a positive number to represent valid data, the monitor returns a value of 32767 for invalid data. If the monitor normally returns 0 or a negative number to represent valid data, the monitor returns a value of -32767 for invalid data.

Symptom: *You are running a monitor against a DB2 partition group endpoint and the returned values do not match the group members of the DB2 partition group. For example, you might be running the "dbpg: Status of nodes in a partitioned database server" monitor. The monitor returns values for nodes 0 and 1, but you changed the DB2 partition group object to contain nodes 1 and 2.*

If you defined a DB2 partition group object that you are running monitors against, and you change the members of the DB2 partition group object, the DB2 ECC monitoring agent might not pick up the changed definition right away. The

monitoring agent refreshes object definitions every 60 minutes. To resolve this problem, run the ECC_Stop_Monitoring_Agent task and then the ECC_Start_Monitoring_Agent task on the node where the DB2 instance resides.

Appendix A. Uninstalling DB2 ECC

The db2eccuninst.sh uninstall script removes DB2 ECC resource definitions from the Tivoli database and the binary files distributed by DB2 ECC.

On AIX or Solaris, you must have root authorization for the uninstall script to run correctly. On Windows NT, you must log on as the Administrator to run the uninstall script.

When you run the uninstall shell script, it removes:
- DB2 managed resource definitions
- DB2 ECC notice groups
- DB2 ECC monitoring capability collections
- DB2 ECC monitoring services or daemons from the current node
- DB2 ECC binary files

The following steps will remove DB2 ECC from a TMR. The steps include a combination of manual steps and executing a script on each managed node from which you want to uninstall DB2 ECC.

To uninstall DB2 ECC:
1. Manually delete all objects in the DB2_ECC-DefaultPolicyRegion. (This step is not performed by the uninstall shell script, so you must manually delete the objects using the **edit** menu fom the Tivoli desktop.)
2. Manually delete the DB2_ECC-DefaultPolicyRegion:
   - If the DB2_ECC-DefaultPolicyRegion was moved to an administrator's desktop, manually delete it from the desktop.
   - If the DB2_ECC-DefaultPolicyRegion was not moved, delete it by entering the following command at a command line:

     ```
     wdelpr /Regions/DB2_ECC-DefaultPolicyRegion
     ```
3. Manually delete any remaining DB2 ECC objects from the Tivoli desktop, including any profiles that you created that contain DB2 ECC monitors.
4. Remove the DB2Instance, DB2Database, DB2Partition, and DB2PartitionGroup resource types as managed resources from all policy regions in the TMR.
5. Remove all DB2 instance, DB2 database, DB2 partition, and DB2 partition group objects from all policy regions.
6. On each managed node where DB2 ECC is installed, initialize the Tivoli environment and then run the uninstall script. Do not start the process on another managed node until you receive the message that DB2 ECC is uninstalled on the current managed node. To initialize the Tivoli environment and run the uninstall script:
   - On AIX or Solaris, as root, enter the following commands:

     ```
     . /etc/Tivoli/setup_env.sh
     $BINDIR/bin/db2eccuninst.sh
     ```
   - On Windows NT 3.51 or 4.0, log on as a member of the Administrators group and enter the following command:

     ```
     %SystemRoot%\system32\drivers\etc\Tivoili\setup_env.cmd
     sh %BINDIR%\bin\db2eccuninst.sh
     ```

Uninstalling DB2 ECC

When you run the uninstall script, it displays a list of actions that you must complete before DB2 ECC can be removed. The script also displays a prompt, asking if you want to continue.

7. To continue, enter `Y`. The DB2 ECC resource definitions are removed. The script displays another prompt, asking if you want to remove the DB2 ECC binary files.

8. Enter `Y` to remove the binary files or `N` to keep them.

DB2 ECC is now uninstalled.

Attention: Most of the files (and soft links for AIX and Solaris) are removed by running db2eccuninst.sh. The exception is the db2eccuninst.sh uninstall script itself.

If you no longer want to manage DB2 resources on a particular managed node in your TMR, you can run the uninstall script to remove only the DB2 ECC binary files from that managed node.

To remove the DB2 ECC binary files from a managed node:

1. Remove all references to DB2 ECC objects on the managed node, such as DB2 instance objects or task endpoints.

2. Initialize the Tivoli environment and then run the uninstall script. Do not start the process on another managed node until you receive the message that DB2 ECC is uninstalled on the current managed node. To initialize the Tivoli environment and run the uninstall script:

   - On AIX or Solaris, as root, enter the following commands:

     ```
     . /etc/Tivoli/setup_env.sh
     $BINDIR/bin/db2eccuninst.sh -b
     ```

   - On Windows NT 3.51 or 4.0, log on as a member of the Administrators group and enter the following command:

     ```
     %SystemRoot%\system32\drivers\etc\Tivoili\setup_env.cmd
     sh %BINDIR%\bin\db2eccuninst.sh -b
     ```

The DB2 ECC binary files are now uninstalled.

Appendix B. DB2 ECC Administrator Roles

This appendix contains a list of the DB2 ECC actions and tasks and the authorization roles required to perform them.

The authorization roles that apply to DB2 ECC are super, senior, admin, and user. The super and senior roles are for activities such as creating and deleting DB2 databases, and running all DB2 ECC tasks. The admin role is for activities such as editing monitors and distributing Sentry profiles, and running selected DB2 ECC tasks. The user role limits an administrator to activities such as modifying DB2 instance properties and running certain basic DB2 ECC tasks.

Table 7 lists the roles that are required to perform each DB2 ECC action. When granting a role in Table 7 to an administrator, you can define it either as a TMR role or as a resource role associated with the DB2 ECC default policy region.

Table 7. Administrator Roles for Tivoli Actions

Action	User	Admin	Senior	Super	Any
DB2Instance					
Create			✔	✔	
Delete			✔	✔	
Launch DB2CC					✔
Startup		✔	✔	✔	
Shutdown		✔	✔	✔	
Other					✔
DB2Database					
Create			✔	✔	
Delete			✔	✔	
Launch DB2CC					✔
Other					✔
DB2Partition					
Create			✔	✔	
Delete			✔	✔	
Startup		✔	✔	✔	
Shutdown		✔	✔	✔	
Other					✔
DB2PartitionGroup					
Create			✔	✔	
Delete			✔	✔	
Startup		✔	✔	✔	
Shutdown		✔	✔	✔	
Change Members			✔	✔	
Other					✔
Indicator Collection					

DB2 ECC Administrator Roles

Table 7. Administrator Roles for Tivoli Actions (continued)

Action	User	Admin	Senior	Super	Any
Create		✔			
View		✔			
Proxy					
Create			✔		
Set Properties			✔		
Set Monitor Filters			✔		
Sentry Profiles					
Distribute		✔			
Add Monitor		✔			
Edit Monitor		✔			
Set Defaults		✔			
Sentry Profile Manager					
Create		✔			
Delete		✔			

Table 8 lists the TMR roles that are required to perform the DB2 ECC tasks.

Table 8. Administrator Roles for Task Library Tasks

Action	User	Admin	Senior	Super
ECC_Activate_Database			✔	✔
ECC_Alter_Bufferpool			✔	✔
ECC_Apply_Start		✔	✔	✔
ECC_Apply_Stop		✔	✔	✔
ECC_Backup_Database			✔	✔
ECC_Broadcast_Message		✔	✔	✔
ECC_Capture_Get_Log_Seq		✔	✔	✔
ECC_Capture_Prune		✔	✔	✔
ECC_Capture_Reinit		✔	✔	✔
ECC_Capture_Resume		✔	✔	✔
ECC_Capture_Start		✔	✔	✔
ECC_Capture_Stop		✔	✔	✔
ECC_Capture_Suspend		✔	✔	✔
ECC_Create_Admin_Server			✔	✔
ECC_Create_Explain_Tables	✔	✔	✔	✔
ECC_Create_Sample_Database			✔	✔
ECC_Deactivate_Database			✔	✔
ECC_Drop_Admin_Server			✔	✔
ECC_Drop_Sample_Database			✔	✔
ECC_Force_All_Applications			✔	✔

Table 8. Administrator Roles for Task Library Tasks (continued)

Action	User	Admin	Senior	Super
ECC_Force_Applications			✔	✔
ECC_Get_Admin_Configuration			✔	✔
ECC_Get_Admin_Server			✔	✔
ECC_Get_Database_Configuration			✔	✔
ECC_Get_Database_Manager_Configuration			✔	✔
ECC_Invoke_Stored_Procedure	✔	✔	✔	✔
ECC_List_Applications			✔	✔
ECC_List_Backup_Recovery_File	✔	✔	✔	✔
ECC_List_DCS_Applications			✔	✔
ECC_List_Node_Directory			✔	✔
ECC_List_Profile_Registry			✔	✔
ECC_Prune_Recovery_History_File		✔	✔	✔
ECC_Quiesce_Tablespaces		✔	✔	✔
ECC_Rebind_All_Packages		✔	✔	✔
ECC_Rebind_Package		✔	✔	✔
ECC_Reorganize_Table		✔	✔	✔
ECC_Reorgchk		✔	✔	✔
ECC_Reset_Admin_Configuration			✔	✔
ECC_Reset_Counters			✔	✔
ECC_Reset_Database_Configuration			✔	✔
ECC_Reset_Database_Manager_ Configuration			✔	✔
ECC_Restart_Database	✔	✔	✔	✔
ECC_Run_Statistics		✔	✔	✔
ECC_Send_Notice	✔		✔	✔
ECC_Set_Up_Monitoring_Agent			✔	✔
ECC_Set_Admin_Server			✔	✔
ECC_Set_Profile_Registry			✔	✔
ECC_Start_Admin_Server			✔	✔
ECC_Start_DB2			✔	✔
ECC_Start_DB2SNMP_Agent			✔	✔
ECC_Start_DB2_NT_Security			✔	✔
ECC_Start_Monitoring_Agent			✔	✔
ECC_Stop_Admin_Server			✔	✔
ECC_Stop_DB2			✔	✔
ECC_Stop_DB2_NT_Security			✔	✔
ECC_Stop_DB2SNMP_Agent			✔	✔
ECC_Stop_Monitoring_Agent			✔	✔

DB2 ECC Administrator Roles

Table 8. Administrator Roles for Task Library Tasks (continued)

Action	User	Admin	Senior	Super
ECC_Update_Admin_Configuration			✔	✔
ECC_Update_Database_Configuration			✔	✔
ECC_Update_Database_Manager_ Configuration			✔	✔

Appendix C. DB2 ECC Task Libraries

The DB2 ECC task libraries are collections of predefined management tasks, such as backing up databases and running statistics. Running, scheduling, or automatically executing these predefined tasks helps ease overall management workload in a complex environment by carrying out actions against all subscribed task endpoints without having to define the task for each endpoint separately.

The tasks are grouped into three libraries based on function:

Monitor and Administration Tasks
> Contains tasks used for DB2 ECC monitoring and administration.

DB2 Database Tasks
> Contains tasks used with DB2 and the database manager.

Data Replication Tasks
> Contains tasks used for data replication with IBM Replication (previously DPROPR).

For detailed descriptions of the tasks, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

The following tables list the tasks in each DB2 ECC task library and which endpoints they run on.

Table 9. Monitoring and Administration Task Library and Task Endpoints

Task	Description	Task Endpoints*				
		DB	**DB Part.**	**DB Part. Grp.**	**Inst.**	**Man. Node**
ECC_Broadcast_Message	Broadcast a message to all TME desktops	✔	✔	✔	✔	✔
ECC_Reset_Counters	Reset the ECC counters for a DB2 instance		✔		✔	
ECC_Send_Notice	Send a notice to all TME administrators	✔	✔	✔	✔	✔
ECC_Set_Up_Monitoring_Agent	Set up the monitoring agent	✔	✔	✔	✔	✔
ECC_Start_DB2SNMP_Agent	Start the DB2 SNMP agent	✔	✔	✔	✔	
ECC_Start_Monitoring_Agent	Start the ECC monitoring agent	✔	✔	✔	✔	✔
ECC_Stop_DB2SNMP_Agent	Stop a DB2 SNMP daemon	✔	✔	✔	✔	✔
ECC_Stop_Monitoring_Agent	Stop the ECC monitoring agent	✔	✔	✔	✔	✔

*Task Endpoints
DB= database, DB Part. = DB2 partition, DB Part. Grp. = DB2 partition group, Inst. = instance (partitioned or nonpartitioned), Man. Node = managed node

DB2 ECC Task Libraries

Table 10. DB2 Database Task Library and Task Endpoints

Task	Description	DB	DB Part.	DB Part. Grp.	Inst.	Man. Node
ECC_Activate_Database	Activate a database	✔	✔	✔		
ECC_Alter_Bufferpool	Modify the size of the bufferpool	✔	✔	✔		
ECC_Backup_Database	Create a backup of a DB2 database	✔	✔	✔		
ECC_Create_Admin_Server	Create the DB2 administration server	✔			✔	
ECC_Create_Explain_Tables	Create explain tables	✔	✔	✔		
ECC_Create_Sample_Database	Create a DB2 sample database	✔	✔	✔	✔	
ECC_Deactivate_Database	Deactivate the database	✔	✔	✔		
ECC_Drop_Admin_Server	Delete the DB2 administration server instance	✔			✔	
ECC_Drop_Sample_Database	Drop the DB2 sample database	✔	✔	✔	✔	
ECC_Force_All_Applications	Force all DB2 applications off the system	✔	✔	✔	✔	
ECC_Force_Applications	Force specific DB2 applications off the system	✔	✔	✔	✔	
ECC_Get_Admin_ Configuration	Get the DB2 administration configuration	✔	✔		✔	
ECC_Get_Admin_Server_Instance	Get the DB2 administration server instance name	✔	✔	✔	✔	
ECC_Get_Database_Configuration	Return the values of individual entries in the database configuration file	✔	✔	✔		
ECC_Get_Database_Manager_ Configuration	Return the values of individual entries in the database manager configuration file	✔	✔		✔	
ECC_Invoke_Stored_Procedure	Invoke a DB2 stored procedure	✔	✔	✔		
ECC_List_Applications	List the DB2 applications	✔	✔	✔	✔	
ECC_List_Backup_Recovery_File	List the DB2 backup and recovery history file	✔	✔	✔		
ECC_List_DCS_Applications	List the DB2 DCS applications	✔	✔	✔	✔	
ECC_List_Node_Directory	List the contents of the node directory	✔	✔		✔	
ECC_Prune_Recovery _ History_File	Prune the DB2 recovery history file	✔	✔	✔		
ECC_Quiesce_Tablespaces	Quiesce DB2 table spaces	✔	✔	✔		
ECC_Rebind_All_Packages	Rebind all DB2 packages	✔	✔	✔		
ECC_Rebind_Package	Rebind a specific DB2 package	✔	✔	✔		
ECC_Reorganize_Table	Reorganize a DB2 table	✔	✔	✔		

Table 10. DB2 Database Task Library and Task Endpoints (continued)

Task	Description	Task Endpoints*				
		DB	**DB Part.**	**DB Part. Grp.**	**Inst.**	**Man. Node**
ECC_Reorgchk	Run the DB2 REORGCHK command	✔	✔	✔		
ECC_Reset_Admin_Configuration	Reset the parameters in the database manager configuration file, relevant to the DB2 administration server, to the system defaults	✔	✔		✔	
ECC_Reset_Database_ Configuration	Reset the configuration of a specific database to the system defaults	✔	✔	✔		
ECC_Reset_Database_ Manager_Configuration	Reset the parameters in the database manager configuration file to the system defaults	✔	✔		✔	
ECC_Restart_Database	Restart a DB2 database	✔	✔	✔		
ECC_Run_Statistics	Run DB2 statistics	✔	✔	✔		
ECC_Set_Admin_Server_ID	Establish or modify the user account associated with the DB2 administration server instance	✔	✔	✔	✔	
ECC_Start_Admin_Server	Start the DB2 administration server	✔	✔	✔	✔	
ECC_Start_DB2	Start DB2	✔	✔	✔	✔	
ECC_Start_DB2_NT_Security	Start the DB2 security service on a Windows NT system	✔			✔	✔
ECC_Stop_Admin_Server	Stop the DB2 administration server	✔	✔	✔	✔	
ECC_Stop_DB2	Stop DB2	✔	✔	✔	✔	
ECC_Stop_DB2_NT_Security	Stop the DB2 security service on a Windows NT system	✔			✔	✔
ECC_Update_Admin_Configuration	Modify individual entries in the database manager configuration file that are relevant to the DB2 administration server	✔	✔		✔	
ECC_Update_Database_ Configuration	Update the DB2 database configuration	✔	✔	✔		
ECC_Update_Database_ Manager_Configuration	Update the DB2 database manager configuration	✔	✔		✔	

*Task Endpoints
DB= database, DB Part. = DB2 partition, DB Part. Grp. = DB2 partition group, Inst. = instance (partitioned or nonpartitioned), Man. Node = managed node

DB2 ECC Task Libraries

Table 11. Data Replication Task Library and Task Endpoints

Task	Description	Task Endpoints*				
		DB	DB Part.	DB Part. Grp.	Inst.	Man. Node
ECC_Apply_Start	Start the data replication Apply program	✔				
ECC_Apply_Stop	Stop the data replication Apply program					✔
ECC_Capture_Get_Log_Seq	Retrieve the current log sequence number and timestamp	✔	✔	✔	✔	
ECC_Capture_Prune	Prune the data replication Capture tables	✔				
ECC_Capture_Reinit	Reinitialize the data replication Capture program	✔				
ECC_Capture_Resume	Resume the data replication Capture program	✔				
ECC_Capture_Start	Start the data replication Capture program	✔				
ECC_Capture_Stop	Stop the data replication Capture program	✔				
ECC_Capture_Suspend	Suspend the data replication Capture program	✔				

*Task Endpoints DB= database, DB Part. = DB2 partition, DB Part. Grp. = DB2 partition group, Inst. = instance (partitioned or nonpartitioned), Man. Node = managed node

Appendix D. DB2 ECC Monitoring Collections and Default Monitors

The DB2 ECC monitoring collections provide over 300 monitoring sources that help you use Tivoli/Sentry to manage distributed DB2 resources (instances and databases) effectively. Each collection is a group of monitoring sources that help you manage a different aspect of your DB2 installations. For example, the DB2 table monitoring collection contains monitoring sources that monitor table row activity. In general, the DB2 ECC monitoring collections are based on the various DB2 monitor groups (for information on the DB2 monitor groups, see the *DB2 System Monitor Guide and Reference* for the version of DB2 that you are using).

The eleven DB2 ECC monitoring collections are:
- DB2 agents and applications
- DB2 basic
- DB2 buffer pool I/O
- DB2 configuration
- DB2 data replication relational
- DB2 locks and deadlocks
- DB2 query
- DB2 SNMP
- DB2 sort work
- DB2 SQL statement activity
- DB2 table

For detailed descriptions of the monitoring sources, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

Some monitoring sources have default thresholds, actions, and monitoring schedules. These monitoring sources are listed in "DB2 ECC Default Monitors" on page 121 .

DB2 Agents and Applications Monitoring Collection

The monitoring sources in this collection provide information about a database manager's agents and their related applications. An agent is a process or thread that carries out the requests made by a client application. Each connected application is served by exactly one coordinator agent and possibly a set of subordinator agents or subagents. Subagents are used for parallel SQL processing in partitioned databases and on SMP machines.

Table 12 lists the monitoring sources in this collection.

Table 12. Monitoring Sources in the DB2 Agents and Applications Collection

GUI Name	CLI Name	Returned Value Format
conn: # of times agents are stolen	connStolenAgents	Numeric

DB2 Agents and Applications Monitoring Collection

Table 12. Monitoring Sources in the DB2 Agents and Applications Collection (continued)

GUI Name	CLI Name	Returned Value Format
conn: Application work load	connApplWorkLoad	Numeric
conn: Maximum associated agents	connMaxAssoAgents	Numeric
conn: Total system CPU time used by agents	connSystemCPUTime	Numeric: seconds
conn: Total user CPU time used by agents	connUserCPUTime	Numeric: seconds
db: Maximum agents associated with applications	dbMaxAgentsAssoWithAppls	Numeric
db: Maximum coordinating agents	dbMaxCoordAgent	Numeric
dbms: # of idle agents	dbmsTotalIdleAgents	Numeric
dbms: # of times agents are stolen	dbmsStolenAgents	Numeric
dbms: Agent creation ratio	dbmsAgentCreationRatio	Numeric
dbms: Agents registered	dbmsAgentsReg	Ratio
dbms: Agents waiting for a token	dbmsAgentsWaiting	Numeric
dbms: Maximum agents registered	dbmsHwmAgentsReg	Numeric
dbms: Maximum agents waiting	dbmsHwmAgentsWaiting	Numeric
dbms: Percent agents waiting	dbmsPctAgentsWait	Percentage
dbms: Percent total connections executing	dbmsPctConnectionsExec	Percentage

DB2 Basic Monitoring Collection

The monitoring sources in this collection provide general information and statistics about:

- Status of the database manager and the DB2 ECC monitoring system
- Catalog caching
- Database activity
- Database connections
- Deadlocks and lock escalations
- Instance piped sorts
- Logging
- Package caching
- Row operations in databases

Some database system monitoring sources provide information about the *Fast Communication Manager* (FCM). In a partitioned database environment, most communication between database partitions is handled by the FCM, which provides internodal communications support. For more information on the FCM, see the *DB2 Administration Guide* for the version of DB2 that you are using.

Table 13. Monitoring Sources in the DB2 Basic Monitoring Collection

GUI Name	CLI Name	Returned Value Format
DB2 ECC monitoring communication agent status	DBTwoMonCommStatus	String: ″up″ or ″down″
DB2 instance status	DBTwoStatus	String: ″up″ or ″down″
DB2 SNMP agent status	DBTwoSnmpStatus	String: ″up″ or ″down″
User-defined OS command (numeric result)	DBTwoSystemCmd	Numeric
User-defined OS command (string result)	DBTwoSystemCmdS	String
conn: Percent locklist space used by application	connPctLocklistUsed	Percentage
db: # of applications connected currently	dbApplsCurCons	Numeric
db: # of applications executing	dbApplsInDbTwo	Numeric
db: # of binds/precomps attempted	dbBindsPrecompiles	Numeric
db: # of catalog cache heap full	dbCatCacheHeapFull	Numeric
db: # of catalog cache inserts	dbCatCacheInserts	Numeric
db: # of catalog cache lookups	dbCatCacheLookups	Numeric
db: # of catalog cache overflows	dbCatCacheOverflows	Numeric
db: # of commit statements attempted	dbCommitSqlStmts	Numeric
db: # of connects since 1st db connect	dbTotalCons	Numeric
db: # of deadlocks detected	dbDeadlocks	Numeric
db: # of internal automatic rebinds	dbIntAutoRebinds	Numeric
db: # of internal commits	dbIntCommits	Numeric
db: # of internal rollbacks	dbIntRollbacks	Numeric
db: # of internal rows deleted	dbIntRowsDeleted	Numeric
db: # of internal rows inserted	dbIntRowsInserted	Numeric
db: # of internal rows updated	dbIntRowsUpdated	Numeric
db: # of lock timeouts	dbLockTimeouts	Numeric
db: # of log pages read	dbLogReads	Numeric: 4 KB pages
db: # of log pages written	dbLogWrites	Numeric: 4 KB pages
db: # of package cache inserts	dbPkgCacheInserts	Numeric
db: # of package cache lookups	dbPkgCacheLookups	Numeric
db: # of rollback statements attempted	dbRollbackSqlStmts	Numeric
db: # of rows deleted	dbRowsDeleted	Numeric
db: # of rows inserted	dbRowsInserted	Numeric
db: # of rows selected	dbRowsSelected	Numeric
db: # of rows updated	dbRowsUpdated	Numeric
db: # of secondary logs allocated currently	dbSecLogsAllocated	Numeric
db: Avg lock escalations per connection	dbAvgLockEscalPerConn	Numeric
db: Internal rollbk due to deadlock	dbIntDeadlockRollbacks	Numeric

DB2 Basic Monitoring Collection

Table 13. Monitoring Sources in the DB2 Basic Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
db: Last backup timestamp	dbLastBackup	String: local timestamp
db: Log I/O	dbLogIo	Numeric
db: Max # of concurrent connections	dbConnectionsTop	Numeric
db: Max database heap allocated	dbDbHeapTop	Numeric: 4 KB pages
db: Max primary log space used	dbMaxPriLogSpaceUsed	Numeric: bytes
db: Max secondary logs space used	dbSecLogUsedTop	Numeric: bytes
db: Max total log space used	dbTotLogUsedTop	Numeric: bytes
db: Percent connections used	dbPctConnUsed	Percentage
db: Percent internal rollbacks due to internal deadlocks	dbPctIntRbPerIntDeadlk	Percentage
db: Percent locklist space used by database	dbPctLocklistUsed	Percentage
db: Percent package cache hit ratio	dbPkgHitRatio	Percentage
db: Percent used in primary log	dbPctUsedInPriLog	Percentage
db: Percent used in secondary log	dbPctUsedInSecLog	Percentage
db: Primary log space allocated	dbPriLogSpaceAlloc	Numeric: bytes
dbms: # of idle agents	dbmsIdleAgents	Numeric
dbms: # of local connections	dbmsLocalCons	Numeric
dbms: # of local connections executing	dbmsLocalConsInExec	Numeric
dbms: # of local databases with current connections	dbmsConLocalDbases	Numeric
dbms: # of piped sorts accepted	dbmsPipedSortsAccepted	Numeric
dbms: # of piped sorts requested	dbmsPipedSortsRequested	Numeric
dbms: # of remote connections	dbmsRemConsIn	Numeric
dbms: # of remote connections executing	dbmsRemConsInExec	Numeric
dbms: Committed private memory	dbmsCommPrivateMem	Numeric
dbms: Last reset timestamp	dbmsLastReset	String: local timestamp
dbms: Percent piped sort hit ratio	dbmsPipedSortHitRatio	Percentage
dbms: Start database manager timestamp	dbmsDbTwostartTime	String: local timestamp
dbpg: Data redistribution status	dbpgNodegroupsRedistStatus	Numeric
dbpg: Data redistribution status for a nodegroup	dbpgNodegroupRedistStatus	Numeric
dbpg: DB2 local nodes status	dbpgNodeStatus	Numeric
dbpg: FCM daemons status	dbpgFcmStatus	Numeric
dbpg: Partitioned database data redistribution status	dbpgRedistStatus	Numeric
dbpg: Percentage differential in row distribution of a table	dbpgPercentDiffRowDistribution	Percentage
tbsp: Percent space used in DMS table space	tbspPctSpaceUsedDMS	Percentage
tbsp: Space used in SMS table space	tbspSpaceUsedSMS	Numeric: bytes

Table 13. Monitoring Sources in the DB2 Basic Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
tbsp: Table space status	tbspStatus	Numeric

DB2 Buffer Pool I/O Monitoring Collection

The monitoring sources in this collection provide statistics about the database buffer pool. The buffer pool is an area of storage into which database pages (that contain table rows or index entries) are read and changed. The purpose of the buffer pool is to improve database system performance. Data can be accessed much faster from memory than from a disk. Therefore, the fewer times the database manager needs to read from or write to a disk, the better the performance.

Table 14 lists the monitoring sources in this collection.

Table 14. Monitoring Sources in the DB2 Buffer Pool I/O Monitoring Collection

GUI Name	CLI Name	Returned Value Format
conn: # of direct reads (4 KB pages)	connDirectReads	Numeric: 4 KB pages
conn: # of direct writes (4 KB pages)	connDirectWrites	Numeric: 4 KB pages
conn: Average # of sectors read per direct read	connAvgSectorsReadPerDirectRead	Numeric
conn: Average # of sectors written per direct write	connAvgSectorsWritePerDirectWrite	Numeric
conn: Buffer pool time waited for prefetch (ms)	connTimeWaitedforPreFetch	Numeric: milliseconds
conn: Data page writes (4 KB pages)	connPoolDataWrites	Numeric: 4 KB pages
conn: Index page reads (4 KB pages)	connPoolIndexPReads	Numeric: 4 KB pages
conn: Index page writes (4 KB pages)	connPoolIndexWrites	Numeric: 4 KB pages
conn: Logical data reads (4 KB pages)	connPoolDataLReads	Numeric: 4 KB pages
conn: Percent buffer pool hit ratio, data+index	connPoolHitRatio	Percentage
conn: Percent buffer pool hit ratio, index	connIndxPlHitRatio	Percentage
conn: Percent package cache hit ratio	connPkgCacheHitRatio	Percentage
conn: Pool read time (s)	connPoolReadTime	Numeric: seconds
conn: Pool write time (s)	connPoolWriteTime	Numeric: seconds
conn: Total pool I/O time (s)	connTotalPoolIoTime	Numeric: seconds
db: # of cleaners due to dirty threshold	dbPlDrtyPgThrshCln	Numeric
db: # of database files closed	dbFilesClosed	Numeric
db: # of direct read requests	dbDirectReadReqs	Numeric
db: # of direct reads from database	dbDirectReads	Numeric
db: # of direct write requests	dbDirectWriteReqs	Numeric
db: # of direct writes to database	dbDirectWrites	Numeric

DB2 Buffer Pool I/O Monitoring Collection

Table 14. Monitoring Sources in the DB2 Buffer Pool I/O Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
db: # of log space cleaners invoked	dbPoolLsnGapClns	Numeric
db: # of page cleans in an interval	dbPageCleans	Numeric
db: # of pages for each prefetch request	dbPagesPerPrefetch	Numeric: 4 KB pages
db: # of pool reads	dbTotalPoolReads	Numeric
db: # of pool writes	dbTotalPoolWrites	Numeric
db: # of prefetch requests in an interval	dbPoolAsyncDatRdRq	Numeric
db: # of synchronous I/O	dbTotalSyncIos	Numeric
db: # of synchronous index reads (4 KB pages)	dbSyncIndexReads	Numeric: 4 KB pages
db: # of synchronous index writes (4 KB pages)	dbSyncIndexWrites	Numeric: 4 KB pages
db: # of synchronous reads	dbSyncDataReads	Numeric
db: # of synchronous writes	dbSyncDataWrites	Numeric
db: # of victim page cleaners invoked	dbPlDrtyPgStealCln	Numeric
db: Average # of async reads per pool read	dbAvgAsyncReadsPerTotReads	Numeric
db: Average # of async writes per pool write	dbAvgAsyncWritesPerTotWrites	Numeric
db: Average # of pool writes per pool read	dbAvgPlWritesPerPlReads	Numeric
db: Average # of sectors read per direct read	dbAvgSectorsReadPerDirectRead	Numeric
db: Average # of sectors written per direct write	dbAvgSectorsWritePerDirectWrite	Numeric
db: Average direct read time (ms)	dbAvgDirectReadTime	Numeric: milliseconds
db: Average direct write time (ms)	dbAvgDirectWriteTime	Numeric: milliseconds
db: Average pool I/O time (ms)	dbAvgPoolIoTime	Numeric: milliseconds
db: Average pool read time (ms)	dbAvgPoolReadTime	Numeric: milliseconds
db: Average pool write time (ms)	dbAvgPoolWriteTime	Numeric: milliseconds
db: Average synchronous data read time (ms)	dbAvgSyncReadTime	Numeric: milliseconds
db: Average synchronous data write time (ms)	dbAvgSyncWriteTime	Numeric: milliseconds
db: Average synchronous I/O (ms)	dbAvgSyncIoTime	Numeric: milliseconds
db: Avg # of pages for each cleaner	dbPagesPerClean	Numeric: 4 KB pages
db: Buffer pool async data reads (4 KB pages)	dbPoolAsyncDataReads	Numeric: 4 KB pages
db: Buffer pool async data writes	dbPoolAsyncDataWrites	Numeric
db: Buffer pool async index reads (4 KB pages)	dbPoolAsyncIndexReads	Numeric: 4 KB pages
db: Buffer pool async index writes	dbPoolAsyncIndexWrites	Numeric
db: Buffer pool async read time (ms)	dbPoolAsyncReadTime	Numeric: milliseconds
db: Buffer pool async write time (ms)	dbPoolAsyncWriteTime	Numeric: milliseconds
db: Buffer pool data logical reads	dbPoolDataLReads	Numeric

Table 14. Monitoring Sources in the DB2 Buffer Pool I/O Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
db: Buffer pool data pages copied from extended storage (4 KB pages)	dbDataPagesCopiedfromExtendedStorage	Numeric: 4 KB pages
db: Buffer pool data pages copied to extended storage (4 KB pages)	dbDataPagesCopiedtoExtendedStorage	Numeric: 4 KB pages
db: Buffer pool data physical reads	dbPoolDataPReads	Numeric
db: Buffer pool data writes	dbPoolDataWrites	Numeric
db: Buffer pool index logical reads	dbPoolIndexLReads	Numeric
db: Buffer pool index pages copied from extended storage (4 KB pages)	dbIndexPagesCopiedfromExtendedStorage	Numeric: 4 KB pages
db: Buffer pool index pages copied to extended storage (4 KB pages)	dbIndexPagesCopiedtoExtendedStorage	Numeric: 4 KB pages
db: Buffer pool index physical reads	dbPoolIndexPReads	Numeric
db: Buffer pool index writes	dbPoolIndexWrites	Numeric
db: Buffer pool I/Os per second	dbPoolIoRate	Numeric
db: Buffer pool time waited for prefetch (ms)	dbTimeWaitedforPreFetch	Numeric: milliseconds
db: Direct read time (ms)	dbDirectReadTime	Numeric: milliseconds
db: Direct write time (ms)	dbDirectWriteTime	Numeric: milliseconds
db: Extended storage read / write ratio	dbExtendedStorageReadWriteRatio	Percentage
db: Percent buffer pool hit ratio, data+index	dbPoolHitRatio	Percentage
db: Percent buffer pool hit ratio, index	dbIndxPlHitRatio	Percentage
db: Percent catalog cache hit ratio	dbCatCacheHitRatio	Percentage
db: Synchronous read time (ms)	dbSyncReadTime	Numeric: milliseconds
db: Synchronous write time (ms)	dbSyncWriteTime	Numeric: milliseconds
db: Total direct I/O time (ms)	dbTotalDirectIoTime	Numeric: milliseconds
db: Total pool physical I/O (ms)	dbTotalPoolIoTime	Numeric: milliseconds
db: Total pool physical read (ms)	dbTotalReadTime	Numeric: milliseconds
db: Total pool physical write (ms)	dbTotalWriteTime	Numeric: milliseconds
db: Total synchronous I/O time (ms)	dbTotalSyncIoTime	Numeric: milliseconds
dbp: # of FCM nodes	dbpTotFcmNodes	Numeric
dbp: Connection status	dbpConnStatus	Numeric
dbp: FCM buffers currently free	dbpFreeFcmBuf	Numeric
dbp: FCM connection entries currently free	dbpFcmConnFree	Numeric
dbp: FCM request blocks currently free	dbpFcmReqBlkFree	Numeric
dbp: Minimum FCM buffers free	dbpMinFcmBufFree	Numeric
dbp: Minimum FCM connection entries free	dbpMinFcmConnEntries	Numeric
dbp: Minimum FCM message anchors free	dbpMinFcmMsgAnchors	Numeric
dbp: Minimum FCM request blocks free	dbpMinFcmReqBlks	Numeric
dbp: Percent FCM buffers currently used	dbpPctFcmBufUsed	Percentage

DB2 Buffer Pool I/O Monitoring Collection

Table 14. Monitoring Sources in the DB2 Buffer Pool I/O Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
dbp: Percent FCM connection entries currently used	dbpPctFcmConnEntriesUsed	Percentage
dbp: Percent FCM request blocks currently used	dbpPctFcmReqBlkUsedPercentage	
dbp: Percent maximum FCM buffers used	dbpPctMaxFcmBufUsed	Percentage
dbp: Percent maximum FCM connection entries used	dbpPctMaxFcmConnEntriesUsed	Percentage
dbp: Percent maximum FCM message anchors used	dbpPctMaxFcmMsgAnchorsUsed	Percentage
dbp: Percent maximum FCM request blocks used	dbpPctMaxFcmReqBlksUsed	Percentage
dbp: Total buffers received	dbpTotBufReceived	Numeric
dbp: Total buffers sent	dbpTotBufSent	Numeric
tbsp: # of direct read requests	tbspDirectReadReqs	Numeric
tbsp: # of direct reads from database	tbspDirectReads	Numeric
tbsp: # of direct write requests	tbspDirectWriteReqs	Numeric
tbsp: # of direct writes to database	tbspDirectWrites	Numeric
tbsp: # of pool reads	tbspTotalPoolReads	Numeric
tbsp: # of pool writes	tbspTotalPoolWrites	Numeric
tbsp: # of synchronous index reads (4 KB pages)	tbspSyncIndexReads	Numeric: 4 KB pages
tbsp: # of synchronous index writes (4 KB pages)	tbspSyncIndexWrites	Numeric: 4 KB pages
tbsp: # of synchronous reads	tbspSyncDataReads	Numeric
tbsp: # of synchronous writes	tbspSyncDataWrites	Numeric
tbsp: Average # of sectors read per direct read	tbspAvgSectorsReadPerDirectRead	Numeric
tbsp: Average # of sectors written per direct write	tbspAvgSectorsWritePerDirectWrite	Numeric
tbsp: Average direct read time (ms)	tbspAvgDirectReadTime	Numeric: milliseconds
tbsp: Average direct write time (ms)	tbspAvgDirectWriteTime	Numeric: milliseconds
tbsp: Average pool I/O time (ms)	tbspAvgPoolIoTime	Numeric: milliseconds
tbsp: Average pool read time (ms)	tbspAvgPoolReadTime	Numeric: milliseconds
tbsp: Average pool write time (ms)	tbspAvgPoolWriteTime	Numeric: milliseconds
tbsp: Average synchronous data read time (ms)	tbspAvgSyncReadTime	Numeric: milliseconds
tbsp: Average synchronous data write time (ms)	tbspAvgSyncWriteTime	Numeric: milliseconds
tbsp: Average synchronous I/O time (ms)	tbspAvgSyncIoTime	Numeric: milliseconds
tbsp: Buffer pool async data reads	tbspPoolAsyncDataReads	Numeric

Table 14. Monitoring Sources in the DB2 Buffer Pool I/O Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
tbsp: Buffer pool async data writes	tbspPoolAsyncDataWrites	Numeric
tbsp: Buffer pool async index reads (4 KB pages)	tbspPoolAsyncIndexReads	Numeric: 4 KB pages
tbsp: Buffer pool async index writes	tbspPoolAsyncIndexWrites	Numeric
tbsp: Buffer pool async read time (ms)	tbspPoolAsyncReadTime	Numeric: milliseconds
tbsp: Buffer pool async write time (ms)	tbspPoolAsyncWriteTime	Numeric: milliseconds
tbsp: Buffer pool data logical reads	tbspPoolDataLReads	Numeric
tbsp: Buffer pool data pages copied from extended storage (4 KB pages)	tbspDataPagesCopiedfromExtendedStorage	Numeric: 4 KB pages
tbsp: Buffer pool data pages copied to extended storage (4 KB pages)	tbspDataPagesCopiedtoExtendedStorage	Numeric: 4 KB pages
tbsp: Buffer pool data physical reads	tbspPoolDataPReads	Numeric
tbsp: Buffer pool data writes	tbspPoolDataWrites	Numeric
tbsp: Buffer pool I/Os per second	tbspPoolIoRate	Numeric
tbsp: Buffer pool index logical reads	tbspPoolIndexLReads	Numeric
tbsp: Buffer pool index pages copied from extended storage (4 KB pages)	tbspIndexPagesCopiedfromExtendedSorage	Numeric: 4 KB pages
tbsp: Buffer pool index pages copied to extended storage (4 KB pages)	tbspIndexPagesCopiedtoExtendedStorage	Numeric: 4 KB pages
tbsp: Buffer pool index physical reads	tbspPoolIndexPReads	Numeric
tbsp: Buffer pool index writes	tbspPoolIndexWrites	Numeric
tbsp: Direct read time	tbspDirectReadTime	Numeric: milliseconds
tbsp: Direct write time (ms)	tbspDirectWriteTime	Numeric: milliseconds
tbsp: Extended storage read/write ratio	tbspExtendedStorageReadWriteRatio	Percentage
tbsp: File closed	tbspFilesClosed	Numeric
tbsp: Percent buffer pool hit ratio, data+index	tbspPoolHitRatio	Percentage
tbsp: Percent buffer pool hit ratio, index	tbspIndxPlHitRatio	Percentage
tbsp: Percent prefetch satisfied	tbspPctAsyDatRdRqS	Percentage
tbsp: Percent total I/O	tbspPctTotalIo	Percentage
tbsp: Prefetch requests in an interval	tbspPlAsyncDatRdRq	Numeric
tbsp: Synchronous read time (ms)	tbspSyncReadTime	Numeric: milliseconds
tbsp: Synchronous write time (ms)	tbspSyncWriteTime	Numeric: milliseconds
tbsp: Total direct I/O time (ms)	tbspTotalDirectIoTime	Numeric: milliseconds
tbsp: Total pool physical I/O time (ms)	tbspTotalPoolIoTime	Numeric: milliseconds
tbsp: Total pool physical read time (ms)	tbspTotalReadTime	Numeric: milliseconds
tbsp: Total pool physical write time (ms)	tbspTotalWriteTime	Numeric: milliseconds
tbsp: Total synchronous I/O	tbspTotalSyncIos	Numeric
tbsp: Total synchronous I/O time (ms)	tbspTotalSyncIoTime	Numeric: milliseconds

DB2 Configuration Monitoring Collection

The monitoring sources in this collection provide information about the configuration parameters for database managers and their databases. Typically, you should tune the values of these parameters to achieve maximum performance for your database environment. For descriptions of these configuration parameters, their default values, and how you can tune them to improve performance of your instances and databases, see the *DB2 Administration Guide* for the version of DB2 that you are using.

Table 15 lists the monitoring sources in this collection.

Table 15. Monitoring Sources in the DB2 Configuration Monitoring Collection

GUI Name	CLI Name	Returned Value Format
db: app_ctl_heap_sz	dbAppCtlHeapSz	Numeric: 4 KB pages
db: applheapsz	dbApplheapsz	Numeric: 4 KB pages
db: avg_appls	dbAvgAppls	Numeric
db: buffpage	dbBuffpage	Numeric
db: catalogcache_sz	dbCatalogcacheSz	Numeric: 4 KB pages
db: chngpgs_thresh	dbChngpgsThresh	Numeric
db: dbheap	dbDbheap	Numeric: 4 KB pages
db: locklist	dbLocklist	Numeric: 4 KB pages
db: logbufsz	dbLogbufsz	Numeric: 4 KB pages
db: logprimary	dbLogprimary	Numeric
db: maxappls	dbMaxappls	Numeric
db: maxlocks	dbMaxlocks	Percentage
db: mincommit	dbMincommit	Numeric
db: newlogpath	dbNewlogpath	Numeric
db: num_iocleaners	dbNumIocleaners	Numeric
db: num_ioservers	dbNumIoservers	Numeric
db: pckcachesz	dbPckcachesz	Numeric: 4 KB pages
db: Percent space used in database heap	dbPctHeapSpaceUsed	Percentage
db: restore_pending	dbRestorePending	String: ″NO″ or ″YES″
db: seqdetect	dbSeqdetect	Numeric
db: sortheap	dbSortheap	Numeric: 4 KB pages
dbms: agentpri	dbmsAgentpri	Numeric
dbms: aslheapsz	dbmsAslheapsz	Numeric: 4 KB pages
dbms: fcm_num_anchors	dbmsFcmNumAnchors	Numeric
dbms: fcm_num_buffers	dbmsFcmNumBuffers	Numeric: 4 KB pages
dbms: fcm_num_connect	dbmsFcmNumConnect	Numeric
dbms: fcm_num_rqb	dbmsFcmNumRqb	Numeric
dbms: max_agents	dbmsMaxAgents	Numeric

Table 15. Monitoring Sources in the DB2 Configuration Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
dbms: max_coordagents	dbmsMaxCoordagents	Numeric
dbms: maxcagents	dbmsMaxcagents	Numeric
dbms: Maximum idle agents in the agent pool	dbmsMaxIdlePoolAgents	Numeric
dbms: Minimum committed memory	dbmsMinCommPrivMem	Numeric: 4 KB pages
dbms: mon_heap_sz	dbmsMonHeapSz	Numeric: 4 KB pages
dbms: num_poolagents	dbmsNumPoolagents	Numeric
dbms: Percent private memory used	dbmsPctPrivMemUsed	Percentage
dbms: query_heap_sz	dbmsQueryHeapSz	Numeric: 4 KB pages
dbms: rqrioblk	dbmsRqrioblk	Numeric
dbms: sheapthres	dbmsSheapthres	Numeric: 4 KB pages
dbms: sqlstmtsz	dbmsSqlstmtsz	Numeric

DB2 Data Replication Relational Monitoring Collection

The monitoring sources in this collection provide information about data replication activity from DataPropagator Relational (DPROPR), which works with DB2 for common servers, and IBM Replication, which is an integrated component of DB2 UDB and DB2 Enterprise-Extended Edition. DPROPR and IBM Replication copy relational data from source tables to target tables.

DPROPR and IBM Replication use two components: the Capture program and the Apply program. The Capture program reads the DB2 database log or journal records to capture data about changes made to source tables. The Apply program refreshes or updates a target table, depending on the applicable source-to-target rules.

For more information about data replication and DB2, see either the *DataPropagator Relational Guide* (for replication using DPROPR) or the *Replication Guide and Reference* (for replication using IBM Replication).

Table 16 lists the monitoring sources in this collection.

Table 16. Monitoring Sources in the DB2 Data Replication Relational Monitoring Collection

GUI Name	CLI Name	Returned Value Format
DataJoiner status	DataJoinerStatus	String: ″up″ or ″down″
Data replication Apply status	DproprApplyStatus	String: ″up″ or ″down″
SNA link status	SnaLinkStatus	String: ″up″ or ″down″
TCP/IP communication link status	TcpLinkStatus	String: ″up″ or ″down″
db: # of Apply lagging subscriptions	dbDproprAppTotSubLag	Numeric
db: Apply failed subscriptions	dbDproprAppFailedSub	Numeric

DB2 Data Replication Relational Monitoring Collection

Table 16. Monitoring Sources in the DB2 Data Replication Relational Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
db: Apply requires full refresh	dbDproprAppNeedFullRefresh	Numeric: **0** No **> 0** Yes
db: Apply subscriptions lag time	dbDproprAppSubLagTime	Numeric: minutes
db: Apply subscription status	dbDproprAppSubStatus	Numeric: **0** Completed **1** In progress **3** Failed
db: Capture errors	dbDproprCapErr	Numeric
db: Capture lag	dbDproprCapLag	Numeric: minutes
db: Capture pruning	dbDpropCapPruning	Numeric

DB2 Locks and Deadlocks Monitoring Collection

The monitoring sources in this collection provide statistics about locks and deadlocks for applications and databases.

Table 17 lists the monitoring sources in this collection.

Table 17. Monitoring Sources in the DB2 Locks and Deadlocks Monitoring Collection

GUI Name	CLI Name	Returned Value Format
conn: Deadlocks found during last interval	connDeadlocksDelta	Numeric
conn: Lock escalations during last interval	connLockEscalsDelta	Numeric
conn: Lock wait time (s)	connLockWaitTimeDelta	Numeric: seconds
conn: Lock waits	connLockWaits	Numeric
conn: Locks held	connLocksHeld	Numeric
conn: UOW lock wait time (s)	connUowLockWaitTime	Numeric: seconds
db: Applications in lock wait	dbApplsInLkwt	Numeric
db: Average lock wait time (s)	dbAvgLockWaitTime	Numeric: seconds
db: Average locks held per application	dbLocksHeldPerAppl	Numeric
db: Deadlocks found during last interval	dbDeadlocksDelta	Numeric
db: Lock escalations during last interval	dbLockEscalationsDelta	Numeric
db: Lock list in use (bytes)	dbLockListInUse	Numeric
db: Lock timeouts during last interval	dbLockTimeoutsDelta	Numeric
db: Lock waits	dbLockWaits	Numeric
db: Lock waits in an interval	dbLockWaitsDelta	Numeric
db: Locks held	dbLocksHeld	Numeric

Table 17. Monitoring Sources in the DB2 Locks and Deadlocks Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
db: Percent application in lock wait	dbPctApplsInLckWt	Percentage
db: Percent deadlock rollbacks	dbPctDlckRollbacks	Percentage
db: Time waited on locks	dbLockWaitTime	Numeric
db: Total exclusive lock escalations	dbXLockEscals	Numeric

DB2 Query Monitoring Collection

The monitoring sources in this collection provide general statistics about database objects.

Table 18 lists the monitoring sources in this collection.

Table 18. Monitoring Sources in the DB2 Query Monitoring Collection

GUI Name	CLI Name	Returned Value Format
db: # of database table spaces	dbTotalDbTbsp	Numeric
db: # of event monitors	dbTotalEventMonitor	Numeric
db: # of invalid triggers	dbTotalInvTrigger	Numeric
db: # of system table space	dbTotalSystemTbsp	Numeric
db: # of tables	dbTotalTable	Numeric
db: # of table spaces	dbTotalTablespace	Numeric
db: # of table spaces with LONG data	dbTotalTbspLONG	Numeric
db: # of triggers	dbTotalTrigger	Numeric
db: # of user indexes	dbTotalUser	Numeric
db: # of views	dbTotalView	Numeric
db: User-defined SQL (numeric result)	dbUserDefinedSqlN	Numeric
db: User-defined SQL (string result)	dbUserDefinedSqlS	String
table: # of rows	tableTotalRow	Numeric
tbsp: # of tables	tbspTotalTable	Numeric
user: # of tables	userTotalTable	Numeric

DB2 SNMP Monitoring Collection

The monitoring sources in this collection provide information from the DB2 Simple Network Management Protocol (SNMP) subagent. This subagent is a program that resides on a managed node and facilitates DB2 systems management through products that conform to the SNMP protocol.

The SNMP subagent:
• Generates alerts to an SNMP manager in the case of a severe DB2 error, such as a condition that requires operator intervention.

DB2 SNMP Monitoring Collection

- Supplies information about the status of the DB2 server and databases. The information provided is documented in the Internet Engineering Task Force standard (RFC 1697). This monitoring collection implements a subset of this available information.

Table 19. Monitoring Sources in the DB2 SNMP Monitoring Collection

GUI Name	CLI Name	Returned Value Format	
DB2 database state	dbRelStat	Numeric:	
		1	Undetermined
		2	Active
		3	Available
		4	Quiesced
		5	Unavailable
DB2 server accumulated connections	srvAccumInAssoc	Numeric	
DB2 server current connections	srvInAssoc	Numeric	
DB2 server finished transactions	srvInfoFinishedTransactions	Numeric	
DB2 server status	srvStatus	Numeric:	
		1	Up
		2	Down
		3	Halted
		4	Congested
		5	Restarting
DB2 server version	srvVersion	String	
Maximum number of agents allowed to register at the same time	srvMaxInboundAssociations	Numeric	
Maximum number of agents registered at the same time	srvHighwaterInboundAssociations	Numeric	
Total Logical bytes read by DB2 server for all active databases	srvLogicalReads	Numeric	
Total logical pages read by DB2 server for all active databases	srvPageReads	Numeric	
Total physical bytes read by DB2 server for all active databases	srvDiskReads	Numeric	
Total physical bytes written by DB2 server for all active databases	srvDiskWrites	Numeric	
Total physical pages written by DB2 server for all active databases	srvPageWrites	Numeric	
Total rows selected returned by DB2 server for all active databases	srvRequestSends	Numeric	
Total SQL statements handled by DB2 server for all active databases	srvHandledRequests	Numeric	

Table 19. Monitoring Sources in the DB2 SNMP Monitoring Collection (continued)

GUI Name	CLI Name	Returned Value Format
Total SQL statements received by DB2 server for all active databases	srvRequestRecvs	Numeric

DB2 Sort Work Monitoring Collection

The monitoring sources in this collection provide information about the database manager sort work. Sorting is often required during a query. Therefore, proper configuration of the sort heap areas can be crucial to improving the query's performance. By monitoring sort work, you can determine the current sort work performance and use the information to tune the sort heap configuration parameters (sortheap and sheapthres) and achieve better performance.

Table 20 lists the monitoring sources in this collection.

Table 20. Monitoring Sources in the DB2 Sort Work Monitoring Collection

GUI Name	CLI Name	Returned Value Format
conn: Application sorts	connSorts	Numeric
conn: Sort time (s)	connSortTime	Numeric
conn: Statement sorts	connStmtSorts	Numeric
db: Active sorts	dbActiveSorts	Numeric
db: Average sort time (ms)	dbAvgSortTime	Numeric: milliseconds
db: Percent sort overflowed	dbPctSortOvflowed	Percentage
db: Sort heap allocated (4 KB pages)	dbSortHeapAllocated	Numeric
db: Sort overflows	dbSortOverflows	Numeric
db: Total sort time (ms)	dbTotalSortTime	Numeric: milliseconds
db: Total sorts	dbTotalSorts	Numeric
dbms: Percent piped sorts rejected	dbmsPctPipSrtsRej	Percentage
dbms: Percent sort heap allocated	dbmsPctSortHeapAlloc	Percentage
dbms: Post threshold sorts	dbmsPostThreshSorts	Numeric
dbms: Sort heap allocated (4 KB pages)	dbmsSortHeapAllocated	Numeric
dbms: Total piped sorts rejected	dbmsPipSrtsRej	Numeric

DB2 SQL Statement Activity Monitoring Collection

The monitoring sources in this collection provide statistics on SQL statement activity. SQL activity is a measure of database throughput and performance.

Table 21 on page 120 lists the monitoring sources in this collection.

DB2 SQL Statement Activity Monitoring Collection

Table 21. Monitoring Sources in the DB2 SQL Statement Activity Monitoring Collection

GUI Name	CLI Name	Returned Value Format
conn: Application section inserts	connApplSectionInserts	Numeric
conn: Application section lookups	connApplSectionLookups	Numeric
conn: Binds/precompiles attempted	connBindsPrecompAttempted	Numeric
conn: Commits	connCommits	Numeric
conn: DDL SQL statements	connDdlSqlStmts	Numeric
conn: Dynamic SQL statements	connDynamicSqlStmts	Numeric
conn: Explicit commits	connCommitSqlStmts	Numeric
conn: Explicit rollbacks	connRollbackSqlStmts	Numeric
conn: Failed SQL statements	connFailedSqlStmts	Numeric
conn: Open block cursors	connOpenBlkCursors	Numeric
conn: Open cursors	connOpenCursors	Numeric
conn: Percent DDL SQL	connPctDdlSql	Percentage
conn: Percent UID SQL	connPctUidSQL	Percentage
conn: Rollbacks	connRollbacks	Numeric
conn: Rows deleted	connRowsDeleted	Numeric
conn: Rows inserted	connRowsInserted	Numeric
conn: Rows selected	connRowsSelected	Numeric
conn: Rows updated	connRowsUpdated	Numeric
conn: Select SQL statements	connSelectSqlStmts	Numeric
conn: SQL requests since last commit	connSQLReqSinceLastCommit	Numeric
conn: Static SQL statements	connStaticSqlStmts	Numeric
conn: Update/Insert/Delete SQL	connUidSqlStmts	Numeric
db: Committed statements per second	dbCommitRate	Numeric
db: DDL SQL statements	dbDdlSqlStmts	Numeric
db: Dynamic SQL statements	dbDySqlStmts	Numeric
db: Failed SQL statements	dbFailedSqlStmts	Numeric
db: Percent DDL SQL	dbPctDdlSqlStmts	Percentage
db: Percent failed SQL	dbPctFailedSqlStmts	Percentage
db: Percent select SQL	dbPctSelectSqlStmts	Percentage
db: Percent UID SQL	dbPctUidSqlStmts	Percentage
db: Rollbacks per second	dbRollbackRate	Numeric
db: Select SQL statements	dbSelectSqlStmts	Numeric
db: SQL statements per second	dbSqlStmtRate	Numeric
db: Static SQL statements	dbStSqlStmts	Numeric
db: Update/Insert/Delete SQL	dbUidSqlStmts	Numeric

DB2 Table Monitoring Collection

The monitoring sources in this collection provide information on:

- Invalid packages
- Table rows (read, written, and overflowed)
- The last time the monitor counters were reset at a table header level
- Whether tables need reorganization

Table 22 lists the monitoring sources in this collection.

Table 22. Monitoring Sources in the DB2 Table Monitoring Collection

GUI Name	CLI Name	Returned Value Format
db: Number of all invalid packages	dbInvalidPkgAll	Numeric
db: Number of invalid SYSTEM packages	dbInvalidPkgSystem	Numeric
db: Number of invalid user packages	dbInvalidPkgUser	Numeric
table: Overflow accesses	tableOverflowAcc	Numeric
table: Reorg needed	tableReorgChkOne	String: YY YN NY NN
table: Rows read per second	tableRowsRead	Numeric
table: Rows written per second	tableRowsWritten	Numeric

DB2 ECC Default Monitors

This section lists DB2 ECC default monitors according to the monitoring collections from which they are derived. The monitors' names are the same as the GUI names of their associated monitoring sources. For the CLI names of these sources, and for the format of the values that the monitors return, see the preceding sections in this appendix. For the default configurations of the monitors, see the *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*.

When adding a monitor for one of these monitoring sources, you can click on the **Add With Defaults** push button in the Add Monitor to Tivoli/Sentry Profile window to add the monitor with its default values.

DB2 agents and applications

- dbms: Agent creation ratio
- dbms: Percent agents waiting
- dbms: Percent of total connections executing

DB2 basic

- DB2 ECC monitoring communication agent status
- DB2 instance status
- db: # of catalog cache heap full
- db: # of catalog cache overflows
- db: # of deadlocks detected

DB2 ECC Default Monitors

- db: # of lock timeouts
- db: Internal rollbk due to deadlock
- db: Percent connections used
- db: Percent internal rollbacks due to internal deadlocks
- db: Percent locklist space used by database
- db: Percent package cache hit ratio
- db: Percent used in primary log
- db: Percent used in secondary log
- dbms: Percent of total connections executing
- dbms: Percent piped sort hit ratio
- tbsp: Percent space used in DMS table space
- tbsp: Space used in SMS table space
- tbsp: Table space status

DB2 buffer pool I/O

- db: # of cleaners due to dirty threshold
- db: # of log space cleaners invoked
- db: # of victim page cleaners invoked
- db: Average # of async reads per pool read
- db: Average # of async writes per pool write
- db: Average # of pool writes per pool read
- db: Average # of sectors read per direct read
- db: Average # of sectors written per direct write
- db: Percent buffer pool hit ratio, data+index
- db: Percent buffer pool hit ratio, index
- db: Percent catalog cache hit ratio
- dbp: Percent FCM buffers currently used
- dbp: Percent FCM connection entries currently used
- dbp: Percent FCM request blocks currently used
- dbp: Percent maximum FCM buffers used
- dbp: Percent maximum FCM connection entries used
- dbp: Percent maximum FCM message anchors used
- dbp: Percent maximum FCM request blocks used

DB2 configuration

- db: Percent space used in database heap

DB2 locks and deadlocks

- db: Applications in lock wait
- db: Deadlocks found during last interval
- db: Lock escalations during last interval
- db: Lock timeouts during last interval
- db: Percent application in lock wait
- db: Percent deadlock rollbacks

DB2 query

- db: # of invalid triggers

DB2 sort work

- db: Percent sort overflowed
- dbms: Percent piped sorts rejected
- dbms: Percent sort heap allocated

DB2 SQL statement activity

- db: Percent DDL SQL
- db: Percent failed SQL

DB2 table

- db: Number of all invalid packages
- db: Number of invalid SYSTEM packages
- table: Reorg needed

DB2 ECC Default Monitors

Appendix E. Setting Up DB2 SNMP Subagents

DB2 ECC provides a monitoring collection to access information supplied by the DB2 SNMP subagent in the AIX and OS/2 environments. The DB2_SNMP collection is made up of 16 monitoring sources chosen from the industry standard RDBMS Management Information Base (MIB).

Before you can activate any monitors derived from these monitoring sources, you must:

* Upgrade the SNMP agent (snmpd) in your AIX or OS/2 base OS package to support the DB2 SNMP subagent (db2snmpd)
* Enable the DB2 SNMP subagent

To upgrade the SNMP agent and enable the DB2 SNMP subagent, perform the following steps. (You need root authority for steps 2 and 3.) If the SNMP agent is already upgraded, perform step 6 only.

1. Acquire the appropriate toolkit. For an AIX environment, acquire the IBM SystemView Agent toolkit for AIX. For an OS/2 environment, acquire the IBM SystemView Agent toolkit for OS/2. You can acquire these toolkits at http://www.networking.ibm.com/sha/shasdk.html. If you do not have access to the Internet, contact your IBM representative.

2. Use the **installp** command to install the SNMP portion of the SystemView agent. The SNMP portion is stored in /usr/lpp/sva and an entry is added to /etc/inittab.

3. If the /usr/sbin/snmpd process is running, stop it. If a dpid2 process is running, stop it also.

4. Start the IBM SystemView agent by entering:

   ```
   export SVA_SNMPD=ACTIVE
   /usr/lpp/sva/bin/svastart
   ```

5. Verify that the correct snmpd daemon is running by entering:

   ```
   export COLUMNS=300
   ps -ef | grep snmp
   ```

   The output should list /usr/lpp/sva/bin/snmpd.

6. Switch to the DB2 instance owner and start the DB2 SNMP subagent by entering:

   ```
   su db2_instance_owner_ID
   /usr/lpp/db2_02_01/bin/db2snmpd -h -c public
   ```

   You must be the DB2 instance owner to start this daemon. If db2snmpd fails to start, ensure that the DB2INSTANCE environment variable is set to one of the DB2 instances on this machine. If DB2INSTANCE is not set, execute the db2profile script found in the sqllib directory of the DB2 instance owner and then restart db2snmpd.

125

Setting Up DB2 SNMP Subagents

Appendix F. Creating Sentry Proxy Endpoints

Before you can distribute a Sentry profile whose monitors are set to capture information from the DB2 SNMP subagent in an OS/2 environment or from the Capture and Apply programs in an MVS/ESA environment, you must set up a proxy endpoint to represent the OS/2 or OS/390 system.

To set up the proxy endpoint:

1. Define **ProxyEndpoint** as a type of managed resource in the policy region that you want the monitors to operate in. For instructions, see "Step 1: Set Up a Policy Region to Contain the Resources That You Want" on page 65.

2. Create a proxy endpoint on a managed node in this policy region:

   a. On your desktop, double-click on the icon for the policy region. The Policy Region window opens.

   b. Select **Create→SentryProxy**. The Create Sentry Proxy Endpoint window opens (Figure 58).
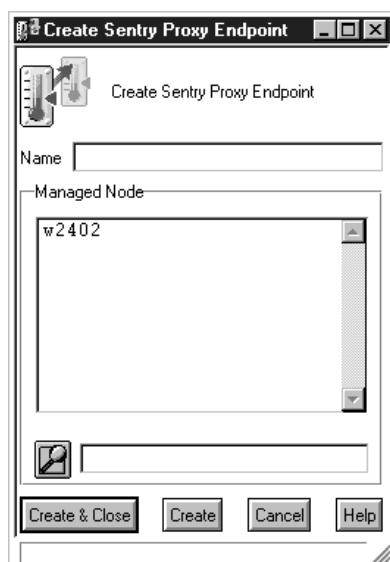
Figure 58. The Create Sentry Proxy Endpoint Window

   c. In the **Name** field, type a name for the proxy endpoint.

   d. In the **Managed Node** box, select the managed node on which you want the proxy endpoint to reside.

   e. Click on the **Create & Close** push button. An icon for the proxy endpoint is displayed on the Policy Region window.
   Tip: If you plan to add many proxy endpoints to the policy region, try to distribute them evenly across the region's managed nodes. This way, you avoid overloading the Sentry engine at any one node with too many monitors.

3. Add the environment variables needed to represent the OS/2 or OS/390 system that you want to monitor:

   a. On the Policy Region window, double-click on the icon for the proxy endpoint. The Sentry Proxy Endpoint window opens.

b. Select **Configure→Set Environment**. The Set Proxy Endpoint Environment window opens (Figure 59).
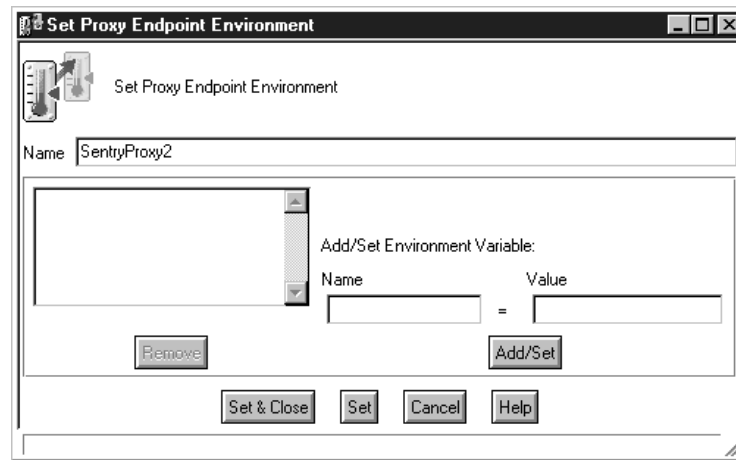
Figure 59. The Set Proxy Endpoint Environment Window

c. For OS/2: Set the environment variable to represent the OS/2 system that you want to monitor.

1) In the **Name** field under **Add/Set Environment Variable**, type:

   ecHostName

2) In the **Value** field, type the machine name of the OS/2 system on which the DB2 SNMP subagent resides. For example, type emerald.

3) Click on the **Add/Set** push button.

4) Click on the **Set & Close** push button.

d. For OS/390: Set each of the following environment variables to represent the OS/390 system that you want to monitor.

1) In the **Name** field under **Add/Set Environment Variable**, type one of the following environment variables:

   ECDB2INSTANCE

   The name of the local DB2 instance registered in the TMR.

   ECDATABASE

   The database alias catalogued at the local DB2 instance.

   ECDB2VERSION

   The version number of the local DB2 instance. For example, type 2 for DB2 for common servers, or type 5 for DB2 UDB.

   ECDB2ISMPP

   Whether or not an instance is MPP, which is specified with one of the following values:

   1   The local DB2 instance is MPP

   0   The local DB2 instance is not MPP

2) In the **Value** field, type the appropriate value for the corresponding environment variable.

3) Click on the **Add/Set** push button.

4) Repeat steps 1 through 3 for each of the environment variables listed in step 1.

      5) Click on the **Set & Close** push button.

4. Set a filter that allows only DB2 SNMP monitors or DB2 data replication relational monitors to capture information:

    a. On the Sentry Proxy Endpoint window, select **Configure**⇥**Set Monitor Filter**. The Set Proxy Monitor Filter window opens (Figure 60).
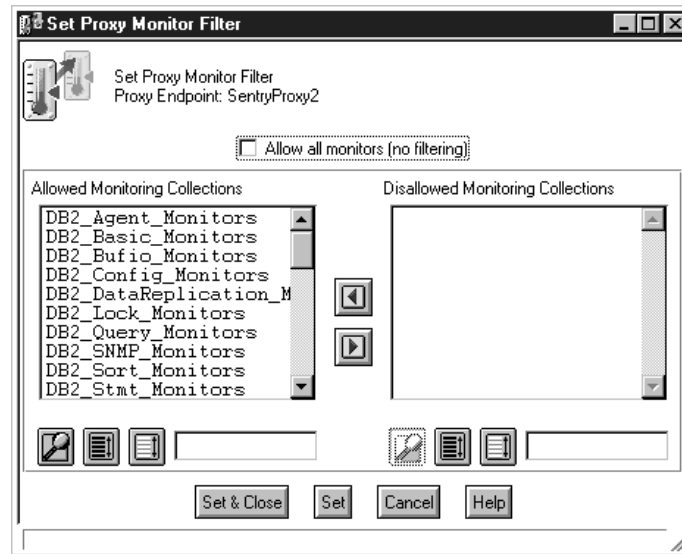
Figure 60. The Set Proxy Monitor Filter Window

    b. In the **Allowed Monitoring Collections** box, select all monitoring collections except DB2_SNMP_Monitors (for OS/2) or DB2_DataReplication_Monitors (for OS/390) and then click on the ▣ push button. The selected collections move to the **Disallowed Monitoring Collections** box.

    c. Click on the **Set & Close** push button.

    d. Close the Sentry Proxy Endpoint window.

5. Subscribe the proxy endpoint to the profile manager that contains the Sentry profile to which you added (or plan to add) monitors of the OS/2 or OS/390 environment. For instructions, see "Step 3: Add Subscribers to the Profile Manager" on page 69.

The proxy endpoint representing the OS/2 or OS/390 environment is set up and you can now distribute a Sentry profile to monitor the OS/2 or OS/390 environment.

Appendix G. Defining DB2 for OS/390 Databases for Monitoring

Some of the monitoring sources in the DB2 data replication relational monitoring collection can be used to monitor the Capture and Apply programs executing at DB2 for OS/390 databases. Before you can use these monitoring sources on DB2 for OS/390 databases, you must:

1. Establish connectivity between the machines with DB2 ECC and the machines with the DB2 for OS/390 databases. Use Distributed Database Connection Services (DDCS) to establish connectivity. For information on connectivity, see the *Distributed Relational Database Architecture Connectivity Guide* and *Wow! DRDA Supports TCP/IP: DB2 Server for OS/390 and DB2 Universal Database*.

2. Define the databases to the DB2 instances that are managed by DB2 ECC (DB2 for AIX, Solaris, or NT instances) by cataloging the DB2 for OS/390 database location in the system database directory on the workstation where the DB2 instance resides. You must specify AUTHENTICATION=CLIENT. End user authentication must occur at the client workstation.

   If you installed Apply for Windows NT and you want Apply for Windows NT to use the AUTHENTICATION=CLIENT scheme to access the source server, you must create a password file. You must have one record in the password file with the following format:

   ```
   SERVER=ABCD PWD=password
   ```

   where ABCD is a nonexistent server name. For more information on creating a password file, see the *Replication Guide and Reference*.

   For more information on how to catalog databases, see the **CATALOG DATABASE** command in the *DB2 Command Reference* for the version of DB2 that you are using.

3. Create Sentry proxy endpoints for the databases (see "Appendix F. Creating Sentry Proxy Endpoints" on page 127 for more information).

The OS/390 user ID must be db2ecc. In addition, if the security product used at the machine where the DB2 for OS/390 database resides is the resource access control facility (RACF), a RACF profile must exist for the OS/390 user ID.

Terms and Abbreviations

A

action.   (1) An operation on a managed object, the semantics of which are defined as part of the managed object class definition. (2) In the UNIX operating system, a defined task that an application performs. An action modifies the properties of an object or manipulates the object in some way.

administrator.   See *TME 10 administrator*.

agent.   (1) In DB2, a separate process or thread that carries out all DB2 requests that are made by a particular client application. Each agent operates with its own private memory and shares the database manager and database global resources such as the buffer pool with other agents. (2) In DB2 ECC, an entity that represents one or more managed objects by emitting notifications regarding the objects and handling requests from managers for management operations to modify or query the objects. See also *daemon*.

alert.   A message sent to a management services focal point in a network to identify a problem or an impending problem.

application ID.   In DB2, a string that uniquely identifies an application across networks. An ID is generated at the time that the application connects to the database. This ID is known on both the client and the server and can be used to correlate the two parts of the application.

Apply program.   A replication program that is used to refresh or update a target table, depending on the applicable source-to-target rules. Contrast with *Capture program*.

authorization role.   In the TME 10 environment, a role assigned to TME 10 administrators to enable them to perform their assigned systems management tasks. A role can be granted over the entire Tivoli Management Region (TMR) or over a specific set of resources, such as those contained in a policy region. Examples of authorization roles include: super, senior, admin, and user. See also *resource role* and *TMR role*.

B

BAROC.   See *Basic Recorder of Objects in C*.

Basic Recorder of Objects in C (BAROC).   In the event server of the TME 10 Enterprise Console, the internal representation of the defined event classes.

C

Capture program.   A replication program that reads database log or journal records to capture data about changes made to source tables. Contrast with *Apply program*.

CLI.   Command line interface.

D

daemon.   A program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their tasks; others operate periodically. See also *agent (2)*.

database partition.   In DB2, a part of a database that consists of its own user data, indexes, configuration files, and transaction logs. Typically, a single database partition exists on each physical node and the processors on each system are used by the database manager at each database partition to manage its part of the database's total data. Because data is divided across database partitions, the power of multiple processors on multiple physical nodes is used to satisfy requests for information.

DB2 database.   In DB2 ECC, a managed resource that represents a particular database in a DB2 instance.

DB2 ECC administrator.   A TME 10 administrator who has been granted the privileges to perform DB2 database management tasks.

DB2 instance.   (1) In DB2, a logical DB2 database manager environment similar to an image of the actual database manager environment. You can have several instances of the DB2 database manager product on the same workstation. (2) In DB2 ECC, a managed object that represents a particular DB2 instance, or a partitioned database server.

DB2 node.   In DB2 ECC, a node that is listed in the db2nodes.cfg node configuration file. This file contains configuration information for all database partitions in an instance, and is shared by all database partitions for that instance. The db2nodes.cfg file is located in the sqllib subdirectory of the home directory for the instance.

DB2 partition.   In DB2 ECC, a managed resource that represents the combination of two items: a DB2 node, and a DB2 database that exists in the partitioned database server that includes that DB2 node. For example, if two logical nodes (0 and 1) exist on one system, and the partitioned database server that includes these two logical nodes has a database named

SAMPLE, you can create two DB2 partitions: one for SAMPLE at node 0, and one for SAMPLE at node 1.

DB2 partition group.   In DB2 ECC, a managed resource that represents a named collection of DB2 partitions. The partitions must belong to the same DB2 instance.

distribute.   In the TME 10 environment, to send a profile to a subscriber or endpoint. The profiles that are sent are copies of the original profile; the original profile remains in the profile manager.

E

endpoint.   (1) In the TME 10 environment, a TME 10 client that is the ultimate recipient for any type of TME 10 operation. Examples are managed nodes and, in DB2 ECC, DB2 instances, DB2 databases, DB2 partitions, and DB2 partition groups. See also *proxy endpoint*. (2) In the TME 10 environment, a TME 10 service that runs on multiple operating systems and performs TME 10 operations on those systems, thereby enabling the TME 10 Framework to manage the systems as TME 10 clients.

event.   (1) An occurrence of significance to a task; for example, an SNMP trap or the completion of an asynchronous operation. (2) In the TME 10 environment, any significant change in the state of a system resource, network resource, or network application. An event can be generated for a problem, for the resolution of a problem, or for the successful completion of a task. Examples of events are: the normal starting and stopping of a process, the abnormal termination of a process, and the malfunctioning of a server. (3) In TME 10 Enterprise Console (TEC), a message sent to the TME 10 Enterprise Console.

F

fast communication manager (FCM).   A group of functions that provide internodal communication support.

FCM.   See *fast communication manager*.

I

indicator.   In Sentry, an icon on the TME 10 desktop that graphically displays the status of a monitor that has been associated with it. The icon resembles a thermometer, which the TME 10 administrator can read to determine the status of the monitor.

indicator collection.   In the TME 10 environment, a single location from which a TME 10 administrator can determine the status of monitors in different profiles, as well as clear and reset alarmed states.

indoubt transaction.   A global transaction whose state is in doubt when the transaction is prepared, but not yet

committed or rolled back; or when one phase of it completes successfully but the system fails before the second phase can complete, leaving the database in an inconsistent state.

J

job.   In the TME 10 environment, a resource consisting of a task and its preconfigured execution parameters. The execution parameters can specify the set of managed resources on which the job is to execute.

M

managed node.   In the TME 10 environment, any managed system (for example, a UNIX host or Windows NT system) on which the TME 10 Framework is installed. Each managed node is provided with the capability of interpreting and processing profiles that are distributed to it.

managed resource.   In the TME 10 environment, any hardware or software entity (machine, service, system, or facility) that is represented by a database object in the TME 10 object database and an icon on the TME 10 desktop. The managed resources in a particular policy region must be of a resource type that is supported in the region, and they are subject to the rules that govern the region. Managed resources include, but are not limited to, profiles, profile managers, managed nodes, and, in DB2 ECC, DB2 instances, DB2 databases, DB2 partitions, and DB2 partition groups.

monitor.   Software that observes, supervises, controls, or verifies operations of a system. Monitors typically monitor information such as available disk space or application errors and compare the information to defined thresholds. When thresholds are exceeded, either system or network administrators can be notified, or an automated response can be performed.

monitoring collection.   In Sentry, a collection of monitoring sources or predefined monitors. Several monitoring collections are packaged with Sentry, but TME 10 administrators can use custom-developed and third-party monitoring collections as well, such as DB2 ECC monitoring collections.

monitoring source.   In DB2 ECC, a monitor that has been predefined to observe and record a particular activity. For example, the "DB2 instance status" monitoring source determines whether the DB2 instance is up or down. DB2 ECC monitoring sources are contained in the various DB2 ECC monitoring collections.

N

notice.   In the TME 10 environment, a message generated by a systems management operation that

contains information about an event or the status of an application. Notices are stored in *notice groups.*

notice group. In the TME 10 environment, an application- or operation-specific container that stores and displays notices pertaining to specific TME 10 functions.

O

oserv. The name of the object request broker used by the TME 10 environment. This object request broker runs on the TMR server and each TMR client. In object-oriented programming, an object request broker is software that serves as an intermediary by transparently enabling objects to exchange requests and responses.

P

partitioned database. A database with two or more database partitions. Data in user tables can be located in one or more database partitions. When a table is on multiple partitions, some of its rows are stored in one partition and others are stored in other partitions. See *database partition*. Contrast with *single-partition database*.

partitioned database server. In DB2 ECC, a DB2 instance that has one or more partitioned databases.

policy. In the TME 10 environment, a set of rules that are applied to managed resources. A specific rule in a policy is referred to as a policy method.

policy region. In the TME 10 environment, a group of managed resources that share one or more common policies. TME 10 administrators use policy regions to model the management and organizational structure of a network computing environment. The administrators can group similar resources, define access to and control the resources, and associate rules for governing the resources. The policy region contains resource types and the list of resources to be managed. A policy region is represented on the TME 10 desktop by an icon that resembles a capitol building (dome icon). When a Tivoli Management Region (TMR) is created, a policy region for the host where Tivoli is installed is also created. In most cases, a TME 10 administrator creates other policy regions and subregions to represent the organization of resources. A TMR addresses the physical connectivity of resources, whereas a policy region addresses the logical organization of resources.

policy subregion. In the TME 10 environment, a policy region created or residing in another policy region.

profile. In the TME 10 environment, a container for application-specific information about a particular type of resource. A TME 10 application specifies the template for its profiles; the template includes information about the resources that can be managed by that TME 10 application. For example, Sentry specifies the templates for its profiles, which contain definitions of monitors. A profile is created in the context of a profile manager; the profile manager links a profile to the TME 10 resource (for example, a managed node) that uses the information contained in the profile. A profile does not have any direct subscribers.

profile manager. In the TME 10 environment, a container for profiles that links the profiles to a set of resources, called *subscribers.* Subscribers can include managed nodes, other profile managers, and, in DB2 ECC, DB2 instances, DB2 databases, DB2 partitions, and DB2 partition groups. A profile manager can contain profiles of multiple types or multiple profiles of the same type. TME 10 administrators use profile managers to organize and distribute profiles. A profile manager is created in the context of a policy region and is a managed resource in a policy region.

proxy endpoint. A representation of an entity (such as a network device or a host) that functions as a subscriber for Sentry profiles. A TME 10 administrator associates each proxy endpoint with a managed node; several proxy endpoints can be associated with a single managed node.

R

registration. In DPROPR, the act of identifying a source table to DPROPR to make the table available for subscription. In DB2 UDB and DB2 Enterprise-Extended Edition, the term source table is used instead.

resource role. In DB2 ECC, an authorization role that allows an administrator to perform an operation, or set of operations, against a specific managed resource.

response. A response action.

response action. In Sentry, a defined operation or task that is performed when the state of a monitor reaches a specified threshold. Sending a Tivoli notice to a TME 10 administrator is a response action.

response level. In Sentry, the state of a monitor when a specified threshold has been reached. An administrator can set thresholds for each response level and have Sentry trigger a different response action for each level. There can also be several response actions for each response level. The five response levels are: critical, severe, warning, normal, and always.

role. See *authorization role*.

root. In the UNIX operating system, the user name for the system user with the most authority.

S

scheduler. A computer program designed to perform functions such as scheduling, initiating, and terminating jobs.

Sentry. A TME 10 product that monitors system resources, initiates any necessary corrective actions, and informs system administrators of potential problems. Sentry consists of a group of monitors that are installed on each managed node that is to be monitored. It resolves some events on its own and can send others to the TME 10 Enterprise Console. See also TME 10 Distributed Monitoring.

Simple Network Management Protocol (SNMP). In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application layer protocol. Information on devices managed is defined and stored in the application's Management Information Base (MIB).

single-partition database. A database that has only one database partition.

SNMP. Simple Network Management Protocol.

source table. In DB2 UDB and DB2 Enterprise-Extended Edition, a table that contains the data that is to be copied to a target table.

subscribe. In the TME 10 environment, to become eligible to receive profiles that reside in a profile manager.

subscriber. In the TME 10 environment, a TME 10 client, a profile manager, and any endpoint type (for example, a PC managed node or a proxy endpoint) that is subscribed to a profile manager. In DB2 ECC, a DB2 instance, DB2 database, DB2 partition, and DB2 partition group can be subscribers. Although profiles are distributed to a subscriber, the subscriber may or may not be the final destination of the profile distribution. See also *endpoint*.

subscription. In the TME 10 environment, the process of identifying the endpoints or profile managers to which profiles will be distributed. Endpoints, profile managers, and profiles are associated with each other in profile managers.

subscription list. In the TME 10 environment, a list that identifies the subscribers to a profile manager. Including a profile manager on a subscription list (in effect, a list within a list) is a way of subscribing several resources simultaneously rather than adding each one individually.

T

task. In the TME 10 environment, the definition of an action that must be routinely performed on various managed nodes throughout the network. A task defines the executables to be run when the task is executed, the authorization role required to execute the task, and the user and group name under which the task will execute. DB2 ECC comes with custom-developed tasks which can be run against DB2 instances, DB2 databases, DB2 partitions, DB2 partition groups, and managed nodes.

task library. In the TME 10 environment, a container in which a TME 10 administrator can create and store tasks and jobs. DB2 ECC comes with custom-developed task libraries.

TEC. See *TME 10 Enterprise Console*.

threshold. A value that defines a limit for a monitored condition.

Tivoli Management Region (TMR). In the TME 10 environment, a network of connected nodes consisting of a TME 10 server and its clients. A TME 10 server is the server that holds or references the complete set of TME 10 software. An organization can have more than one TMR. A TMR addresses the physical layout of resources, whereas a policy region addresses the logical organization of resources.

TME 10. The suite of Tivoli applications that enable system administrators to manage their network computing enterprise.

TME 10 administrator. In the TME 10 environment, a system administrator who has been authorized to perform systems management tasks and manage policy regions in one or more networks. Each TME 10 administrator is represented by an icon on the TME 10 desktop.

TME 10 client. A client of a TME 10 server.

TME 10 desktop. In the TME 10 environment, the graphical user interface that system administrators use to manage their network computing environment.

TME 10 Distributed Monitoring. In this book, the term Sentry is used to refer to TME 10 Distributed Monitoring. See *Sentry*.

TME 10 Enterprise Console (TEC). A TME 10 product that collects, processes, and automatically initiates corrective actions for system, application, network, and database events; it is the central control point for events from all supported sources. The TME 10 Enterprise Console provides a centralized, global view of the network computing environment; it uses distributed event monitors to collect information, distributed event consoles to present the information to

system administrators, and a central event server to correlate and interpret the information.

TME 10 environment. The TME 10 applications, based on the TME 10 Framework, that are installed at a specific customer location and that address network computing management issues across many platforms. In a TME 10 environment, a system administrator can distribute software, manage user configurations, change access privileges, automate operations, monitor resources, and schedule jobs.

TME 10 Framework. The base software that is required to run the applications in the TME 10 product suite. This software infrastructure enables the integration of systems management applications from Tivoli and the Tivoli Partners. The Framework includes:

- Object request broker (oserv)
- Distributed object database
- Platform services and functions, such as notice groups and the graphical user interface

In a TME 10 environment, the TME 10 Framework is installed on every client and every server with these exceptions:

- The TME 10 Framework is never installed on a client PC; rather, the PC agent is installed on the PC.
- In a particular TMR, the TME 10 server is the only server that contains the full object database.

TME 10 server. A server that holds or references the complete set of TME 10 software, including the full object database.

TMR. See *Tivoli Management Region (TMR)*.

TMR client. In a TME 10 environment, any computer—except the TMR server—on which the TME 10 Framework is installed. See also *TMR server*.

TMR role. In DB2 ECC, an authorization role that allows an administrator to perform an operation, or set of operations, that involves an entire TMR, or that can be run against any managed node within a TMR.

TMR server. The server that holds or references the complete set of TME 10 software, including the full object database, for a specific Tivoli Management Region (TMR).

V

victim. A page or buffer that has been selected for an action to be performed on it. For example, a page that will be written to disk.

Bibliography

This bibliography includes all publications cited in this book and publications that provide additional information for DB2 ECC users.

The DB2 Enterprise Control Center for TME 10 V1.2 Library

The following books comprise the DB2 Enterprise Control Center for TME 10 V1.2 library:

- *DB2 Enterprise Control Center for TME 10 User's Guide*, SC26-9084

  This book describes how to install and use the DB2 ECC product.

- *DB2 Enterprise Control Center for TME 10 Monitoring Collection and Task Library Reference*, SC26-9085

  This book contains information on the monitoring collections and task libraries provided by DB2 ECC.

Related IBM Publications

This section lists books associated with IBM products.

DB2 for common servers Publications

- *DB2 Administration Guide*, S20H-4580
- *DB2 API Reference for common servers*, S20H-4984
- *DB2 Command Reference*, S20H-4645
- *DB2 Database System Monitor Guide and Reference*, S20H-4871
- *IBM DATABASE 2 for Windows 95 & Windows NT Installation and Operation Guide*, S33H-0312
- *DB2 SQL Reference*, S20H-4665

DB2 Universal Database Publications

- *DB2 Universal Database Administration Getting Started*, S10J-8154
- *DB2 Universal Database Administration Guide*, S10J-8157
- *DB2 Universal Database API Reference*, S10J-8167
- *DB2 Universal Database Command Reference*, S10J-8166
- *DB2 Universal Database System Monitor Guide and Reference*, S10J-8164
- *DB2 Universal Database for Windows NT Quick Beginnings*, S10J-8149
- *DB2 Universal Database SQL Reference*, S10J-8165

DDCS and DB2 Connect Publications

- *Distributed Database Connection Services User's Guide for common servers*, S20H-4793
- *DB2 Connect User's Guide*, S10J-8163

DRDA Publications

- *Distributed Relational Database Architecture Connectivity Guide*, SC26-4783

IBM Replication Publications

- *DataPropagator Relational Guide*, SC26-3399
- *Replication Guide and Reference*, S95H-0999

ITSO (Redbook) Publications

- *TME 10 Cookbook for AIX Systems Management and Networking*, SG24-4867
- *Understanding Tivoli's TME 3.0 and TME 10*, SG24-4948
- *Wow! DRDA Supports TCP/IP: DB2 Server for OS/390 and DB2 Universal Database*, SG24-2212

Tivoli Publications

This section lists books produced by Tivoli.

Tivoli/Sentry and Distributed Monitoring Publications

- *Tivoli/Sentry User's Guide*, GC31-8382
- *TME 10 Distributed Monitoring User's Guide*, GC31-8382

TME 10 Versions 3.1 and 3.2 Publications

- *TME 10 Framework Planning and Installation Guide*, SC31-8432

- *TME 10 Framework Reference Manual*, SC31-8434

- *TME 10 Framework User's Guide*, GC31-8433

Index

Special Characters
$BINDIR, definition   13

A
Add Monitor to Tivoli/Sentry Profile window   70
Add Scheduled Job window   56
adding a DB2 instance   17
administration troubleshooting   88
administrators
   description   6
   establishing
     existing   27
     first time   23
     overview   23
   roles, tables of   97
Administrators window   6
Adobe Acrobat Reader
   downloading from the Web   19
   installing   19
authorization roles, tables of   97
automatically started processes, stopping   19

B
book
   installing online   19
   terminology   1

C
command line processor (CLP), opening   45
commands
   running from the CLP   45
   setup_env.cmd   95
   waddmon
     scheduling monitors   80
     selecting monitoring sources   80
     specifying responses to monitored information
       80
   wcrdb2db   35
   wcrdb2inst   33
   wcrdb2part   39
   wcrdb2partgp   41
   wcrtjob   15
   wcrtpf   80
   wcrtpr   28
   wcrtprfmgr   80
   wcrtsntcoll   80
   wdelpr   95
   wdelsched   15
   wdistrib   65, 81
   wgetjob   15
   wmv   30
   wruntask   52
   wschedjob   59
   wsetcoll   80
   wsetpr   80

commands *(continued)*
   wsetsntid   81
   wstopeng   90
   wsub   64, 80
contents, viewing for DB2 objects   31
Create Administrator window   23
Create DB2Database window   34
Create DB2Instance window   32
Create DB2Partition window   38
Create DB2PartitionGroup window   40
Create Job window   52
Create Policy Region window   28
Create Profile Manager window   67
Create Profile window   67
Create Sentry Indicator Collection window   68
Create Sentry Proxy Endpoint window   127
creating
   DB2 databases
     from the command line   35
     from the desktop   34
   DB2 instances
     from the command line   33
     from the desktop   31
   DB2 partition groups
     from the command line   41
     from the desktop   39
   DB2 partitions
     from the command line   39
     from the desktop   37
   db2ecc OS user ID   13
   policy regions   28
Customize Monitoring Schedule Restrictions window
   77

D
daemon monitoring process, turning off   19
data replication task endpoints   104
data replication tasks   8
DB2 agents and applications monitoring collection   105
DB2 basic monitoring collection   106
DB2 buffer pool I/O monitoring collection   109
DB2 Command Line Processor window   45
DB2 commands, running from the CLP   45
DB2 configuration monitoring collection   114
DB2 data replication relational monitoring collection
   115
DB2 database task endpoints   102
DB2 database tasks   8
DB2 databases
   creating
     from the command line   35
     from the desktop   34
   description   5
   discovering   36
   running commands   45
   viewing contents   31
   viewing properties   42

141

Readers' Comments — We'd Like to Hear from You

**DB2 Enterprise Control Center for TME 10 V1.2
User's Guide**

SC26-9084-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you?　☐ Yes　☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____　Address _____

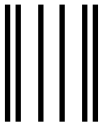Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You

SC26-9084-01

IBM ®

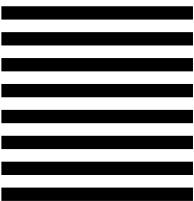Fold and Tape   **Please do not staple**   Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL   PERMIT NO. 40   ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department W92/H3
PO Box 49023
San Jose, CA
 95161-9945

Fold and Tape   **Please do not staple**   Fold and Tape

SC26-9084-01

IBM ®

Part Number: 4076856