

IBM Tivoli Monitoring for Databases: DB2 Warehouse Enablement Pack Implementation Guide

Version 1.1.0.1

Edition notice

Second Edition

Copyright Notice

© Copyright IBM Corporation 2004. All rights reserved. May only be used pursuant to a Tivoli Systems Software License Agreement, an IBM Software License Agreement, or Addendum for Tivoli Products to IBM Customer or License Agreement. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without prior written permission of IBM Corporation. IBM Corporation grants you limited permission to make hardcopy or other reproductions of any machine-readable documentation for your own use, provided that each such reproduction shall carry the IBM Corporation copyright notice. No other rights under copyright are granted without prior written permission of IBM Corporation. The document is not intended for production and is furnished "as is" without warranty of any kind. **All warranties on this document are hereby disclaimed, including the warranties of merchantability and fitness for a particular purpose.**

U.S. Government Users Restricted Rights—Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corporation.

Trademarks

IBM, the IBM logo, Tivoli, the Tivoli logo, AIX, Cross-Site, NetView, OS/2, Planet Tivoli, RS/6000, Tivoli Enterprise, Tivoli Enterprise Console, Tivoli Ready, and TME are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Lotus is a registered trademark of Lotus Development Corporation and/or IBM Corporation in the United States or other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

PC Direct is a trademark of Ziff Communications Company in the United States, other countries, or both and is used by IBM Corporation under license.

ActionMedia, LANdesk, MMX, Pentium, and ProShare are trademarks of Intel Corporation in the United States, other countries, or both. For a complete list of Intel trademarks, see <http://www.intel.com/sites/corporate/tradmarx.htm>.

SET and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC. For further information, see <http://www.setco.org/aboutmark.html>.



Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

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

Notices

References in this publication to Tivoli Systems or IBM products, programs, or services do not imply that they will be available in all countries in which Tivoli Systems or IBM operates. Any reference to these products, programs, or services is not intended to imply that only Tivoli Systems or IBM products, programs, or services can be used. Subject to valid intellectual property or other legally protectable right of Tivoli Systems or IBM, any functionally equivalent product, program, or service can be used instead of the referenced product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by Tivoli Systems or IBM, are the responsibility of the user. Tivoli Systems or 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 the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, New York 10504-1785, U.S.A.

Contents

1	About this document	8
1.1	Related Documentation.....	8
1.1.1	Tivoli Enterprise Data Warehouse	8
1.1.2	IBM DB2 Universal Database, DB2 Universal Database Data Warehouse Center, and DB2 Universal Database Warehouse Manager	8
1.1.3	IBM Tivoli Monitoring for Databases: DB2.....	9
1.1.4	Tivoli Enterprise Data Warehouse	10
1.1.5	DB2	10
2	Overview	12
2.1	Overview of Tivoli Enterprise Data Warehouse	12
2.2	Overview of the DB2 Warehouse Enablement Pack.....	13
2.3	Typical usage scenario	13
3	Installing and configuring	15
3.1	Prerequisites	15
3.2	Supported hardware and software.....	15
3.3	Limitations.....	15
3.4	Database sizing considerations	16
3.5	Data sources and targets	20
3.6	Pre-installation steps	20
3.7	Installation procedure	21
3.8	Post-installation steps	21
4	Maintaining	24
4.1	Backing up and restoring	24
4.2	Pruning	24
4.3	Other Tools.....	24
5	ETL processes	25
5.1	ETL Processes	25
5.2	CTD_m05_ETL2_Process	25
5.3	CTD_m10_Reset_ETL2_Process.....	26
6	Generic schema implementation	27
6.1	Component configuration.....	27
6.1.1	Component type (table CompTyp).....	28
6.1.2	Component (table Comp)	29
6.1.3	Component relationship type (table RelnTyp)	29
6.1.4	Component relationship rule (table RelnRul).....	29
6.1.5	Component relationship (table CompReln).....	30
6.1.6	Attribute type (table AttrTyp).....	31
6.1.7	Attribute rule (table AttrRul)	32
6.1.8	Attribute domain (table AttrDom)	32
6.1.9	Component attribute (table CompAttr).....	33

6.2	Component measurement	33
6.2.1	Measurement group type (table MgrpTyp)	33
6.2.2	Measurement group (table MGrp)	33
6.2.3	Measurement group member (table MGrpMbr)	36
6.2.4	Measurement unit category (table MunitCat)	41
6.2.5	Measurement unit (table Munit)	41
6.2.6	Time summary (table TmSum)	41
6.2.7	Measurement source (table MSrc)	42
6.2.8	Measurement type (table MsmtTyp)	42
6.2.9	Component measurement rule (table MsmtRul)	50
6.2.10	Measurement (table Msmt)	57
6.3	Helper tables	57
6.4	Exception tables	57
6.5	Incremental extraction	58
7	IBM Tivoli Monitoring integration	60
7.1	Metadata tables for applications that use the generic IBM Tivoli Monitoring warehouse enablement pack	60
7.1.1	Resource translation (table Resource_Transl)	60
7.1.2	Category translation (table Category_Transl)	60
7.1.3	Component type translation (table CompTyp_Transl)	61
7.1.4	Attribute translation (table AttrTyp_Transl)	61
7.2	IBM Tivoli Monitoring resource models	61
7.2.1	DB2 Agents Resource Model	62
7.2.2	DB2 Apply Replication Resource Model	63
7.2.3	DB2 BufferPool Resource Model	63
7.2.4	DB2 BufferPool External Storage Resource Model	64
7.2.5	DB2 Catalog Cache Resource Model	65
7.2.6	DB2 CPU Utilization Resource Model	66
7.2.7	DB2 Cursor Resource Model	66
7.2.8	DB2 Instance Status Resource Model	67
7.2.9	DB2 Database Status Resource Model	68
7.2.10	DB2 Direct IO Resource Model	72
7.2.11	DB2 FCM Activity Resource Model	73
7.2.12	Host Throughput Resource Model	74
7.2.13	DB2 Locks- Deadlocks Resource Model	74
7.2.14	DB2 Lock Waits Resource Model	75
7.2.15	DB2 Logging Resource Model	76
7.2.16	DB2 Package Cache Resource Model	76
7.2.17	DB2 Replication Capture Resource Model	77
7.2.18	DB2 SAP Tablespace Usage Status Resource Model	77
7.2.19	DB2 Sorts Resource Model	78
7.2.20	DB2 SQL Statement Activity Resource Model	80
7.2.21	DB2 Table Apply Replication Resource Model	81
7.2.22	DB2 Table Activity Resource Model	82
8	Data mart schema information	83
8.1	Star schemas	83
8.1.1	CTD Instance Hourly Star Schema	83

8.1.1.1	Fact table CTD.F_INST_HOUR	83
8.1.2	CTD Instance Daily Star Schema	83
8.1.2.1	Fact table CTD.F_INST_DAY	84
8.1.3	CTD Instance Weekly Star Schema	84
8.1.3.1	Fact table CTD.F_INST_WEEK	84
8.1.4	CTD Instance Monthly Star Schema	84
8.1.4.1	Fact table CTD.F_INST_MONTH	85
8.1.5	CTD Database Hourly Star Schema	85
8.1.5.1	Fact table CTD.F_DB_HOUR	85
8.1.6	CTD Database Daily Star Schema	85
8.1.6.1	Fact table CTD.F_DB_DAY	86
8.1.7	CTD Database Weekly Star Schema	86
8.1.7.1	Fact table CTD.F_DB_WEEK	86
8.1.8	CTD Database Monthly Star Schema	86
8.1.8.1	Fact table CTD.F_DB_MONTH	87
8.1.9	CTD Database Partition Hourly Star Schema	87
8.1.9.1	Fact table CTD.F_NODE_HOUR	87
8.1.10	CTD Database Partition Daily Star Schema	87
8.1.10.1	Fact table CTD.F_NODE_DAY	88
8.1.11	CTD Database Partition Weekly Star Schema	88
8.1.11.1	Fact table CTD.F_NODE_WEEK	88
8.1.12	CTD Database Partition Monthly Star Schema	88
8.1.12.1	Fact table CTD.F_NODE_MONTH	89
8.1.13	CTD Gateway Hourly Star Schema	89
8.1.13.1	Fact table CTD.F_GTWY_HOUR	89
8.1.14	CTD Gateway Daily Star Schema	89
8.1.14.1	Fact table CTD.F_GTWY_DAY	90
8.1.15	CTD Gateway Weekly Star Schema	90
8.1.15.1	Fact table CTD.F_GTWY_WEEK	90
8.1.16	CTD Gateway Monthly Star Schema	90
8.1.16.1	Fact table CTD.F_GTWY_MONTH	91
8.1.17	CTD Application Hourly Star Schema	91
8.1.17.1	Fact table CTD.F_APPL_HOUR	91
8.1.18	CTD Application Daily Star Schema	91
8.1.18.1	Fact table CTD.F_APPL_DAY	92
8.1.19	CTD Application Weekly Star Schema	92
8.1.19.1	Fact table CTD.F_APPL_WEEK	92
8.1.20	CTD Application Monthly Star Schema	92
8.1.20.1	Fact table CTD.F_APPL_MONTH	93
8.1.21	CTD Table Hourly Star Schema	93
8.1.21.1	Fact table CTD.F_TABLE_HOUR	93
8.1.22	CTD Table Daily Star Schema	93
8.1.22.1	Fact table CTD.F_TABLE_DAY	94
8.1.23	CTD Table Weekly Star Schema	94
8.1.23.1	Fact table CTD.F_TABLE_WEEK	94
8.1.24	CTD Table Monthly Star Schema	94
8.1.24.1	Fact table CTD.F_TABLE_MONTH	95
8.1.25	CTD Tablespace Partition Hourly Star Schema	95
8.1.25.1	Fact table CTD.F_TBSP_ND_HOUR	95

8.1.26	CTD Tablespace Partition Daily Star Schema	95
8.1.26.1	Fact table CTD.F_TBSP_ND_DAY	96
8.1.27	CTD Tablespace Partition Weekly Star Schema	96
8.1.27.1	Fact table CTD.F_TBSP_ND_WEEK	96
8.1.28	CTD Tablespace Partition Monthly Star Schema	96
8.1.28.1	Fact table CTD.F_TBSP_ND_MONTH	97
8.1.29	CTD Apply Program Star Hourly Schema	97
8.1.29.1	Fact table CTD.F_APPLY_HOUR	97
8.1.30	CTD Apply Program Star Daily Schema	97
8.1.30.1	Fact table CTD.F_APPLY_DAY	98
8.1.31	CTD Apply Program Star Weekly Schema	98
8.1.31.1	Fact table CTD.F_APPLY_WEEK	98
8.1.32	CTD Apply Program Star Monthly Schema	98
8.1.32.1	Fact table CTD.F_APPLY_MONTH	99
8.1.33	CTD Table (Apply Level) Hourly Star Schema	99
8.1.33.1	Fact table CTD.F_TAPPLY_HOUR	99
8.1.34	CTD Table (Apply Level) Daily Star Schema	99
8.1.34.1	Fact table CTD.F_TAPPLY_DAY	100
8.1.35	CTD Table (Apply Level) Weekly Star Schema	100
8.1.35.1	Fact table CTD.F_TAPPLY_WEEK	100
8.1.36	CTD Table (Apply Level) Monthly Star Schema	100
8.1.36.1	Fact table CTD.F_TAPPLY_MONTH	101
8.2	Metric dimension tables	101
8.2.1	CTD.D_INST_METRIC	101
8.2.2	CTD.D_DB_METRIC	102
8.2.3	CTD.D_NODE_METRIC	102
8.2.4	CTD.D_GTWY_METRIC	102
8.2.5	CTD.D_APPL_METRIC	102
8.2.6	CTD.D_APPLY_METRIC	102
8.2.7	CTD.D_TBSP_ND_METRIC	102
8.2.8	CTD.D_TABLE_METRIC	102
8.2.9	CTD.D_TAPPLY_METRIC	102
8.3	Dimension tables	102
8.3.1	Dimension table CTD.D_HOST	103
8.3.2	Dimension table CTD.D_HOST_IP_AD	103
8.3.3	Dimension table CTD.D_INST	103
8.3.4	Dimension table CTD.D_INST_VERS	103
8.3.5	Dimension table CTD.D_DB	103
8.3.6	Dimension table CTD.D_NODE	103
8.3.7	Dimension table CTD.D_ND_LGPTH	103
8.3.8	Dimension table CTD.D_TABLE	103
8.3.9	Dimension table CTD.D_TBSP_ND	103
8.3.10	Dimension table CTD.D_GTWY	103
8.3.11	Dimension table CTD.D_APPL	104
8.3.12	Dimension table CTD.D_APPLY	104
8.3.13	Dimension table CTD.D_TAPPLY	104
8.4	Data marts and reports	104
8.4.1	DB2 Data Mart	104
8.4.2	Reports	104

8.4.2.1	DB2 Buffer Pool Activity at the Application Level (Daily Data) - Health Check	104
8.4.2.2	DB2 Buffer Pool Activity at the Database Level (Daily Data) - Health Check	105
8.4.2.3	DB2 Buffer Pool Activity at the Database Partition Level (Daily Data) - Health Check	105
8.4.2.4	DB2 Buffer Pool Activity at the Tablespace Partition Level (Daily Data) - Health Check	106
8.4.2.5	DB2 Connections Used (Daily Data) - Summary	106
8.4.2.6	DB2 Database Primary Log Usage (Daily Data) - Health Check	107
8.5	How to enable new data in Tivoli Enterprise Data Warehouse when creating new resource models in IBM Tivoli Monitoring.....	107

1 About this document

This document describes the warehouse enablement pack for IBM Tivoli Monitoring for Databases, Version 5.1.1 : DB2. It covers the following topics:

- Installing and configuring the warehouse pack
- The data flow and data structures used by the warehouse pack

Resource models are designed to monitor and report critical system metrics, such as disk utilization. This warehouse enablement pack provides a data mart ETL that is used to package metric data reported by IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 resource models into a database arrangement known as star schemas. Users may create historical reports of metrics, using Crystal Enterprise or any other OLAP tool.

This warehouse pack relies on other IBM Tivoli products, including IBM Tivoli Monitoring, which are prerequisite to its use. The prerequisites are explained later this document.

1.1 Related Documentation

You can access many Tivoli® publications online using the Tivoli Information Center, which is available on the Customer Support Web site

<http://www.tivoli.com/support/documents/>

The following sets of documentation are available to help you understand, install, and manage this warehouse pack:

- Tivoli Enterprise™ Data Warehouse
- IBM DB2 Universal Database Enterprise Edition, DB2 Universal Database Data Warehouse Center, and DB2 Universal Database Warehouse Manager

The following sections list and briefly describe these libraries.

1.1.1 Tivoli Enterprise Data Warehouse

The following Tivoli Enterprise Data Warehouse documents are available on the Tivoli Enterprise Data Warehouse Documentation CD:

Tivoli Enterprise Data Warehouse Release Notes, GI11-0857

Provides late-breaking information about Tivoli Enterprise Data Warehouse and lists hardware requirements and software prerequisites.

Installing and Configuring Tivoli Enterprise Data Warehouse, GC32-0744

Describes how Tivoli Enterprise Data Warehouse fits into your enterprise, explains how to plan for its deployment, and gives installation and configuration instructions. It provides an introduction to the built-in program for creating and running reports, and contains maintenance procedures and troubleshooting information.

Enabling an Application for Tivoli Enterprise Data Warehouse, GC32-0745

Provides information about connecting an application to Tivoli Enterprise Data Warehouse. This book is for application programmers who use Tivoli Enterprise Data Warehouse to store and report on their application's data, data warehousing experts who import Tivoli Enterprise Data Warehouse data into business intelligence applications, and customers who use their local data in the Tivoli Enterprise Data Warehouse.

1.1.2 IBM DB2 Universal Database, DB2 Universal Database Data Warehouse Center, and DB2 Universal Database Warehouse Manager

The DB2 Universal Database library contains important information about the database and data warehousing technology provided by IBM DB2 Universal Database, DB2 Universal Database Data Warehouse Center, and DB2 Universal Database Warehouse Manager. Refer to the DB2 Universal Database library for help in installing, configuring, administering, and troubleshooting DB2 Universal Database. The DB2 Universal Database library is

available on the Tivoli Customer Support Web site. After you install DB2 Universal Database, its library is also available on your system.

The following DB2 Universal Database documents are particularly relevant for people working with Tivoli Enterprise Data Warehouse:

IBM DB2 Universal Database for Windows Quick Beginnings, GC09-2971

Guides you through the planning, installation, migration (if necessary), and setup of a partitioned database system using the IBM DB2 Universal Database product on Microsoft® Windows®.

IBM DB2 Universal Database for UNIX Quick Beginnings, GC09-2970

Guides you through the planning, installation, migration (if necessary), and setup of a partitioned database system using the IBM DB2 Universal Database product on UNIX®.

IBM DB2 Universal Database Administration Guide: Implementation, SC09-2944

Covers the details of implementing your database design. Topics include creating and altering a database, database security, database recovery, and administration using the Control Center, a DB2 Universal Database graphical user interface.

IBM DB2 Universal Database Data Warehouse Center Administration Guide, SC26-9993

Provides information on how to build and maintain a data warehouse using the Data Warehouse Center.

IBM DB2 Warehouse Manager Installation Guide, GC26-9998

Provides the information to install the following Warehouse Manager components: Information Catalog Manager, warehouse agents, and warehouse transformers.

IBM DB2 Universal Database and DB2 Connect Installation and Configuration Supplement, GC09-2957

Provides advanced installation considerations and guides you through the planning, installation, migration (if necessary), and set up a platform-specific DB2 Universal Database client. Once the DB2 Universal Database client is installed, you then configure communications for both the client and server, using the DB2 Universal Database GUI tools or the Command Line Processor. This supplement also contains information on binding, setting up communications on the server, the DB2 Universal Database GUI tools, DRDA™ AS, distributed installation, the configuration of distributed requests, and accessing heterogeneous data sources.

IBM DB2 Universal Database Message Reference Volume 1, GC09-2978 and *IBM DB2 Universal Database Message Reference Volume 2*, GC09-2979

Lists the messages and codes issued by DB2 Universal Database, the Information Catalog Manager, and the Data Warehouse Center, and describe the actions you should take.

1.1.3 IBM Tivoli Monitoring for Databases: DB2

The following IBM Tivoli Monitoring for Databases: DB2 documents are available on the IBM Tivoli Monitoring for Databases: DB2 documentation CD:

- *IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 User's Guide*
- Provides information about how to install and use the IBM Tivoli Monitoring for Databases: DB2 software to manage DB2 database resources with Tivoli Enterprise software.
- *IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 Reference Guide*
- Provides task and resource model information about procedures described in the *IBM Tivoli Monitoring for Databases, Version 5.1.1: DB2 User's Guide*.
- *IBM Tivoli Monitoring for Databases: Installation and Setup Guide*
- Provides information on how to install the IBM Tivoli Monitoring for Databases: DB2 software.

- *IBM Tivoli Monitoring for Databases, Version 5.1.0 Release Notes*, GI11-0933

Describes product features and provides information about the latest changes to the installation requirements and procedures. The release notes also describe known limitations related to installation and explain how to work around each limitation.

- *IBM Tivoli Monitoring for Databases: DB2 Limitations and Workarounds Supplement*, SC23-4786

Provides the latest information about known product limitations and workarounds. To ensure that the information is the latest available, this document is provided only on the Web, where it is updated on a regular basis. You can access the Limitations and Workarounds document through the IBM Tivoli Monitoring for Databases: DB2 link on the Tivoli Information Center Web site:

http://www.tivoli.com/support/public/Prodman/public_manuals/td/TD_PROD_LIST.html

1.1.4 Tivoli Enterprise Data Warehouse

The following Tivoli Enterprise™ Data Warehouse documents are available on the Tivoli Enterprise Data Warehouse Documentation CD:

- *Tivoli Enterprise Data Warehouse Release Notes*, GI11-0857

Provides late-breaking information about Tivoli Enterprise Data Warehouse and lists hardware requirements and software prerequisites.

- *Installing and Configuring Tivoli Enterprise Data Warehouse*, GC32-0744

Describes how Tivoli Enterprise Data Warehouse fits into your enterprise, explains how to plan for its deployment, and gives installation and configuration instructions. It provides an introduction to the built-in program for creating and running reports, and contains maintenance procedures and troubleshooting information.

- *Enabling an Application for Tivoli Enterprise Data Warehouse*, GC32-0745

Provides information about connecting an application to Tivoli Enterprise Data Warehouse. This book is for application programmers who use Tivoli Enterprise Data Warehouse to store and report on their application's data, data warehousing experts who import Tivoli Enterprise Data Warehouse data into business intelligence applications, and customers who use their local data in Tivoli Enterprise Data Warehouse.

1.1.5 DB2

The DB2 library contains important information about the database and data warehousing technology provided by IBM DB2, DB2 Data Warehouse Center, and DB2 Warehouse Manager. Refer to the DB2 library for help in installing, configuring, administering, and troubleshooting DB2. The DB2 library is available on the Tivoli Customer Support Web site. After you install DB2, its library is also available on your system.

The following DB2 documents are particularly relevant for people working with Tivoli Enterprise Data Warehouse:

- *IBM DB2 Universal Database for Windows Quick Beginnings*, GC09-2971

Guides you through the planning, installation, migration (if necessary), and setup of a partitioned database system using the IBM DB2 product on Microsoft Windows.

- *IBM DB2 Universal Database for UNIX Quick Beginnings*, GC09-2970

Guides you through the planning, installation, migration (if necessary), and setup of a partitioned database system using the IBM DB2 product on UNIX.

- *IBM DB2 Universal Database Administration Guide: Implementation*, SC09-2944

Covers the details of implementing your database design. Topics include creating and altering a database, database security, database recovery, and administration using the Control Center, a DB2 graphical user interface.

- *IBM DB2 Universal Database Data Warehouse Center Administration Guide*, SC26-9993

Provides information on how to build and maintain a data warehouse using the Data Warehouse Center.

- *IBM DB2 Warehouse Manager Installation Guide*, GC26-9998

Provides the information to install the following Warehouse Manager components: Information Catalog Manager, warehouse agents, and warehouse transformers.

- *IBM DB2 Universal Database and DB2 Connect Installation and Configuration Supplement*, GC09-2957

Provides advanced installation considerations and guides you through the planning, installation, migration (if necessary), and set up a platform-specific DB2 client. Once the DB2 client is installed, you then configure communications for both the client and server, using the DB2 GUI tools or the Command Line Processor. This supplement also contains information on binding, setting up communications on the server, the DB2 GUI tools, DRDA™ AS, distributed installation, the configuration of distributed requests, and accessing heterogeneous data sources.

- *IBM DB2 Universal Database Message Reference Volume 1*, GC09-2978 and *IBM DB2 Universal Database Message Reference Volume 2*, GC09-2979

Lists the messages and codes issued by DB2, the Information Catalog Manager, and the Data Warehouse Center, and describes the actions you should take.

2 Overview

The following sections provide an overview of Tivoli Enterprise Data Warehouse and the IBM Tivoli Monitoring for Databases, Version 5.1.2: DB2 warehouse pack.

2.1 Overview of Tivoli Enterprise Data Warehouse

Tivoli Enterprise Data Warehouse provides the infrastructure for the following:

- Extract, transform, and load (ETL) processes through the IBM DB2 Data Warehouse Center tool
- Schema generation of the central data warehouse
- Historical reporting

As shown in Figure 1, Tivoli Enterprise Data Warehouse consists of a centralized data store where historical data from many management applications can be stored, aggregated, and correlated.

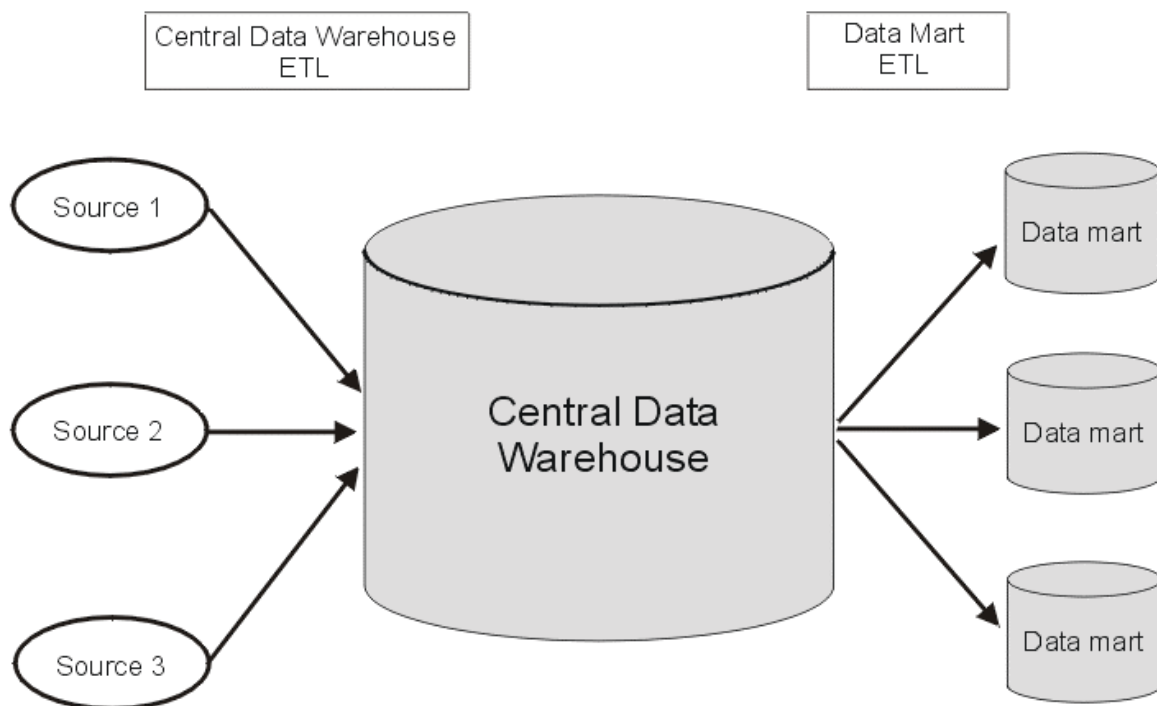


Figure 1. Tivoli Enterprise Data Warehouse overview

The *central data warehouse* uses a generic schema that is the same for applications. As new components or new applications are added, more data is added to the database; however, no new tables or columns are added in the schema.

A *data mart* is a subset of a data warehouse that contains data tailored and optimized for the specific reporting needs of a department or team.

The *central data warehouse ETL* reads the data from the operational data stores of the application that collects it, verifies the data, makes the data conform to the schema, and places the data into the central data warehouse.

The *data mart ETL* extracts a subset of data from the central data warehouse, transforms it, and loads it into one or more star schemas, which can be included in data marts to answer specific business questions.

A program that provides these ETLs is called a *warehouse enablement pack*, or *warehouse pack*.

2.2 Overview of the DB2 Warehouse Enablement Pack

The IBM Tivoli Monitoring for Databases: DB2 warehouse enablement pack works with the Tivoli Enterprise Data Warehouse and other IBM Tivoli software.

- The data source consists of the resource models provided by IBM Tivoli Monitoring for Databases, Version 5.1.2: DB2 in conjunction with IBM Tivoli Monitoring. These resource models monitor and report selected metrics of a DB2 database server.
- The central data warehouse ETL is provided by the IBM Tivoli Monitoring - Generic ETL1 Warehouse Enablement Pack. This ETL may be used by more than one warehouse pack, for example, IBM Tivoli Monitoring for Databases: Informix, and IBM Tivoli Monitoring for Databases: Oracle.
- The IBM Tivoli Monitoring for Databases, Version 5.1.2: DB2 Warehouse Enablement Pack provides the data mart ETL.
- The IBM Tivoli Enterprise Data Warehouse provides the central data warehouse and data mart databases.

You can create historical reports of metrics using Crystal Enterprise or any other OLAP tool.

The warehouse pack provides the following components:

- Data mart ETL. The data mart ETL is utilized to load the DB2 star schemas in the data mart database from the DB2 data stored in the central data warehouse database.
- Star schemas. There are hourly/daily/weekly and monthly star schemas for DB2 components. A DB2 component is a DB2 managed element in IBM Tivoli Monitoring.
- Data mart. The Data Mart database will contain all the DB2 star schemas.
- Reports. Crystal reports are predefined which represent the historical values of selected metrics.

You may use the Crystal Enterprise Server to customize the prepackaged reports as desired, or you may choose to use them as-is.

2.3 Typical usage scenario

The DB2 warehouse enablement pack is typically used as follows:

1. The IBM Tivoli Monitoring operator or other authorized user chooses appropriate IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 resource models, enable the Tivoli Enterprise Data Warehouse logging option and then distributes a profile containing the resource models. The resource models chosen depend upon which metrics he wants to create reports on. For further information, refer to the IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 User's Guide.
2. The resource models run and store data in the endpoint database.
3. The operator uses the IBM Tivoli Monitoring **wdmcollect** command to schedule and upload the collection of the data into a unique relational database management system (RDBMS) used by IBM Tivoli Monitoring warehouse enablement pack.
4. The IBM Tivoli Monitoring, Version 5.1.1 central data warehouse ETL is run, which inserts historical data into the central data warehouse from this IBM Tivoli Monitoring RDBMS.
5. The DB2 warehouse pack's data mart ETL is run, which populates the DB2 star schemas.
6. The operator utilizes either its own OLAP tool or the Tivoli Reporting Interface interface to run and optionally print selected reports.

The IBM Tivoli Monitoring, Version 5.1.1 central data warehouse ETL does the following:

1. Copies IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 resource model data from the Tivoli Monitoring's RDBMS storage area into temporary staging tables in the Tivoli Enterprise Data Warehouse central data warehouse.

2. Data is extracted from these tables and is placed into the Tivoli Enterprise Data Warehouse central data warehouse.

The IBM Tivoli Monitoring, Version 5.1.1 central data warehouse ETL may also be utilized to do similar steps for other warehouse packs, such as IBM Tivoli Monitoring for Databases: Oracle.

The central data warehouse ETL is named *AMX_co5_ETL1_Process*.

The DB2 warehouse pack's data mart ETL does the following:

1. Extracts data from the Tivoli Enterprise Data Warehouse central data warehouse and pushes it into the data mart database that contains DB2 hourly star schemas.
2. Rolls-up the hourly data and populates the daily, weekly, and monthly star schemas.

The data mart ETL is named *CTD_m05_ETL2_Process*.

3 Installing and configuring

This section describes installation of the warehouse enablement pack. Please refer to the *Installing and Configuring Tivoli Enterprise Data Warehouse* manual for related information.

3.1 Prerequisites

Before installing the warehouse pack you should have already installed the following products:

- IBM Tivoli Monitoring, Version 5.1.1
- IBM Tivoli Monitoring 5.1.1 – Tivoli Enterprise Data Warehouse Support
- IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2
- Tivoli Enterprise Data Warehouse, Version 1.1, fix pack 3 (1.1-TDW-FP03)
- IBM Tivoli Monitoring - Generic Central Data Warehouse ETL Enablement Pack v5.1.1
- IBM Tivoli Monitoring - Generic Central Data Warehouse ETL Enablement Pack Fixpack 6 (5.1.1-ITM-FP06)

3.2 Supported hardware and software

The IBM Tivoli Monitoring for Databases: DB2, Warehouse Enablement Pack Implementation Guide, Version 1.1.0.1, supports IBM Tivoli Monitoring for Databases: DB2, Version 5.1.0. It supports all versions of DB2, Informix, Microsoft SQL Server, Oracle, and Sybase database products for the ITM source database as documented in the IBM Tivoli Monitoring 5.1.1 Release Notes, G110-5797-01. It only support DB2 for the Central Data warehouse and Data Mart databases as documented in the Tivoli Enterprise Data Warehouse 1.1 Release Notes.

3.3 Limitations

This warehouse pack must be installed using the user "db2". If that is not the user name used when installing the Tivoli Enterprise Data Warehouse core application, you must create a user temporary tablespace for use by the installation program. The user temporary tablespace that is created in each central data warehouse database and data mart database during the installation of Tivoli Enterprise Data Warehouse is accessible only to the user that performed the installation.

If you are installing the warehouse pack using the same database user that installed Tivoli Enterprise Data Warehouse, or if your database user has access to another user temporary tablespace in the target databases, no additional action is required.

If you do not know the user name that was used to install Tivoli Enterprise Data Warehouse, you can determine whether the tablespace is accessible by attempting to declare a temporary table while connected to each database as the user that will install the warehouse pack. The following commands are one way to do this:

```
db2 "connect to TWH_CDW user installing_user using password"
```

```
db2 "declare global temporary table t1 (c1 char(1))with replace on commit preserve rows not logged"
```

```
db2 "disconnect TWH_CDW"
```

```
db2 "connect to TWH_MART user installing_user using password"
```

```
db2 "declare global temporary table t1 (c1 char(1))with replace on commit preserve rows not logged"
```

```
db2 "disconnect TWH_CDW"
```

Where:

installing_user Identifies the database user that will install the warehouse pack.

password Specifies the password for the installing user.

If the declare command is successful, the specified database user can install the warehouse pack. No additional action is required.

If the declare command fails, run the following DB2 commands to create a new tablespace for the installation in both the central data warehouse database and data mart databases:

```
db2 "connect to TWH_CDW user installing_user using password"
db2 "create user temporary tablespace usertmp2 managed by system using (' usertmp2)"
db2 "disconnect TWH_CDW"
db2 "connect to TWH_MART user installing_user using password"
db2 "create user temporary tablespace usertmp3 managed by system using ('
usertmp3') "
db2 "disconnect TWH_MART"
```

Where:

installing_user Identifies the database user that will install the warehouse pack.

password Specifies the password for the installing user.

3.4 Database sizing considerations

Ensure that you have sufficient space in the central data warehouse database for the historical data collected by this warehouse pack. To estimate how much space is required for the WebLogic warehouse pack, complete the following worksheets.

This estimation is based on running the actual DB2 resource models provided by IBM Tivoli Monitoring for Databases, v5.1.0: DB2

TWH_CDW estimation

Object Managed	Number of Components	Component space	Measurements per Day space
DB2 Host	num_host	1183	0
DB2 Instance	num_inst	1741	24480
DB2 Database	num_db	625	183600
DB2 Table	num_table	1183	4896
DB2 Database Partition	num_node	1741	171360
DB2 Tablespace Partition	num_tbsp	625	93024
DB2 Apply	num_apply	625	4896
DB2 Table Apply	num_table_apply	1183	12240
DB2 Application	num_appl_	625	53856
DB2 Gateway	num_gtwy	625	19584

Total	Σ total components	Σ Components	Σ Measurements
-------	---------------------------	---------------------	-----------------------

Number of Days	num_days
Estimate TWH_CDW database size in megabytes	$(\Sigma \text{Components} + (\Sigma \text{Measurements} * \text{num_days})) * 1.2 / 1024000$

Object Managed	Total Components	Component space	Measurements per Day space
DB2 Host	1	1183	
DB2 Instance	10	17410	244800
DB2 Database	10	6250	1836000
DB2 Table	150	177450	734400
DB2 Database Partition	10	17410	1713600
DB2 Tablespace Partition	20	12500	1860480
DB2 Apply	1	625	4896
DB2 Table Apply	50	59150	612000
DB2 Application	5	3125	269280
DB2 Gateway	1	625	19584
Total	258	295728	7295040

Number of Days	180
Estimate TWH_CDW database size in megabytes	1540

TWH_MART estimation

Object Managed	Total Components	Metrics	Dimension space	Translation Dimension Space
DB2 Host	<i>num_host</i>		<i>num_unique_host</i> * 400	<i>num_host</i> * 949
DB2 Instance	<i>num_inst</i>	<i>num_met</i>	<i>num_unique inst</i> * 1070	<i>num_inst</i> * 951
DB2 Database	<i>num_db</i>	<i>num_met</i>	<i>num_unique db</i> * 803	<i>num_db</i> * 819

DB2 Table	<i>num_table</i>	<i>num_met</i>	<i>num_unique_table</i> * 935	<i>num_table</i> * 951
DB2 Database Partition	<i>num_node</i>	<i>num_met</i>	<i>num_unique_node</i> * 1202	<i>num_node</i> * 1083
DB2 Tablespace Partition	<i>num_tbsp_node</i>	<i>num_met</i>	<i>num_unique_tbsp_node</i> * 803	<i>num_tbsp_node</i> * 819
DB2 Apply	<i>num_apply</i>	<i>num_met</i>	<i>num_unique_apply</i> * 803	<i>num_apply</i> * 819
DB2 Table Apply	<i>num_table_apply</i>	<i>num_met</i>	<i>num_unique_table_apply</i> * 935	951
DB2 Application	<i>num_appl_</i>	<i>num_met</i>	<i>num_unique_appl</i> * 803	819
DB2 Gateway	<i>num_gtwy</i>	<i>num_met</i>	<i>num_unique_gtwy</i> * 803	819
Total	Σtotal components		ΣDimension Space	ΣDimension Translation Space

Num_unique_object means that 2 objects with the same name but 2 different parents will exist only once in the dimension table for that object. On the contrary, the translation table will contain all the objects original identifiers from the TWG.CDW and the corresponding translated object identifiers in the dimension table.

For instance a table SYS.CAT may exist in all the DB2 databases of the enterprise. The components SYS.CAT of type CTD_TABLE will exist as separate objects in the TWG.COMP table for each database they belong to. However, only one entry will exist in the CTD.D_TABLE dimension table.

Object Managed	Fact_hour space for num days (col4)	Fact_day space for num days (col5)	Fact_week space for num days	Fact_month space for num days
DB2 Instance	<i>num_inst</i> * <i>num_met</i> * <i>num_days</i> * 24 * 225	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Database	<i>num_db</i> * <i>num_met</i> * <i>num_days</i> * 24 * 245	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Table	<i>num_table</i> * <i>num_met</i> * <i>num_days</i> * 24 * 266	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Database Partition	<i>num_node</i> * <i>num_met</i> * <i>num_days</i> * 24 * 296	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Tablespace Partition	<i>num_tbsp_node</i> * <i>num_met</i> * <i>num_days</i> * 24 * 327	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Apply	<i>num_apply</i> * <i>num_met</i> * <i>num_days</i> * 24 * 266	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30
DB2 Table Apply	<i>num_table_apply</i> * <i>num_met</i> * <i>num_days</i>	<i>Col4</i> / 24	<i>Col5</i> / 7	<i>Col5</i> / 30

	* 24 * 296			
DB2 Application	$num_appl * num_met * num_days * 24 * 266$	Col4 / 24	Col5 / 7	Col5 / 30
DB2 Gateway	$num_gtwy * num_met * num_days * 24 * 245$	Col4 / 24	Col5 / 7	Col5 / 30
Total	ΣFact Hour Space	ΣFact Day Space	ΣFact Week Space	ΣFact Month Space

Number of Days	num_days
Estimate TWH_MART database size in megabytes	(Σ Dimension space + Σ Translation Dimension space + Σ Fact_hour_space + Σ Fact_day_space + Σ Fact_week_space + Σ Fact_month_space) * 1.2 / 1024000

Object Managed	Total Components	Metrics	Dimension space	Translation Dimension Space
DB2 Host	1	0	400	949
DB2 Instance	10	10	10700	9510
DB2 Database	10	75	8030	8190
DB2 Table	150	2	140250	142650
DB2 Database Partition	10	70	12020	10830
DB2 Tablespace Partition	20	38	16060	16380
DB2 Apply	1	2	803	819
DB2 Table Apply	50	5	46750	47550
DB2 Application	5	22	4015	4095
DB2 Gateway	1	8	803	819
TOTAL			239831	241792

Object Managed	Fact_hour space for 180 days (col4)	Fact_day space for 180 days (col5)	Fact_week space for 180 days	Fact_month space for 180 days
DB2 Host	0	0	0	0

DB2 Instance	9720000	405000	57857	13500
DB2 Database	79380000	3307500	472500	110250
DB2 Table	2298240	95760	13680	3192
DB2 Database Partition	89510400	3729600	532800	124320
DB2 Tablespace Partition	53680320	2236680	319526	74556
DB2 Apply	2298240	95760	13680	3192
DB2 Table Apply	6393600	266400	38057	8880
DB2 Application	25280640	1053360	150480	35112
DB2 Gateway	8467200	352800	50400	11760
TOTAL	277028640	11542860	1648980	384762

Number of Days	180
Estimate TWH_MART database size in megabytes	$(239831 + 241792 + 277028640 + 11542860 + 1648980 + 384762) * 1.2 / 1024000 = 341$

3.5 Data sources and targets

The data source for the DB2 warehouse pack is the data in the central data warehouse database inserted by the *IBM Tivoli Monitoring – Generic Central Data Warehouse ETL*. The primary target for the warehouse pack is the star schemas in the data mart database. Another target is the control database. These are represented in the DB2 Data Warehouse Center as CTD_TWH_CDW_Source, CTD_TWH_MART_Target, and the CTD_TWH_MD_Target, respectively.

3.6 Pre-installation steps

Before installing, check that your control server has connectivity to databases control database, central data warehouse, and data mart. These reside on your control server, the central data warehouse server, and the data mart server, respectively. These servers may be installed on one system in a single system installation, or distributed across two or three systems in your enterprise.

Perform the following steps to check database connectivity:

1. Click **Start** and select **Programs**, **IBM DB2**, and **Client Configuration Assistant** on the Windows taskbar of your control server.
2. Click the **Database alias** tab.
3. Right click the database name.
4. Click the **Test** button on the right-hand side of the panel.
5. Enter the database User ID and Password.
6. Click **OK**.

If you do not have connectivity, use the Client Configuration Assistant to establish a connection. Refer to the Client Configuration Assistant's online help or printed DB2 documentation for instructions.

3.7 Installation procedure

To install IBM Tivoli Monitoring for Databases: DB2 warehouse enablement pack, perform the following steps:

1. Make sure that all prerequisite products are installed.
2. Be sure that all prerequisite product patches are applied.
3. Perform the pre-installation steps as described in the section Pre-Installation steps.
4. Install the warehouse pack as described in the *Installing and Configuring Tivoli Enterprise Data Warehouse* manual.
The installation media for this warehouse pack is located on the IBM Tivoli Monitoring for Databases, Version 5.1.0: DB2 Component Software CD in the /TEDW/tedw_apps_etl/ctd directory.

The warehouse pack should be installed on the Tivoli Enterprise Data Warehouse control server in your enterprise. Use the Tivoli Enterprise Data Warehouse installation program to install the warehouse pack. Be sure to select Application Installation Only in the setup type window.

During installation you are asked for the path to the installation media for the warehouse pack. Specify the complete path to the tedw_apps_etl directory; for example, on Microsoft® Windows NT this might be R:\TEDW\tedw_apps_etl, where R is the drive name of the CDROM in your computer.

If an installation error occurs, review the installation log for the cause of the error. On Windows NT and Windows 2000, the application installation log is named %TWH_TOPDIR%\apps\ctd\TWHApp.log. Occasionally, it may direct you to another log, %TEMP%\twh_ibm_db2_runlog.log. Correct the source of the error and then rerun the installation.

5. Perform the post-installation steps described in the next section.

3.8 Post-installation steps

Proceed with the *Installing and Configuring Tivoli Enterprise Data Warehouse* manual. The following five items should be kept in mind as you proceed through the chapter:

Properties for Data Sources and Targets

After you install the warehouse enablement pack, use the procedures in *Installing and Configuring Tivoli Enterprise Data Warehouse* to use the Data Warehouse Center to perform the following configuration tasks for data sources and targets:

1. Make sure the control database is set to control database.
2. Specify the properties for the **CTD_TWH_MART_Source** data source. These properties are in the Data Source page.
 - Do not change the value of the Data Source field. It must be data mart.
 - Set the User ID field to the Instance name for the configuration repository. The default value is db2admin
 - Set the Password field to the password used to access the data mart database.
3. Specify the properties for the **CTD_TWH_CDW_Source** data source. These properties are in the Data Source page.
 - Do not change the value of the Data Source field. It must be central data warehouse.
 - Set the User ID field to the Instance name for the configuration repository. The default value is db2admin
 - Set the Password field to the password used to access the central data warehouse database.
4. Specify the properties for the warehouse target **CTD_TWH_MART_Target**. These properties are in the Database page.

- Do not change the value of the Data Source field. It must be data mart.
 - Set the User ID field to the user ID used to access the Tivoli Enterprise Data Warehouse mart database. The default value is db2admin.
 - In the Password field, type the password used to access the central data mart database.
5. Specify the properties for the warehouse target **CTD_TWH_CDW_Target**. These properties are in the Database page.
 - Do not change the value of the Data Source field. It must be central data warehouse.
 - Set the User ID field to the user ID used to access the Tivoli Enterprise Data Warehouse central data warehouse database. The default value is db2admin.
 - In the Password field, type the password used to access the central data warehouse database.
 6. Specify the following properties for the warehouse target **CTD_TWH_MD_Target**. These properties are in the Database page.
 - Do not change the value of the Data Source field. It must be control database.
 - Set the User ID field to the user ID used to access the control database. The default value is db2admin.
 - In the Password field, type the password used to access the control database.
 6. Update the Report Execution Engine (REE) Agent Site. Perform the following steps to make the update:
 1. Select **Administration** from the Data Warehouse Center.
 2. Select **Agent Sites**
 3. Select **REE Agent Site**
 4. Right click on the **REE Agent Site** and select **Properties**
 5. Update the host name, user ID, and password

Note: Indicate the name of your control server for the host name, because localhost is not supported. Also, the userID connects to the control database.
 7. Specify dependencies between processes and schedule processes that are to run automatically. The processes for this warehouse pack are located in the CTD_IBM_Tivoli_Monitoring_for_Databases:_DB2_v5.1.0_Subject_Area subject area. The processes should be run in the following order:
 - AMX_IBM_Tivoli_Monitoring_v5.1.1_Subject_Area
 - CTD_IBM_Tivoli_Monitoring_for_Databases:_DB2_v5.1.0_Subject_Area

Additional ODBC connections

No additional ODBC connections need to be specified for the warehouse pack.

ETL Initialization processes

The warehouse pack's data mart ETL has no required initialization process.

ETL Process Dependencies

The warehouse pack's data mart ETL is dependent upon the IBM Tivoli Monitoring, Version 5.1.1 central data warehouse ETL. For further information regarding the central data warehouse ETL, see the *IBM Tivoli Monitoring: Generic Central Data Warehouse ETL Warehouse Enablement Pack Implementation Guide* manual.

ETL Process Steps

The process steps for the warehouse pack's data mart warehouse ETL are CTD_m05_s002_Update_Step, CTD_m05_s005_Prepere_Stage_Step, CTD_m05_s020_Mart_Pre_Extract_Step, CTD_m05_s020_Mart_Extract_Step, CTD_m05_s030_Mart_Load_Step, CTD_m05_s040_Mart_Rollup_Step and CTD_m05_s050_Mart_Prune_Step.

4 Maintaining

4.1 *Backing up and restoring*

The **dbrest.bat** (that is calling the **dbrest.sh**) script in the misc\tools directory is an example script that shows you how to restore these 3 databases on a NT/2000 box.

4.2 *Pruning*

If you have established a schedule to automatically run the warehouse pack's data mart ETL process on a periodic basis, occasionally manually prune the logs in the directory %VWS_LOGGING%.

The CTD_m05_s050_Mart_Prune step prunes the hourly, daily, weekly, and monthly fact tables as soon as they have data older than 3 months.

If you schedule the data mart ETL process to run daily, as recommended, you do not need to schedule pruning separately.

4.3 *Other Tools*

The **CTD_reset_extract_window.bat** script resets the Extract Control window for IBM Tivoli Monitoring for Databases: DB2, warehouse enablement pack. You should use this script only to restart the Extract Control window for the CTD_m05_ETL2_Process. If you want to reset the window to the last extract, use the extract_log to get the last values of each DB2 (CTD) extracts.

The **CTD_m05_ETL2_Process.bat** script executes the CTD_m05_ETL2_Process from the command line.

The **extract_win.bat** script shows the TWG.extract_control and TWG.extract_log windows reformatted in the integer sequence (intseq). It also reformats the columns.

The **ctd_verify_etl.sh** script checks that all the DB2 data that is in the central data warehouse database has been loaded in the data mart.

5 ETL processes

This warehouse pack has the following processes.

5.1 ETL Processes

With the new DB2 Warehouse Enablement Fixpack v1.1.0.1, 2 processes exist:

- CTD_m05_ETL2_Process
- CTD_m10_Reset_ETL2_Process

During the installation of the Fixpack v1.1.0.1, all the tables under the schema CTD in the data mart database are renamed into CTD.X<tablename> to preserve data already loaded using the CTD Warehouse Enablement Package v1.1.0.

New tables, with new columns and schema CTD are created. They will be populated with the last 3 months of data from the central data warehouse database.

Since the extraction of the measurement data is based on the prune parameter, which is by default set to 3 months, only the last 3 months of data will be re-extracted. If you want to re-extract more data, you must set accordingly the prune parameter in the CTD.PRUNE_MART_CONTROL table in the data mart database, before executing the CTD_m05_ETL2_Process.

5.2 CTD_m05_ETL2_Process

The CTD_m05_ETL2_Process extracts the DB2 data from the central data warehouse database, transforms it and loads it into the DB2 dimension and fact tables in the data mart database. This process should be run once a day and should be ran after the AMX_c05_ETL1_Process that extracts data from the IBM Tivoli Monitoring RIM database and transforms and loads it into the central data warehouse database.

This process has the following steps:

□ CTD_m05_s002_Update_Step

This step created and initialize the table CTD.PRUNE_MART_CONTROL in the central data warehouse database. This table is the exact copy of the CTD.PRUNE_MART_CONTROL table in the data mart database. If you need to update this table, you should update it in the data mart database.

□ CTD_m05_s005_Prepare_Stage_Step

This step drops and creates the staging tables used by the subsequent steps.

□ CTD_m05_s020_Mart_Pre_Extract_Step

This step extracts the data from the central data warehouse database and places it into staging tables.

□ CTD_m05_s020_Mart_Extract_Step

This step populates "translation" dimension tables and stage fact tables. The translation dimension tables are used to track IDs from the original central data warehouse into the target Star Schema, so you can tell where the data actually came from when looking at a Star

Schema. Star Schemas re-assign IDs, so that data from various central data warehouse use the same IDs for the same thing. Note that the translation dimension tables are "permanent", while the stage fact table is dropped and recreated each time (in the pre-extract step)..

□ CTD_m05_s030_Mart_Load_Step

This step loads the data from the staging tables into the DB2 dimension and fact tables in the data mart database.

□ CTD_m05_s040_Mart_Rollup_Step

This step rolls up the DB2 hourly fact tables into daily, weekly, and monthly fact tables.

Once the hourly fact tables have been populated by the preceding step (CTD_m05_s030_Mart_Load_Step), the rollup step populates the daily, weekly and monthly fact tables in the data mart based on the data in the stage fact table. The stage fact table contains only the data of the current day.

The rollup step also populates the RPI.SSUpdated table in the control database to enable report scheduling. The report gets rerun when the runReport user-defined program is run if the following are true:

- ☐ The RPI.SSUpdated table has an entry for the star schema indicating that data is new
- ☐ When the user creates the report in the report graphical user interface, they selected the option to schedule reports

☐ CTD_m05_s050_Mart_Prune_Step

This step prunes the hourly, daily, weekly, and monthly fact tables as soon as they have data older than 3 months.

The three-month age is a parameter that is set into the table CTD.Prune_Mart_Control in the data mart database. The column PmartC_duration is a DB2 date duration. The amount of data to be pruned is based on the format 'yyyymmdd', so an entry of 300 indicates 3 months of data to be pruned.

☐ CTD_m05_s060_Mart_Schedule_Step

This step runs the schedule reports and saves the report output.

5.3 CTD_m10_Reset_ETL2_Process

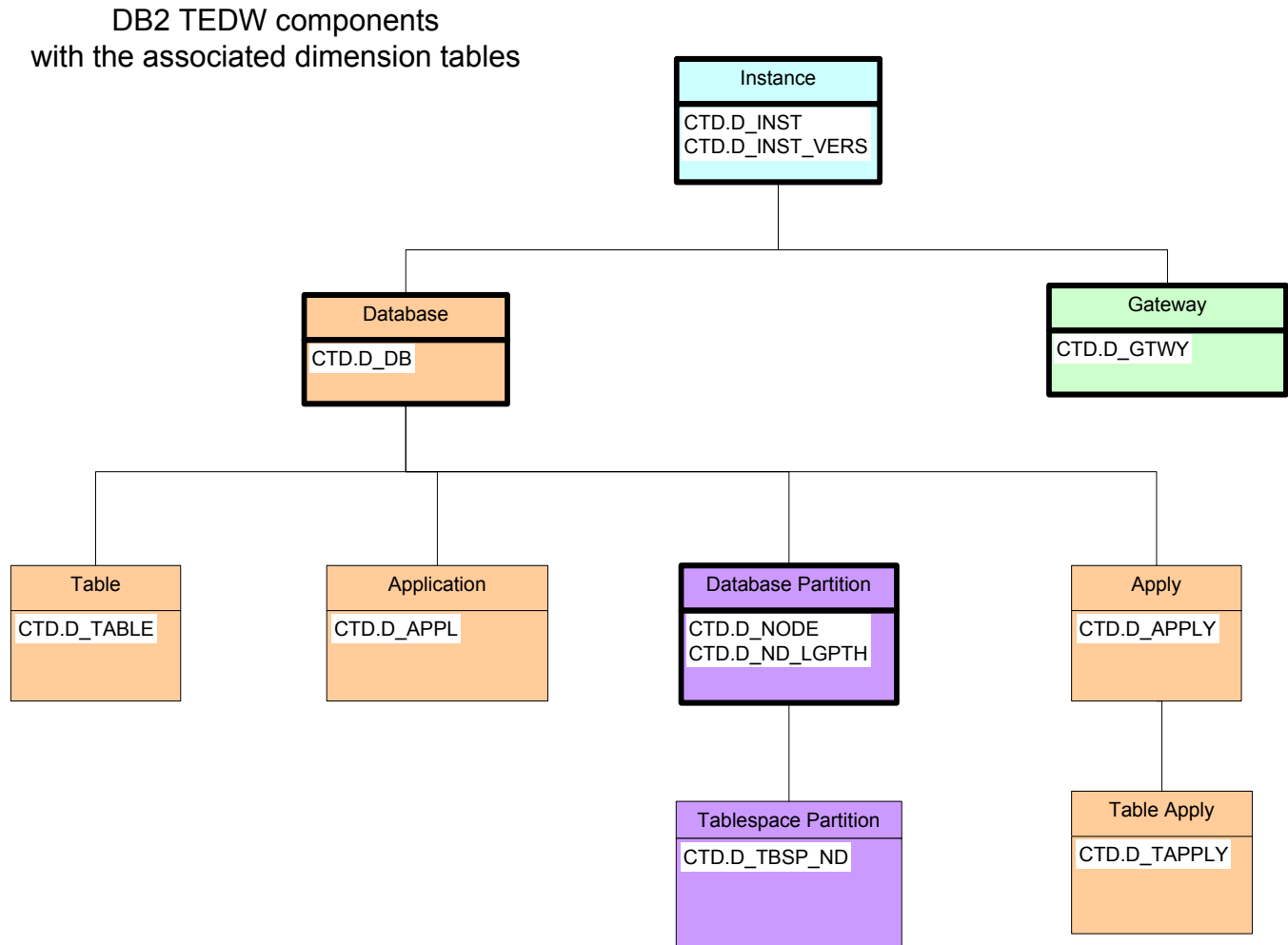
This Process is used only for maintenance purpose; it purges all the records from the DB2 star schemas and resets the extract functions in the TWG.Extract_Control table, so that the CTD_m05_ETL2_Process can be executed again, reloading all the data from the central data warehouse database. After executing this process, all the data previously loaded into the dimension and fact tables is erased, so you should use it very carefully and only after taking a full backup of the data mart database.

6 Generic schema implementation

Before reading this section, read about the generic schema for the Tivoli Enterprise Data Warehouse central data warehouse, which is described in *Enabling an Application for Tivoli Enterprise Data Warehouse*. That document defines the content of each table and explains the relationships between the tables in this document.

Shaded columns in the following tables are translated. These columns are also marked with an asterisk in the column heading. *Installing and Configuring Tivoli Enterprise Data Warehouse* contains instructions for installing support for additional languages.

The following graphic shows the DB2 object hierarchy.



6.1 Component configuration

6.1.1 Component type (table CompTyp)

CompTyp_Cd CHAR(17)	CompTyp_Parent_Cd CHAR(17)	CompTyp_Nm * VARCHAR(120)	CompTyp_Strt_DtTm TIMESTAMP	CompTyp_End_DtTm TIMESTAMP
CTD_APPL	NULL	DB2 Application	10/16/2003 5:32:08 PM	1/1/9999
CTD_APPLY_ID	NULL	DB2 Apply Program Identifier	10/16/2003 5:32:08 PM	1/1/9999
CTD_DATABASE	NULL	DB2 Distributed Database	10/16/2003 5:32:08 PM	1/1/9999
CTD_GATEWAY	NULL	DB2 DCS Database	10/16/2003 5:32:08 PM	1/1/9999
CTD_HOST	NULL	DB2 Host	10/16/2003 5:32:08 PM	1/1/9999
CTD_INSTANCE	NULL	DB2 Distributed Instance	10/16/2003 5:32:08 PM	1/1/9999
CTD_NODE	NULL	DB2 Database Partition	10/16/2003 5:32:08 PM	1/1/9999
CTD_TABLE	NULL	DB2 Table	10/16/2003 5:32:08 PM	1/1/9999
CTD_TABLE_APPLY	NULL	DB2 Table Level Apply	10/16/2003 5:32:08 PM	1/1/9999
CTD_TBSP	NULL	DB2 Table Space Partition	10/16/2003 5:32:08 PM	1/1/9999

An Apply is a running instance of the apply side of a replication process

A Table Level Apply is a replicated table in a context of a running apply instance.

6.1.2 Component (table Comp)

Comp_ID INTEGER	CompTyp_Cd CHAR (17)	Centr_Cd CHAR(6)	Cust_ID INTEGER	Comp_Corr_ID INTEGER	Comp_Nm VARCHAR (254)	Comp_Corr_Val VARCHAR (254)	Comp_Strt_Dt TIMESTAMP	Comp_End_Dt TIMESTAMP	Comp_Ds VARCHAR (254)
0	IP_HOST	CDW	1	0	cordoba.austin.ibm.com				IP_HOST
1	IP_HOST	CDW	1	0	toledo.austin.ibm.com				IP_HOST
2	CTD_INSTANCE	CDW	1	1	db2Inst1	db2Inst1			
3	CTD_DATABASE	CDW	1	2	MyDb1	MyDb1			
4	CTD_GATEWAY	CDW	1	2	MyDb1	MyDb1			
5	CTD_APPL	CDW	1	3	*LOCAL.db2.03101622	*LOCAL.db2.03101622			
6	CTD_APPLY_ID	CDW	1	3	2	2			
8	CTD_NODE	CDW	1	3	0	0			
9	CTD_NODE	CDW	1	3	1	1			
10	CTD_NODE	CDW	1	3	2	2			
11	CTD_TABLE	CDW	1	3	DB2@Table1	DB2@Table1			
15	CTD_TBSP	CDW	1	9	USERSPACE1	USERSPACE1			
16	CTD_TBSP	CDW	1	10	USERSPACE1	USERSPACE1			

6.1.3 Component relationship type (table RelnTyp)

RelnTyp_Cd CHAR(6)	RelnTyp_Nm * VARCHAR(120)
PCHILD	Parent Child Relation

6.1.4 Component relationship rule (table RelnRul)

CompTyp_Source_Cd CHAR(17)	CompTyp_Target_Cd CHAR(17)	ReInTyp_Cd CHAR(6)	ReInRul_Strt_DtTm TIMESTAMP	ReInRul_End_DtTm TIMESTAMP
CTD_APPLY_ID	CTD_TABLE_APPLY	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_DATABASE	CTD_APPL	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_DATABASE	CTD_APPLY_ID	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_DATABASE	CTD_NODE	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_DATABASE	CTD_TABLE	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_HOST	CTD_INSTANCE	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_INSTANCE	CTD_DATABASE	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_INSTANCE	CTD_GATEWAY	PCHILD	10/16/2003 5:32:12 PM	1/1/9999
CTD_NODE	CTD_TBSP	PCHILD	10/16/2003 5:32:12 PM	1/1/9999

6.1.5 Component relationship (table CompReIn)

CompReIn_ID INTEGER	Comp_Source_ID INTEGER	Comp_Target_ID INTEGER	ReInTyp_Cd CHAR(6)	CompReIn_Strt_DtTm TIMESTAMP	CompReIn_End_DtTm TIMESTAMP
1	1	2	PCHILD	2003-02-17-18.13.05.773000	9999-01-01-00.00.00.000000
2	2	3	PCHILD	2003-02-17-18.13.10.350000	9999-01-01-00.00.00.000000
3	2	4	PCHILD	2003-02-17-18.13.11.061000	9999-01-01-00.00.00.000000
4	3	5	PCHILD	2003-02-17-18.13.19.653000	9999-01-01-00.00.00.000000
5	3	6	PCHILD	2003-02-17-18.13.20.264000	9999-01-01-00.00.00.000000
6	3	8	PCHILD	2003-02-17-18.13.20.264000	9999-01-01-00.00.00.000000

8	3	9	PCHILD	2003-02-17- 18.13.22.317000	9999-01-01- 00.00.00.000000
9	3	10	PCHILD	2003-02-17- 18.13.22.317000	9999-01-01- 00.00.00.000000
10	3	11	PCHILD	2003-02-17- 18.13.25.191000	9999-01-01- 00.00.00.000000
11	9	15	PCHILD	2003-02-17- 18.13.32.031000	9999-01-01- 00.00.00.000000
12	10	16	PCHILD	2003-02-17- 18.13.36.447000	9999-01-01- 00.00.00.000000

6.1.6 Attribute type (table AttrTyp)

AttrTyp_Cd CHAR(17)	AttrTyp_Nm * VARCHAR(120)
CTD_APPL_NAME	DB2 Application Name
CTD_HOST	DB2 Instance Host Name (DB2 Catalog Node Host Name)
CTD_ISMPP	DB2 Parallel System
CTD_LOGPATH	DB2 Log Path
CTD_NODE_HOST_NM	DB2 Database Partition Host Name
CTD_NODE_NUM	DB2 Database Partition Number
CTD_NODE_PORT_NUM	DB2 Database Partition Port Number
NAME	Name
OWNER	Owner
VERSION	Version number

Remark : the DB2 Instance host name corresponds to the host name where the catalog node is located, so that a parallel database is only identified on one unique host name.

The following attributes have been deprecated in the Warehouse Enablement Package v1.1.0.1:

- CTD_APPLY_STATUS
- CTD_ASUB_STATUS
- CTD_BACKUP_TMSTMP
- CTD_CNTRSST_TM
- CTD_CONN_STATUS
- CTD_LAST_DB_BACKUP
- CTD_RESET_TMSTMP

- CTD_RESTORE_PEND
- CTD_RSTPND_STATUS
- CTD_TBSP_NAME

6.1.7 Attribute rule (table AttrRul)

CompTyp_Cd CHAR(17)	AttrTyp_Cd CHAR(17)	AttrRul_Strt_DtTm TIMESTAMP	AttrRul_End_DtTm TIMESTAMP	AttrRul_Dom_Ind CHAR
CTD_APPL	CTD_APPL_NAME	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N
CTD_HOST	AMX_EID	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_HOST	AMX_GMT_OFFSET	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_INSTANCE	CTD_HOST	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_INSTANCE	CTD_ISMPP	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N
CTD_INSTANCE	LAST_IP_ADDRESS	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_INSTANCE	VERSION	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_NODE	CTD_LOGPATH	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_NODE	CTD_NODE_HOST_NM	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_NODE	CTD_NODE_NUM	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_NODE	CTD_NODE_PORT_NUM	2/2/2004 8:47:44 PM	1/1/9999 12:00:00 PM	N
CTD_TABLE	NAME	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N
CTD_TABLE	OWNER	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N
CTD_TABLE_APPLY	NAME	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N
CTD_TABLE_APPLY	OWNER	10/16/2003 5:32:13 PM	1/1/9999 12:00:00 PM	N

6.1.8 Attribute domain (table AttrDom)

This warehouse pack does not use the attribute domain table.

6.1.9 Component attribute (table CompAttr)

CompAttr_ID INTEGER	Comp_ID INTEGER	AttrTyp_Cd CHAR(17)	CompAttr_Strt_DtTm TIMESTAMP	CompAttr_End_DtTm TIMESTAMP	CompAttr_Val VARCHAR(254)
1	0	AMX_EID	2003-02-17-18.12.55.889000	9999-01-01-00.00.00.000000	2
2	1	AMX_EID	2003-02-17-18.12.55.889000	9999-01-01-00.00.00.000000	1
3	0	AMX_GMT_OFFSET	2003-02-17-18.12.55.999000	9999-01-01-00.00.00.000000	-300
4	1	AMX_GMT_OFFSET	2003-02-17-18.12.55.999000	9999-01-01-00.00.00.000000	-300
5	0	LAST_IP_ADDRESS	2003-02-17-18.12.56.079000	9999-01-01-00.00.00.000000	107.33.474.228
6	1	LAST_IP_ADDRESS	2003-02-17-18.12.56.079000	9999-01-01-00.00.00.000000	107.33.474.227
7	5	CTD_APPL_NAME	2003-02-17-18.12.56.079000	9999-01-01-00.00.00.000000	RIM_DB2_AGENT
8	2	CTD_ISMPP	2002-06-01-02.00.00.000000	9999-01-01-00.00.00.000000	TRUE
9	11	NAME	2003-02-17-18.13.25.341000	9999-01-01-00.00.00.000000	Table1
10	11	OWNER	2003-02-17-18.13.25.572000	9999-01-01-00.00.00.000000	DB2

6.2 Component measurement

6.2.1 Measurement group type (table MgrpTyp)

MgrpTyp_Cd CHAR(6)	MgrpTyp_Nm * VARCHAR(120)
CATEG	Category
GROUP	Aggregate Types or Group Functions
STATE	State

6.2.2 Measurement group (table MGrp)

This table shows only the measurement groups that are valid after the installation of the DB2 warehouse enablement pack fixpack 1.

MGrp_Cd CHAR(6)	MgrpTyp_Cd CHAR(6)	MGrp_Parent_Cd CHAR(6)	MGrp_Nm * VARCHAR(120)
AVL	CATEG	NULL	Availability

CTDAGT	CATEG	UTIL	DB2 Agents
CTDAPP	CATEG	UTIL	DB2 Application Activity
CTDBP	CATEG	UTIL	DB2 Buffer Pool Activity
CTDBPE	CATEG	UTIL	DB2 BufferPool - External Storage
CTDCAP	CATEG	UTIL	DB2 Capture
CTDCC	CATEG	UTIL	DB2 Catalog Cache
CTDCFG	CATEG	UTIL	DB2 Configuration
CTDCNX	CATEG	UTIL	DB2 Connections
CTDDBC	CATEG	UTIL	DB2 Database Configuration
CTDDIO	CATEG	UTIL	DB2 Direct IO
CTDDLK	CATEG	UTIL	DB2 Locks/Deadlocks
CTDFCM	CATEG	UTIL	DB2 Fast Communication Manager
CTDHGW	CATEG	UTIL	DB2 Host Gateway
CTDHSB	CATEG	UTIL	DB2 Hash Joins
CTDHTP	CATEG	UTIL	DB2 Host Throughput
CTDLB	CATEG	UTIL	DB2 Load
CTDLKW	CATEG	UTIL	DB2 Lock Waits
CTDLOG	CATEG	UTIL	DB2 Logging
CTDPKC	CATEG	UTIL	DB2 Package Cache
CTDPBH	CATEG	UTIL	DB2 Pass Through
CTDREP	CATEG	UTIL	DB2 Replication

CTDSQL	CATEG	UTIL	DB2 SQL Activity
CTDSRT	CATEG	UTIL	DB2 Sorts
CTDSTO	CATEG	UTIL	DB2 Storage
CTDTBL	CATEG	UTIL	DB2 Table Activity
MEMORY	CATEG	NULL	Memory
NETWK	CATEG	NULL	Network
PERF	CATEG	NULL	Performance
STATE	CATEG	NULL	Percentage State Measurements
STORAG	CATEG	NULL	Storage
UTIL	CATEG	NULL	Utilization
AVG_E	GROUP	NULL	Average Value Exists
MAX_E	GROUP	NULL	Maximum Value Exists
MIN_E	GROUP	NULL	Minimum Value Exists
TOT_E	GROUP	NULL	Total Value Exists

6.2.3 Measurement group member (table MGrpMbr)

The columns MGrp_Nm and MsmtTyp_Nm are shaded because they do not exist in the TWG.MGrpMbr. They have been added in this documentation for clarity purposes only.

MGrp_Cd CHAR(6)	MGrp_Nm VARCHAR(120)	MgrpTyp_Cd CHAR(6)	MsmtTyp_Nm Varchar(254)
TOT_E	Total Value Exists	GROUP	No metrics
MIN_E	Minimum Value	GROUP	All metrics

	Exists		
MAX_E	Maximum Value Exists	GROUP	All metrics
AVG_E	Average Value Exists	GROUP	All metrics

CTDAGT	DB2 Agents	CATEG	AgentCreationRatio
			PctAgentsWaiting
			WorkLoad
CTDAPP	DB2 Application Activity	CATEG	PctConnectionsUsed
CTDBP	DB2 Buffer Pool Activity	CATEG	AvgAsyncReadsPerPoolRead
			AvgAsyncWritesPerPoolWrite
			AvgPoolIOTime
			AvgPoolReadTime
			AvgPoolWriteTime
			AvgPoolWritesPerPoolRead
			AvgSyncIOTime
			AvgSyncReadTime
			AvgSyncWriteTime
			InputOutputRate
			PctBufferPoolHits
			PctIndexHits
CTDBPE	DB2 BufferPool - External Storage	CATEG	PctExtStorageReadWrite
CTDCAP	DB2 Capture	CATEG	ApplyRequiresRefresh
			ApplySubscriptionLagTime
			CaptureLag
			FailedSubscriptions
			LaggingSubscriptions
CTDCC	DB2 Catalog Cache	CATEG	PctCatCacheHits
CTDCFG	DB2 Configuration	CATEG	ApplControlHeapSize
			ApplHeapAvailSize
			BufferPoolDefaultSize
			ChangedPagesThreshold
			InputOutputCleaners
			InputOutputServers
			LogBufferSize
			MaxLocklistStorage
			MaxPctLocklist
			MinCommits
			NumPrimaryLogs

			PackageCacheSize
			PctPrivateMemUsed
CTDCNX	DB2 Connections	CATEG	ConnWaitingForHostCurrent
			ConnectionErrors
			CurrentConnections
			MostRecentConnectResponseTime
			PctConnectionsExecuting
CTDDIO	DB2 Direct IO	CATEG	AvgDirectReadTime
			AvgDirectWriteTime
			DirectReadAvgSectorsPer
CTDDLK	DB2 Locks/Deadlocks	CATEG	AvgLockEscalationConn
			AvgLocksHeld
			PctDeadlockRollbacks
			PctIntDeadlockedRollbacks
			PctLockListUsed
CTDFCM	DB2 Fast Communication Manager	CATEG	PctFcmBufsInUse
			PctFcmConnInUse
			PctMaxFcmConnUsed
			PctMaxFcmMsgAnchorUsed
CTDHTP	DB2 Host Throughput	CATEG	HostThroughput
			HostTimePerStatement
			NetworkTimePerStatement
			TimePerStatement
CTDLKW	DB2 Lock Waits	CATEG	PctApplsInLockWait
CTDLOG	DB2 Logging	CATEG	LogPageReads
			LogPageWrites
			PctLogSpaceUsedByPrimary
			PctLogSpaceUsedBySecondary
			TotLogInputOutput
			TotSecondaryLogsInUse
CTDPKC	DB2 Package Cache	CATEG	PackageCacheInserts
			PackageCacheLookups
			PctPackageCacheHits
CTDSQL	DB2 SQL Activity	CATEG	CommitRate
			DDLSqlStatements

			FailedSqlStatements
			OpenBlockingCursors
			OpenCursors
			PctFailedSqlStatements
			PctSelectSqlStatements
			RollbackRate
			SqlSelectsExecuted
			SqlStatementsRate
			UIDSqlExecuted
CTDSRT	DB2 Sorts	CATEG	PctPipedSortHits
			PctPipedSortReqRejected
			PctSortOverflowed
			PipedSortReqRejected
CTDSTS	State Measurements for DB2 Resources	STATE	DB2Down
			DB2Up
			NumInactiveNodes
			TbspBackupPending
			TbspBackupProgress
			TbspDMSRebalance
			TbspDeletePending
			TbspDisablePending
			TbspDropPending
			TbspLoadPending
			TbspNormal
			TbspOffline
			TbspQuiescedEx
			TbspQuiescedSh
			TbspQuiescedUp
			TbspRFpending
			TbspRFprogress
			TbspReorgProgress
			TbspRestorePending
			TbspRestoreProgress
			TbspStorDef
			TbspStorDefChanged

			TbspStorDefFinal
			TbspStorageUndefined
			TbspTBScreeate
			TbspTBSdelete
CTDTBL	DB2 Table Activity	CATEG	RowsReadRate
			RowsWrittenRate

6.2.4 Measurement unit category (table MunitCat)

MunitCat_Cd CHAR(6)	MunitCat_Nm * VARCHAR(120)
TM	Time Duration
QTY	Quantity
PRC	Percentage
RT	Rate

6.2.5 Measurement unit (table Munit)

Munit_Cd CHAR(6)	MunitCat_Cd CHAR(6)	Munit_Nm * VARCHAR(120)
PRC	PRC	Percentage
Bps	RT	Bytes Per Second
MBps	RT	Megabytes Per Second
KBps	RT	Kilobytes Per Second
Rps	RT	Requests Per Second
Qps	RT	Quantity Per Second
Qpm	RT	Quantity Per Minute
QTY	QTY	Quantity
GB	QTY	Gigabytes
KB	QTY	Kilobytes
MB	QTY	Megabytes
B	QTY	Bytes
Msec	TM	Milliseconds
Sec	TM	Seconds
Min	TM	Minutes
Hr	TM	Hours
Day	TM	Days
HSc	TM	Hundredths of a Second
4KPgs	QTY	Pages (4 KB)

6.2.6 Time summary (table TmSum)

The period over which a measurement may be summarized.

TmSum_Cd CHAR	TmSum_Nm * VARCHAR(120)
H	Hourly
D	Daily
W	Weekly
M	Monthly
Q	Quarterly
Y	Yearly

6.2.7 Measurement source (table MSrc)

MSrc_Cd CHAR(6)	MSrc_Parent_Cd CHAR(6)	MSrc_Nm * VARCHAR(120)
Tivoli	NULL	Tivoli Application
AMX	Tivoli	IBM Tivoli Monitoring
CTD	AMX	IBM Tivoli Monitoring for Databases: DB2

6.2.8 Measurement type (table MsmtTyp)

The following table lists all of the measurement types that are used by the resource models for IBM Tivoli Monitoring for Databases: DB2. The resource models do not use all of the measurement types defined in the managed object format (MOF) file. You can write a customized resource model to collect data for the additional measurement types.

MsmTyp_ID INTEGER	Munit_Cd CHAR(6)	Msrc_Cd CHAR (6)	MsmTyp_Nm * VARCHAR(120)	MsmTyp_Ds * VARCHAR(254)
1	PRC	CTD	AgentCreationRatio	The percentage of the number of agents created from a pool to the number of agents assigned from a pool.
2	MSec	CTD	AppLockWaitTime	Total time (in seconds) that the application waits for a lock to be granted to it during the monitoring interval
3	4KPgs	CTD	ApplControlHeapSize	Maximum size (in 4-KB pages) for the application control heap in the database during the monitoring interval
4	QTY	CTD	ApplDeadlocks	Total number of deadlocks that occurred for the application during the monitoring interval
5	4KPgs	CTD	ApplHeapAvailSize	Size (in 4-KB pages) of the application heap that is available for each individual agent in the database during the monitoring interval
6	QTY	CTD	ApplyRequiresRefresh	Number of subscriptions the Apply program failed to replicate
7	Min	CTD	ApplySubscriptionLagTime	The amount of time for lagging subscriptions for the Apply program. A negative number indicates that the subscription is due in that number of minutes.
8	QTY	CTD	AvgAsyncReadsPerPoolRead	The ratio of buffer pool asynchronous data reads to the total number of pool reads.
9	QTY	CTD	AvgAsyncWritesPerPoolWrite	The ratio of buffer pool asynchronous writes (data and index) to the total number of pool writes.
10	MSec	CTD	AvgDirectReadTime	The average time (in milliseconds) used to perform a direct read.
11	MSec	CTD	AvgDirectWriteTime	The average time (in milliseconds) used to perform a direct write.
12	QTY	CTD	AvgLockEscalationConn	Average number of lock escalations per connection
13	QTY	CTD	AvgLocksHeld	The average number of locks held by each connected application.
14	MSec	CTD	AvgPoolIOTime	The average time (in milliseconds) needed to perform a buffer pool input or output operation.
15	MSec	CTD	AvgPoolReadTime	The average time (in milliseconds) for processing a database read request that caused data or index pages to be physically read from the disk to the buffer pool.
16	MSec	CTD	AvgPoolWriteTime	The average time (in milliseconds) to process a write request that caused data or index pages to be physically written from buffer pool to disk.
17	QTY	CTD	AvgPoolWritesPerPoolRead	The ratio of pool writes to pool reads.
18	MSec	CTD	AvgSyncIOTime	The average time (in milliseconds) needed to perform a synchronous input or output operation.
19	MSec	CTD	AvgSyncReadTime	Average time (in milliseconds) for synchronous data reads to be performed.

20	MSec	CTD	AvgSyncWriteTime	average time (in milliseconds) to perform synchronous data writes.
21	4KPGs	CTD	BufferPoolDefaultSize	The default number of buffer pool pages.
22	QTY	CTD	CaptureErrors	Number of errors encountered by the Capture program within the last five minutes
23	Min	CTD	CaptureLag	Time difference (in minutes) between the current timestamp and the last timestamp recorded by the Capture program. This time difference is the Capture lag
24	4KPGs	CTD	CatalogCacheSize	The number of catalog cache pages.
25	PRC	CTD	ChangedPagesThreshold	The threshold for changed pages.
26	Qps	CTD	CommitRate	The unit-of-work commit attempt rate per second. Unit-of-work commits include SQL COMMIT statements that are issued from applications as well as internal commits that are initiated by the database manager.
27	QTY	CTD	Commits	Total number of SQL COMMIT statements that were attempted by the application. This value includes SQL COMMIT statements that are issued from the application and internal commits that are initiated by the database manager for the application
28	QTY	CTD	ConnWaitingForHostCurrent	Number of connections to host database being handled by the DB2 Connect gateway that are waiting for reply from the host
29	QTY	CTD	ConnectionErrors	Number of times that a communication error occurred while a DCS application was attempting to connect to a host database, or while it was processing an SQL statement
30	QTY	CTD	CurrentConnections	The current number of connections.
31	PRC	CTD	DB2Down	Percentage of time the DB2 instance was down.
32	PRC	CTD	DB2Up	Percentage of time the DB2 instance was running.
33	QTY	CTD	DDLSqlStatements	The total number of executed SQL DDL statements.
34	QTY	CTD	DataLogicalReads	Number of requests made by the application to read data pages using the buffer pool
35	QTY	CTD	DeadlocksDelta	Number of deadlocks detected during the last monitoring interval
36	QTY	CTD	DirectReadAvgSectorsPer	The average number of sectors that are read in a direct read request.
37	QTY	CTD	DirectWriteAvgSectorsPer	Average number of sectors that are written by a direct write.
38	QTY	CTD	DynamicSqlStatements	The total number of attempted dynamic SQL statements.
39	QTY	CTD	FailedSqlStatements	The total number of failed attempted SQL statements.
40	QTY	CTD	FailedSubscriptions	Number of subscriptions the Apply program failed to replicate

41	QTY	CTD	FilesClosed	The total number of files that have been closed.
42	QTY	CTD	HostThroughput	Number of bytes sent plus the number of bytes received divided by the cumulative host response time
43	MSec	CTD	HostTimePerStatement	The cumulative host response time divided by the number of statements attempted.
44	QTY	CTD	InputOutputCleaners	Current value of the number of asynchronous page cleaners
45	Qps	CTD	InputOutputRate	The buffer pool input and output operation rate per second. This includes all physical data and index pages that go through the buffer pool when read or written.
46	QTY	CTD	InputOutputServers	Current value of the number of input and output servers
47	QTY	CTD	LaggingSubscriptions	Number of lagging subscriptions for the Apply program
48	QTY	CTD	LockEscalations	number of lock escalations during the monitoring interval. exclusive lock escalations are included in this number
49	QTY	CTD	LockEscalationsDelta	Number of lock escalations that occurred in the database during the monitoring interval. Exclusive lock escalations are included in this number
50	QTY	CTD	LockTimeoutsDelta	Number of times that a request to lock an object timed out instead of being granted during the monitoring interval
51	QTY	CTD	LockWaitsDelta	number of times that applications had to wait for locks during the monitoring interval
52	4KPgs	CTD	LogBufferSize	The number of log buffer pages.
53	QTY	CTD	LogPageReads	Number of log pages read from disk by the DB2 logging system
54	QTY	CTD	LogPageWrites	Number of log pages written to disk by the DB2 logging system
55	QTY	CTD	LogicalDataReads	The number of logical data page read requests that went through the buffer pool.
56	QTY	CTD	LogicalIndexReads	The number of logical index page read requests that went through the buffer pool.
57	4KPgs	CTD	MaxLocklistStorage	The maximum allowed number of lock list pages
58	PRC	CTD	MaxPctLocklist	The percentage of the lock list held by an application that must be filled before the database manager performs an escalation.
59	QTY	CTD	MinCommits	Value of the number of commits to group
60	MSec	CTD	MostRecentConnectResponseTime	Elapsed time between the start of connection processing and actual establishment of a connection for the most recent DCS application that connected to this database
61	MSec	CTD	NetworkTimePerStatement	The cumulative host response time, minus cumulative statement execution time, divided by the number of statements attempted.

62	QTY	CTD	NumInactiveNodes	The number of inactive nodes.
63	QTY	CTD	NumPrimaryLogs	Value of the number of primary log files
64	QTY	CTD	OpenBlockingCursors	total number of blocking cursors (local and remote) that are currently open
65	QTY	CTD	OpenCursors	total number of cursors (local and remote) that are currently open.
66	QTY	CTD	PackageCacheInserts	Number of times that a section had to be loaded into the package cache because an application found that the section it wanted was not in the package cache
67	QTY	CTD	PackageCacheLookups	Number of times that an application looked for a section in the package cache
68	4KPgs	CTD	PackageCacheSize	The number of package cache size pages.
69	PRC	CTD	PctAgentsWaiting	Percentage of registered agents that are waiting to execute a transaction. The value is defined as agents waiting for a token/maximum agents registered x 100
70	PRC	CTD	PctApplsInLockWait	The percentage of currently connected applications that are waiting on a lock.
71	QTY	CTD	PctBufferPoolHits	The percentage of successful index and data page buffer pool hits.
72	QTY	CTD	PctCatCacheHits	catalog cache hit ratio (as a percentage) during the monitoring interval
73	PRC	CTD	PctConnectionsExecuting	The percentage of the maximum number of applications allowed that are connected to a database and are processing a unit of work.
74	PRC	CTD	PctConnectionsUsed	Percentage of the allowed number of connections that are concurrent connections
75	PRC	CTD	PctDDLSqlStatements	The percentage of total SQL statements that were Data Definition Language (DDL) statements.
76	PRC	CTD	PctDeadlockRollbacks	Percentage of rollbacks that were due to deadlock during the monitoring interval
77	PRC	CTD	PctExtStorageReadWrite	Ratio (as a percentage) of data+index pages copied from extended storage to pages copied to extended storage during the monitoring interval
78	PRC	CTD	PctFailedSqlStatements	Percentage of total SQL statements that failed during the monitoring interval
79	PRC	CTD	PctFcmBufsInUse	Percentage of FCM buffers that are used within the partitioned database server during the monitoring interval
80	PRC	CTD	PctFcmConnInUse	Percentage of FCM connection entries that are used within the partitioned database server during the monitoring interval
81	PRC	CTD	PctIndexHits	Application index page hit ratio (as a percentage) for the buffer pool during the monitoring interval

82	PRC	CTD	PctIntDeadlockedRollbacks	Percentage of the total number of internal rollbacks due to deadlocks
83	PRC	CTD	PctLockListUsed	The percentage of the locklist used.
84	PRC	CTD	PctLogSpaceUsedByPrimary	Percentage of log space used by the primary log
85	PRC	CTD	PctLogSpaceUsedBySecondary	The percentage of log space used by the secondary log.
86	PRC	CTD	PctMaxFcmBufsUsed	Percentage of maximum FCM buffers used during processing within the partitioned database server
87	PRC	CTD	PctMaxFcmConnUsed	Maximum percentage of FCM connection entries used during processing within the partitioned database server
88	PRC	CTD	PctMaxFcmMsgAnchorUsed	Percentage of maximum FCM message anchors during processing within the partitioned database server
89	QTY	CTD	PctPackageCacheHits	Package cache hit ratio (as a percent) for the last monitoring interval. The package cache hit ratio is the ratio of: (package_cache_lookups - package_cache_inserts) / all_package_cache_lookups
90	PRC	CTD	PctPipedSortHits	Piped sort hit ratio (as a percentage) for last interval The piped sort hit ratio is the ratio of piped sorts accepted to piped sorts requested
91	QTY	CTD	PctPipedSortReqRejected	percentage of piped sort requests that were rejected during the monitoring interval
92	QTY	CTD	PctPrivateMemUsed	Percentage of private memory used
93	PRC	CTD	PctSelectSqlStatements	Percentage of total SQL statements that were SQL SELECT statements during the monitoring interval
94	PRC	CTD	PctSortHeapUsed	The percentage of the allocated sort heap used.
95	PRC	CTD	PctSortOverflowed	percentage of sorts that overflowed during the monitoring interval
96	PRC	CTD	PctSpaceUsedDMS	Percentage of space used in the Database Managed Space (DMS) tablespace
97	PRC	CTD	PctUidSqlStatements	Percentage of total SQL statements that were SQL UPDATE, INSERT, and DELETE statements executed during the monitoring interval
98	QTY	CTD	PhysicalIndexReads	The number of physical read requests used to get index pages into the buffer pool.
99	QTY	CTD	PhysicalIndexWrites	The number of times that a buffer pool index page was physically written to disk.
100	QTY	CTD	PipedSortReqRejected	Total number of piped sorts that were rejected during the monitoring interval
101	Qps	CTD	RollbackRate	Rate (counts per second) at which unit-of-work rollbacks were attempted during the monitoring interval.
102	QTY	CTD	RollbackSqlStatements	Total number of rollbacks explicitly attempted. This value

				includes only the number of SQL ROLLBACK statements issued from the application(s)
103	QTY	CTD	Rollbacks	total number of rollbacks. this value includes sql rollback statements that are issued from the application(s) and internal rollbacks that are initiated by the database manager for the application(s)
104	Qps	CTD	RowsReadRate	Rate (per second) at which rows were read from the table during the monitoring interval
105	Qps	CTD	RowsWrittenRate	Rate (per second) at which rows were changed (inserted, deleted, or updated) in the table during the monitoring interval
106	QTY	CTD	SectionInserts	Number of inserts of SQL sections by an application from its SQL work area
107	QTY	CTD	SectionLookups	Number of lookups of SQL sections by an application from its SQL work area
108	B	CTD	SpaceUsedSMS	Number of bytes allocated to the System Managed Space (SMS) tablespace
109	QTY	CTD	SqlReqSinceLastCommit	number of sql requests that were submitted since the last commit
110	QTY	CTD	SqlSelectsExecuted	Number of SQL SELECT statements that were executed.
111	Qps	CTD	SqlStatementsRate	Rate (statements per second) at which SQL statements were executed during the monitoring interval
112	QTY	CTD	StaticSqlStatements	Number of static SQL statements that were attempted by the application
113	PRC	CTD	TableDataReorgNeeded	Percentage of time the tables data needed to be reorganized.
114	PRC	CTD	TableDataReorgNotNeeded	Percentage of time the tables data did not need to be reorganized.
115	PRC	CTD	TableIndexReorgNeeded	Percentage of time at least one index needed to be reorganized.
116	PRC	CTD	TableIndexReorgNotNeeded	Percentage of time no index needed to be reorganized.
117	PRC	CTD	TbspBackupPending	Percentage of time tablespace was in Backup Pending state
118	PRC	CTD	TbspBackupProgress	Percentage of time tablespace was in Backup in Progress state
119	PRC	CTD	TbspDMSRebalance	Percentage of time tablespace was in DMS Rebalancer in progress state
120	PRC	CTD	TbspDeletePending	Percentage of time tablespace was in Delete Pending state
121	PRC	CTD	TbspDisablePending	Percentage of time tablespace was in Disable Pending state
122	PRC	CTD	TbspDropPending	Percentage of time tablespace was in Drop Pending

				state
123	PRC	CTD	TbspLoadPending	Percentage of time tablespace was in Load Pending state
124	PRC	CTD	TbspNormal	Percentage of time tablespace was in normal state.
125	PRC	CTD	TbspOffline	Percentage of time tablespace was in Offline state.
126	PRC	CTD	TbspQuiescedEx	Percentage of time tablespace was in quiesced:EXCLUSIVE state
127	PRC	CTD	TbspQuiescedSh	Percentage of time tablespace was in quiesced:SHARE state
128	PRC	CTD	TbspQuiescedUp	Percentage of time tablespace was in quiesced:UPDATE state
129	PRC	CTD	TbspRFpending	Percentage of time tablespace was in Roll Forward pending state
130	PRC	CTD	TbspRFprogress	Percentage of time tablespace was in Roll Forward in progress state
131	PRC	CTD	TbspReorgProgress	Percentage of time tablespace was in Reorg in Progress state
132	PRC	CTD	TbspRestorePending	Percentage of time tablespace was in Restore Pending state
133	PRC	CTD	TbspRestoreProgress	Percentage of time tablespace was in Restore in Progress state
134	PRC	CTD	TbspStorDef	Percentage of time tablespace was in Storage may be defined state
135	PRC	CTD	TbspStorDefChanged	Percentage of time tablespace was in StorDef was changed state
136	PRC	CTD	TbspStorDefFinal	Percentage of time tablespace was in StorDef in final state
137	PRC	CTD	TbspStorageUndefined	Percentage of time tablespace was in Storage must be Defined state.
138	PRC	CTD	TbspTBScreate	Percentage of time tablespace was in create in progress state.
139	PRC	CTD	TbspTBSdelete	Percentage of time tablespace was in delete in progress state.
140	MSec	CTD	TimePerStatement	Statement execution elapsed time/number of statements attempted
141	4KPGs	CTD	TotLogInputOutput	Total amount of log input and output. This amount is the sum of the number of log pages read and the number of log pages written within the monitoring interval
142	QTY	CTD	TotSecondaryLogsInUse	total number of secondary log files that are currently being used.
143	QTY	CTD	TotSorts	total number of sorts performed during the monitoring interval

144	Sec	CTD	TotSystemCpuTime	Total system CPU time used by this applications database manager agents. System CPU time represents the time, in seconds, spent in system calls.
145	Sec	CTD	TotUserCpuTime	total user cpu time. user cpu time represents time spent executing database manager code.
146	QTY	CTD	UIDSqlExecuted	number of sql update insert and delete statements that were executed.
147	QTY	CTD	WorkLoad	Ratio of the maximum number of subagents associated with this application to the number of agents that are stolen from the application by DB2 to work on a different application

6.2.9 Component measurement rule (table MsmtRul)

The column MsmtTyp_Nm is shaded because it does not exist in the TWG.MsmtRul. It replaces the MsmtTyp_id column in this documentation for clarity purposes only.

CompTyp_Cd CHAR(17)	MsmtTyp_Nm VARCHAR(120)
CTD_APPL	AppLockWaitTime
	ApplDeadlocks
	Commits
	DDLSqlStatements
	DirectReadAvgSectorsPer
	DirectWriteAvgSectorsPer
	FailedSqlStatements
	LockEscalations
	OpenBlockingCursors
	OpenCursors
	PctBufferPoolHits
	PctDDLSqlStatements
	PctIndexHits
	PctLockListUsed
	PctUidSqlStatements
	RollbackSqlStatements
	Rollbacks
	SectionInserts
	SectionLookups

	SqlReqSinceLastCommit
	SqlSelectsExecuted
	StaticSqlStatements
	TotSorts
	TotSystemCpuTime
	TotUserCpuTime
	UIDSqlExecuted
	WorkLoad
CTD_APPLY_ID	FailedSubscriptions
	LaggingSubscriptions
CTD_DATABASE	ApplControlHeapSize
	ApplHeapAvailSize
	AvgAsyncReadsPerPoolRead
	AvgAsyncWritesPerPoolWrite
	AvgDirectReadTime
	AvgDirectWriteTime
	AvgLockEscalationConn
	AvgLocksHeld
	AvgPoolIOTime
	AvgPoolReadTime
	AvgPoolWriteTime
	AvgPoolWritesPerPoolRead
	AvgSyncIOTime
	AvgSyncReadTime
	AvgSyncWriteTime
	BufferPoolDefaultSize
	CaptureErrors
	CaptureLag
	CatalogCacheSize
	ChangedPagesThreshold
	CommitRate
	DDLSqlStatements
	DataLogicalReads
	DeadlocksDelta
	DirectReadAvgSectorsPer
	DirectWriteAvgSectorsPer

	DynamicSqlStatements
	FailedSqlStatements
	FilesClosed
	InputOutputCleaners
	InputOutputRate
	InputOutputServers
	LockEscalationsDelta
	LockTimeoutsDelta
	LockWaitsDelta
	LogBufferSize
	LogPageReads
	LogPageWrites
	MaxLocklistStorage
	MaxPctLocklist
	MinCommits
	NumPrimaryLogs
	PackageCacheInserts
	PackageCacheLookups
	PackageCacheSize
	PctApplsInLockWait
	PctBufferPoolHits
	PctCatCacheHits
	PctConnectionsUsed
	PctDDLSqlStatements
	PctDeadlockRollbacks
	PctExtStorageReadWrite
	PctFailedSqlStatements
	PctIndexHits
	PctIntDeadlockedRollbacks
	PctLockListUsed
	PctLogSpaceUsedByPrimary
	PctLogSpaceUsedBySecondary
	PctPackageCacheHits
	PctSelectSqlStatements
	PctSortOverflowed

	RollbackRate
	SqlSelectsExecuted
	SqlStatementsRate
	TotLogInputOutput
	TotSecondaryLogsInUse
	TotSorts
	UIDSqlExecuted
CTD_GATEWAY	ConnWaitingForHostCurrent
	ConnectionErrors
	CurrentConnections
	HostThroughput
	HostTimePerStatement
	MostRecentConnectResponseTime
	NetworkTimePerStatement
CTD_INSTANCE	TimePerStatement
	AgentCreationRatio
	DB2Down
	DB2Up
	PctAgentsWaiting
	PctConnectionsExecuting
	PctConnectionsUsed
	PctPipedSortHits
	PctPipedSortReqRejected
	PctPrivateMemUsed
	PctSortHeapUsed
CTD_NODE	PipedSortReqRejected
	ApplControlHeapSize
	ApplHeapAvailSize
	AvgAsyncReadsPerPoolRead
	AvgAsyncWritesPerPoolWrite
	AvgDirectReadTime
	AvgDirectWriteTime
	AvgLockEscalationConn
	AvgLocksHeld
	AvgPoolIOTime
	AvgPoolReadTime

	AvgPoolWriteTime
	AvgPoolWritesPerPoolRead
	AvgSyncIOTime
	AvgSyncReadTime
	AvgSyncWriteTime
	BufferPoolDefaultSize
	CatalogCacheSize
	ChangedPagesThreshold
	CommitRate
	DDLSqlStatements
	DataLogicalReads
	DeadlocksDelta
	DirectReadAvgSectorsPer
	DirectWriteAvgSectorsPer
	DynamicSqlStatements
	FailedSqlStatements
	FilesClosed
	InputOutputCleaners
	InputOutputRate
	InputOutputServers
	LockEscalationsDelta
	LockTimeoutsDelta
	LockWaitsDelta
	LogBufferSize
	LogPageReads
	LogPageWrites
	MaxLocklistStorage
	MaxPctLocklist
	MinCommits
	NumPrimaryLogs
	PackageCacheInserts
	PackageCacheLookups
	PctApplsInLockWait
	PctBufferPoolHits
	PctCatCacheHits

	PctConnectionsUsed
	PctDDLSqlStatements
	PctDeadlockRollbacks
	PctExtStorageReadWrite
	PctFailedSqlStatements
	PctFcmBufsInUse
	PctFcmConnInUse
	PctIndexHits
	PctIntDeadlockedRollbacks
	PctLockListUsed
	PctLogSpaceUsedByPrimary
	PctLogSpaceUsedBySecondary
	PctMaxFcmConnUsed
	PctMaxFcmMsgAnchorUsed
	PctPackageCacheHits
	PctSelectSqlStatements
	PctSortOverflowed
	RollbackRate
	SqlSelectsExecuted
	SqlStatementsRate
	TotLogInputOutput
	TotSecondaryLogsInUse
	TotSorts
	UIDSqlExecuted
CTD_TABLE	RowsReadRate
	RowsWrittenRate
	TableDataReorgNeeded
	TableDataReorgNotNeeded
	TableIndexReorgNeeded
	TableIndexReorgNotNeeded
CTD_TABLE_APPLY	ApplyRequiresRefresh
	ApplySubscriptionLagTime
CTD_TBSP	AvgDirectReadTime
	AvgDirectWriteTime
	AvgPoolIOTime
	AvgPoolReadTime

	AvgPoolWriteTime
	AvgSyncIOTime
	AvgSyncReadTime
	AvgSyncWriteTime
	DataLogicalReads
	DirectReadAvgSectorsPer
	DirectWriteAvgSectorsPer
	FilesClosed
	InputOutputRate
	PctBufferPoolHits
	PctExtStorageReadWrite
	PctIndexHits
	PctSpaceUsedDMS
	SpaceUsedSMS
	TbspBackupPending
	TbspBackupProgress
	TbspDMSRebalance
	TbspDeletePending
	TbspDisablePending
	TbspDropPending
	TbspLoadPending
	TbspNormal
	TbspOffline
	TbspQuiescedEx
	TbspQuiescedSh
	TbspQuiescedUp
	TbspRFpending
	TbspRFprogress
	TbspReorgProgress
	TbspRestorePending
	TbspRestoreProgress
	TbspStorDef
	TbspStorDefChanged
	TbspStorDefFinal
	TbspStorageUndefined

	TbspTBSCreate
	TbspTBSdelete

6.2.10 Measurement (table Msmt)

Msmt_ID BIGINT	Comp_ID INTEGER	Msmt_Typ_ID INTEGER	TmSutm_Cd CHAR	Msmt_Strt_Dt DATE	Msmt_Strt_Tm TIME	Msmt_Min_Val FLOAT	Msmt_Max_Val FLOAT	Msmt_Avg_Val FLOAT	Msmt_Tot_Val FLOAT	Msmt_Smpl_Cnt FLOAT	Msmt_Err_Cnt FLOAT
1	2	69	H	20020601	02.00.00	+1.3000000000000E+001	+1.3000000000000E+001	+1.3000000000000E+001	NULL	NULL	NULL
2	2	69	H	20020601	03.00.00	+5.4000000000000E+001	+5.4000000000000E+001	+5.4000000000000E+001	NULL	NULL	NULL
3	2	1	H	20020601	02.00.00	+5.4000000000000E+001	+5.4000000000000E+001	+5.4000000000000E+001	NULL	NULL	NULL
4	2	1	H	20020601	03.00.00	+5.4000000000000E+001	+5.4000000000000E+001	+5.4000000000000E+001	NULL	NULL	NULL

6.3 Helper tables

This warehouse pack does not generate helper tables.

However the AMX Warehouse enablement pack on which the DB2 Warehouse Enablement pack depends uses helper tables.

6.4 Exception tables

If the DB2 resource model does not log data that correspond to the DB2 metadata inserted in the Central Datawarehouse database (ctd_cdw_data.sql), exceptions will appear in the following exception tables:

- ✓ AMX.stage_attrtyp_transl_ex: an exception occurs when:
 - Some attributes of a component type are not declared in the metadata table AMX.AttrTyp_Transl.
 - A component having a unique key (not a composite key) has this key declared as an attribute and it should not.
 - The IBM Tivoli Monitoring attribute name that is part of a composite key is not the same in the metadata tables AMX.CompTyp_Transl and the AMX.AttrTyp_Transl.

- ✓ AMX.stage_comptyp_transl_ex: an exception occurs when there is a component type in the TWG.RelnRul that do not exist in the AMX.CompTyp_Transl table.
- ✓ AMX.stage_inst_comp_pac_ex: an exception occurs when the IBM Tivoli Monitoring instance key in the AMX.stage_key_parsed contains at least one key that is not pre-defined in the metadata for that application.
- ✓ AMX.stage_categories_ex: an exception occurs when properties that are inside the categories table are not declared as attributes for that component type in the metadata table AMX.AttrTyp_Transl.
- ✓ AMX.stage_metrics_ex: an exception occurs when metric names in the METRICS table in the IBM Tivoli Monitoring database (ITM_DB) are not defined in the metadata table TWG.MsmtTyp.
- ✓ AMX.stage_relnrul_transl_ex: an exception occurs when in the AMX.RelRul_Transl table there is a source component type that does not correspond to a component type in the metadata table TWG.CompTyp table.
- ✓ AMX.stage_resource_transl_ex: an exception occurs when:
 - In the AMX.stage_resources table, there is no corresponding component type code in the table
 - The AMX.RelRul_Transl table contains a relation type different from PROXY, PCHILD, SAME
 - A component source type without a host for parent has a PROXY relationship.
 - AMX.stage_rmprofiles_ex: an exception occurs when there is in the source RMPROFILES table (in ITM_DB), a category that does not correspond to a category in the metadata table AMX.Category_Transl table.

6.5 Incremental extraction

This warehouse pack uses incremental extraction to extract data from the central data warehouse and store it into the data mart tables. The data in the TWG.Extract_Control table controls this process.

The following DB2 entries appear in the TWG.Extract_Control table.

EXTCTL_SOURCE	EXTCTL_TARGET	EXTCTL_FROM_INTSEQ	EXTCTL_TO_INTSEQ
CTD.STG_APPLY_MET	CTD.T_APPLY_METRIC	-1	-1
CTD.STG_APPL_MET	CTD.T_APPL_METRIC	-1	-1
CTD.STG_DB_MET	CTD.T_DB_METRIC	-1	-1
CTD.STG_GTWY_MET	CTD.T_GTWY_METRIC	-1	-1
CTD.STG_INST_MET	CTD.T_INST_METRIC	-1	-1
CTD.STG_NODE_MET	CTD.T_NODE_METRIC	-1	-1
CTD.STG_TABLE_MET	CTD.T_TABLE_METRIC	-1	-1
CTD.STG_TAPPLY_MET	CTD.T_TAPPLY_METRIC	-1	-1
CTD.STG_TBSP_ND_MET	CTD.T_TBSP_ND_METRIC	-1	-1
TWG.COMP	CTD.T_HOST	-1	-1
TWG.COMP	CTD.T_INST	-1	-1
TWG.COMP	CTD.T_DB	-1	-1

TWG.COMP	CTD.T_GTWY	-1	-1
TWG.COMP	CTD.T_NODE	-1	-1
TWG.COMP	CTD.T_APPLY	-1	-1
TWG.COMP	CTD.T_TABLE	-1	-1
TWG.COMP	CTD.T_APPL	-1	-1
TWG.COMP	CTD.T_TBSP_ND	-1	-1
TWG.COMP	CTD.T_TAPPLY	-1	-1
TWG.MSMT	CTD.STG_INST_HR	-1	-1
TWG.MSMT	CTD.STG_DB_HR	-1	-1
TWG.MSMT	CTD.STG_GTWY_HR	-1	-1
TWG.MSMT	CTD.STG_NODE_HR	-1	-1
TWG.MSMT	CTD.STG_APPLY_HR	-1	-1
TWG.MSMT	CTD.STG_TABLE_HR	-1	-1
TWG.MSMT	CTD.STG_APPL	-1	-1
TWG.MSMT	CTD.STG_TBSP_ND_HR	-1	-1
TWG.MSMT	CTD.STG_TAPPLY_HR	-1	-1

7 IBM Tivoli Monitoring integration

7.1 *Metadata tables for applications that use the generic IBM Tivoli Monitoring warehouse enablement pack*

This section defines the data in the Tivoli Enterprise Data Warehouse central data warehouse.

7.1.1 Resource translation (table Resource_Transl)

Resource	CompTyp_cd
DB2Application	CTD_APPL
DB2Apply	CTD_APPLY_ID
DB2Database	CTD_DATABASE
DB2Gateway	CTD_GATEWAY
DB2Instance	CTD_INSTANCE
DB2Partition	CTD_NODE
DB2Table	CTD_TABLE
DB2TableApply	CTD_TABLE_APPLY
DB2TableSpace	CTD_TBSP

7.1.2 Category translation (table Category_Transl)

Category	MSrc_cd
DB2Monitoring	CTD

7.1.3 Component type translation (table CompTyp_Transl)

MSrc_cd	ITM_Key_Property ¹	CompTyp_Cd	Comp_Format_Nm ¹
CTD	DB2Application.db2AppID	CTD_APPL	DB2Application.db2AppID
CTD	DB2Apply.db2ApplyID	CTD_APPLY_ID	DB2Apply.db2ApplyID
CTD	DB2Database.db2Database	CTD_DATABASE	DB2Database.db2Database
CTD	DB2Gateway.db2Database	CTD_GATEWAY	DB2Gateway.db2Database
CTD	DB2Instance.db2Instance	CTD_INSTANCE	DB2Instance.db2Instance
CTD	DB2Partition.db2Node	CTD_NODE	DB2Partition.db2Node
CTD	DB2Table.db2Schema;DB2Table.db2TableName	CTD_TABLE	DB2Table.db2Schema;DB2Table.db2TableName
CTD	DB2TableApply.db2Schema;DB2TableApply.db2TableName	CTD_TABLE_APPLY	DB2TableApply.db2Schema;DB2TableApply.db2TableName
CTD	DB2TableSpace.db2TbspName	CTD_TBSP	DB2TableSpace.db2TbspName

7.1.4 Attribute translation (table AttrTyp_Transl)

MSrc_cd	ITM_Attr_Property	AttrTyp_Cd
CTD	DB2Application.db2AppName	CTD_APPL_NAME
CTD	DB2Instance.isMPP	CTD_ISMPP
CTD	DB2Table.db2TableName	NAME
CTD	DB2Table.db2Schema	OWNER
CTD	DB2TableApply.db2TableName	NAME
CTD	DB2TableApply.db2Schema	OWNER

7.2 IBM Tivoli Monitoring resource models

The following sections contain information in the IBM Tivoli Monitoring database that is used as the operational data source for Tivoli Enterprise Data Warehouse. For more information about resource models, see the *IBM Tivoli Monitoring Workbench User's Guide*.

7.2.1 DB2 Agents Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	InstanceAgents
	Resourc VARCHAR(64)	DB2Instance
Instances ¹	Instance_Key VARCHAR(2096)	Db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AgentCreationRatio
		PctAgentsWaiting
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	ApplicationAgents
	Resourc VARCHAR(64)	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	db2AppID=id1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	WorkLoad
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.2 DB2 Apply Replication Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Apply Replication
	Resource	DB2Apply
Instances ¹	InstanceKey VARCHAR(2096)	db2ApplyID=2;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	FailedSubscriptions
		LaggingSubscriptions
Categories	Name VARCHAR(128)	

¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.

7.2.3 DB2 BufferPool Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database BufferPool
	Resource	DB2Database
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AvgAsyncReadsPerPoolRead
		AvgAsyncWritesPerPoolWrite
		AvgPoolIOTime
		AvgPoolReadTime
		AvgPoolWriteTime
		AvgPoolWritesPerPoolRead
		AvgSyncIOTime
		AvgSyncReadTime
		AvgSyncWriteTime
		PctBufferPoolHits
		PctIndexHits
Categories	Name VARCHAR(128)	

¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Application BufferPool
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	db2AppID=appl1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctIndexHits
		PctBufferPoolHits
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace BufferPool
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AvgPoolIOTime
		AvgPoolReadTime
		AvgPoolWriteTime
		InputOutputRate
		PctBufferPoolHits
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.4 DB2 BufferPool External Storage Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	DB_BufferPool_ExtStorage
	Resource	DB2Database
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctExtStorageReadWrite
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tbsp_BufferPool_ExtStorage
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctExtStorageReadWrite
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.5 DB2 Catalog Cache Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Catalog Cache
	Resource	DB2Database
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyDb1; DB2Instance.db2Instance=db2Inst1;

Metrics	Name VARCHAR(128)	PctCatCacheHits
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.6 DB2 CPU Utilization Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	CpuUtilization
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(209 6)	db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db 2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	TotUserCpuTime
		TotSystemCpuTime
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.7 DB2 Cursor Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	SQLCursors
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(209 6)	Db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db 2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	OpenCursors
		OpenBlockingCursors
Categories	Name VARCHAR(128)	

¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.

7.2.8 DB2 Instance Status Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Connections
	Resource	DB2Instance
Instances ¹	Instance_Key VARCHAR(2096)	db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctConnectionsExecuting
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Memory
	Resource	DB2Instance
Instances ¹	Instance_Key VARCHAR(2096)	db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctPrivateMemUsed
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Status
	Resource	DB2Instance
Instances ¹	Instance_Key VARCHAR(2096)	db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	DB2Down
		DB2Up
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.9 DB2 Database Status Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database Configuration
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	ApplControlHeapSize
		ApplHeapAvailSize
		BufferPoolDefaultSize
		CatalogCacheSize
		ChangedPagesThreshold
		InputOutputCleaners
		InputOutputServers
		LogBufferSize
		MaxLocklistStorage
		MaxPctLocklist
		MinCommits
		NumPrimaryLogs
		PackageCacheSize

Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace SMS
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	SpaceUsedSMS
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace DMS
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128))	PctSpaceUsedDMS
Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Application Activity
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctConnectionsUsed
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace Status
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	TbspBackupPending
		TbspBackupProgress
		TbspDMSRebalance
		TbspDeletePending
		TbspDisablePending
		TbspDropPending
		TbspLoadPending
		TbspNormal
		TbspOffline
		TbspQuiescedEx
		TbspQuiescedSh
		TbspQuiescedUp
		TbspRFpending
		TbspRFprogress
		TbspReorgProgress

		TbspRestorePending
		TbspRestoreProgress
		TbspStorDef
		TbspStorDefChanged
		TbspStorDefFinal
		TbspStorageUndefined
		TbspTBSCreate
		TbspTBSdelete
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Gateway Status
	Resource	DB2Gateway
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyGW1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	CurrentConnections
		MostRecentConnectResponseTime
		ConnectionErrors
		ConnWaitingForHostCurrent
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.10 DB2 Direct IO Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database Direct IO
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AvgDirectReadTime
		AvgDirectWriteTime
		DirectReadAvgSectorsPer
		DirectWriteAvgSectorsPer
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Application Direct IO
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	DirectReadAvgSectorsPer
		DirectWriteAvgSectorsPer
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Direct IO
	Resource	DB2TableSpace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AvgDirectReadTime
		AvgDirectWriteTime
		DirectReadAvgSectorsPer
		DirectWriteAvgSectorsPer
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.11 DB2 FCM Activity Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Fast Communication Manager
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctFcmBufsInUse
		PctFcmConnInUse
		PctMaxFcmBuffUsed
		PctMaxFcmConnUsed
		PctMaxFcmMsgAnchorUsed
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.12 Host Throughput Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Host Throughput
	Resource	DB2Gateway
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyGW1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	HostThroughput
		HostTimePerStatement
		NetworkTimePerStatement
		TimePerStatement
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.13 DB2 Locks- Deadlocks Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database Locks/Deadlocks
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AvgLockEscalationConn
		AvgLocksHeld
		DeadlockRate
		LockEscalationRate
		LockTimeoutRate
		PctIntDeadlockedRollbacks
		PctDeadlockRollbacks
		PctLockListUsed
Categories	Name VARCHAR(128)	

¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Application Locks/Deadlocks
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	Db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	DeadlockRate
		PctLockListUsed
		LockEscalations
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.14 DB2 Lock Waits Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	DatabaseLocks
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	LockWaitsDelta
		PctApplsInLockWait
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	ApplicationLocks
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	Db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	AppLockWaitTime
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.15 DB2 Logging Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Logging
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	LogPageReads
		LogPageWrites
		PctLogSpaceUsedByPrimary
		PctLogSpaceUsedBySecondary
		TotLogInputOutput
		TotSecondaryLogsInUse
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.16 DB2 Package Cache Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Package Cache
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PackageCacheInserts
		PackageCacheLookups
		PctPackageCacheHits
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.17 DB2 Replication Capture Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Replication
	Resource	DB2Database
Instances ¹	Instance_Key VARCHAR(2096)	db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	CaptureErrors
		CaptureLag
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.18 DB2 SAP Tablespace Usage Status Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace Status
	Resource	DB2Tablespace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;

Metrics	Name VARCHAR(128))	TableSpaceStatus
Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace SMS
	Resource	DB2Tablespace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	SpaceUsedSMS
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Tablespace DMS
	Resource	DB2Tablespace
Instances ¹	Instance_Key VARCHAR(2096)	db2TbspName=t1;DB2Partition.db2Node=node1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128))	PctSpaceUsedDMS
Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.19 DB2 Sorts Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database Sorts
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctSortOverflowed
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Instance Sorts
	Resource	DB2Instance
Instances ¹	Instance_Key VARCHAR(2096)	db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctPipedSortHits
		PctSortHeapUsed
		PctPipedSortRejected
		PipedSortReqRejected
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Application Sorts
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	Db2ApplID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;

Metrics	Name VARCHAR(128))	TotSorts
Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.20 DB2 SQL Statement Activity Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Database SQL Activity
	Resource	DB2Partition
Instances ¹	Instance_Key VARCHAR(2096)	db2Node=0;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128))	CommitRate
		DLLSqlStatements
		FailedSqlStatements
		PctFailedSqlStatements
		PctSelectSqlStatements
		RollbackRate
		SqlSelectsExecuted
		SqlStatementRate
		UIDSqlExecuted
Categories	Name VARCHAR(128))	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	SQL Activity
	Resource	DB2Application
Instances ¹	Instance_Key VARCHAR(2096)	Db2AppID=App1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	PctDDLSqlStatements
		PctUidSqlStatements
		Rollbacks
		RollbackSqlStatements
		SectionInserts
		SectionLookups
		StaticSqlAttempts
		SqlReqSinceLastCommit
		SqlSelectsExecuted
		UIDSqlExecuted
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.21 DB2 Table Apply Replication Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	Table Apply Replication
	Resource	DB2Apply
Instances ¹	Instance_Key VARCHAR(2096)	Db2ApplyID=1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	ApplyRequiresRefresh
		ApplySubscriptionLagTime
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

7.2.22 DB2 Table Activity Resource Model

ITM Table Name	Column Name	Value
Resources	Context VARCHAR(64)	SQL Activity
	Resource	DB2Table
Instances ¹	Instance_Key VARCHAR(2096)	db2Schema=XXX;db2TableName=tbl1;DB2Database.db2Database=MyDb1;DB2Instance.db2Instance=db2Inst1;
Metrics	Name VARCHAR(128)	RowsReadRate
		RowsWrittenRate
Categories	Name VARCHAR(128)	
¹ IBM Tivoli Monitoring uses a semicolon (;) to separate values within Instances and to terminate the final value.		

8 Data mart schema information

The following sections contain the definition of star schemas, metric dimension tables, data marts, and reports provided with the IBM Tivoli Monitoring for Databases, Version 5.1.1: DB2 warehouse pack.

Shaded columns in the following tables are translated by the application. Translated columns are also identified by an asterisk (*) after the name in the column heading.

8.1 Star schemas

Before using this section, read about the data mart schema for the Tivoli Enterprise Data Warehouse central data warehouse, which is described in *Enabling an Application for Tivoli Enterprise Data Warehouse*. That document defines the content of each table and explains the relationships between the tables in this document.

This warehouse pack provides the following star schemas.

8.1.1 CTD Instance Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Instance Hourly Data
Name of fact table	CTD.F_INST_HOUR
Name of metric dimension table	CTD.D_INST_METRIC
Names of other dimension tables	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.1.1 Fact table CTD.F INST HOUR

Inst_ID INTEGER	Inst_Vers_Id INTEGER	Host_ID INTEGER	Host_IP_AD INTEGER	Metric_ID INTEGER	Meas_hour TIMESTAMP	Min_value DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_count DOUBLE
			INTEGER							

8.1.2 CTD Instance Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Instance Daily Data
Name of fact table	CTD.F_INST_DAY
Name of metric dimension table	CTD.D_INST_METRIC
Names of other dimension tables	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.2.1 Fact table CTD.F INST DAY

Inst_ID INTEGER	Inst_Vers_Id INTEGER	Host_ID INTEGER	Host_IP_AD INTEGER	Metric_ID INTEGER	Meas_date TIMESTAMP	Min_value DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_count DOUBLE
--------------------	-------------------------	--------------------	---------------------------	----------------------	------------------------	---------------------	---------------------	---------------------	-----------------------	------------------------

8.1.3 CTD Instance Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Instance Weekly Data
Name of fact table	CTD.F_INST_WEEK
Name of metric dimension table	CTD.D_INST_METRIC
Names of other dimension tables	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.3.1 Fact table CTD.F INST WEEK

Inst_ID INTEGER	Inst_Vers_Id INTEGER	Host_ID INTEGER	Host_IP_AD INTEGER	Metric_ID INTEGER	Meas_date TIMESTAMP	Min_value DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_count DOUBLE
--------------------	-------------------------	--------------------	---------------------------	----------------------	------------------------	---------------------	---------------------	---------------------	-----------------------	------------------------

8.1.4 CTD Instance Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Instance Monthly Data
Name of fact table	CTD.F_INST_DAY
Name of metric dimension table	CTD.D_INST_METRIC
Names of other dimension tables	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.4.1 Fact table CTD.F INST MONTH

Inst_ID INTEGER	Inst_Ver s_Id INTEGER	Host_ID INTEGER	Host_IP_ AD INTEGER	Metric_ID INTEGER	Meas_da te TIMEST AMP	Min_valu e DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_co unt DOUBLE

8.1.5 CTD Database Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Hourly Data
Name of fact table	CTD.F_DB_HOUR
Name of metric dimension table	CTD.D_DB_METRIC
Names of other dimension tables	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.5.1 Fact table CTD.F DB HOUR

DB_ID INTEGER	Inst_ID INTEGER	Inst_Ve rs_ID INTEGER	Host_ID INTEGER	Host_IP_ _AD INTEGER	Metric_I D INTEGER	Meas_h our TIMEST AMP	Min_valu e DOUBLE	Max_valu e DOUBLE	Avg_valu e DOUBLE	Total_val ue DOUBLE	Sample_c ount DOUBLE
------------------	--------------------	-----------------------------	--------------------	----------------------------	--------------------------	--------------------------------	-------------------------	-------------------------	-------------------------	---------------------------	----------------------------

8.1.6 CTD Database Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Daily Data
Name of fact table	CTD.F_DB_DAY
Name of metric dimension table	CTD.D_DB_METRIC
Names of other dimension tables	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.6.1 Fact table CTD.F DB DAY

DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.7 CTD Database Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Weekly Data
Name of fact table	CTD.F_DB_WEEK
Name of metric dimension table	CTD.D_DB_METRIC
Names of other dimension tables	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.7.1 Fact table CTD.F DB WEEK

DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.8 CTD Database Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Monthly Data
Name of fact table	CTD.F_DB_MONTH
Name of metric dimension table	CTD.D_DB_METRIC
Names of other dimension tables	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.8.1 Fact table CTD.F DB MONTH

DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.9 CTD Database Partition Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Partition Hourly Data
Name of fact table	CTD.F_NODE_HOUR
Name of metric dimension table	CTD.D_NODE_METRIC
Names of other dimension tables	CTD.D_NODE
	CTD.D_ND_LGPTH
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.9.1 Fact table CTD.F NODE HOUR

Node_ID	Nd_IgPth_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.10 CTD Database Partition Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Partition Daily Data
Name of fact table	CTD.F_NODE_DAY
Name of metric dimension table	CTD.D_NODE_METRIC
Names of other dimension tables	CTD.D_NODE
	CTD.D_ND_LGPTH
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS

	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.10.1 Fact table CTD.F NODE DAY

Node_ID	Nd_IgPth_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Success
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.11 CTD Database Partition Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Partition Weekly Data
Name of fact table	CTD.F_NODE_WEEK
Name of metric dimension table	CTD.D_NODE_METRIC
Names of other dimension tables	CTD.D_NODE
	CTD.D_ND_LGPTH
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.11.1 Fact table CTD.F NODE WEEK

Node_ID	Nd_IgPth_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Success
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.12 CTD Database Partition Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Database Partition Monthly Data
Name of fact table	CTD.F_NODE_MONTH
Name of metric dimension table	CTD.D_NODE_METRIC
Names of other dimension tables	CTD.D_NODE
	CTD.D_ND_LGPTH
	CTD.D_DB
	CTD.D_INST

	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.12.1 Fact table CTD.F NODE MONTH

Node_ID	Nd_Ig Pth_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.13 CTD Gateway Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Gateway (DCS Connection) Hourly Data
Name of fact table	CTD.F_GTWAY_HOUR
Name of metric dimension table	CTD.D_GTWAY_METRIC
Names of other dimension tables	CTD.D_GTWAY
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.13.1 Fact table CTD.F GTWAY HOUR

Gtway_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.14 CTD Gateway Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Gateway (DCS Connection) Daily Data
Name of fact table	CTD.F_GTWAY_DAY
Name of metric dimension table	CTD.D_GTWAY_METRIC
Names of other dimension tables	CTD.D_GTWAY
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.14.1 Fact table CTD.F GTWY DAY

Gtwy_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.15 CTD Gateway Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Gateway (DCS Connection) Weekly Data
Name of fact table	CTD.F_GTWY_WEEK
Name of metric dimension table	CTD.D_GTWY_METRIC
Names of other dimension tables	CTD.D_GTWY
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.15.1 Fact table CTD.F GTWY WEEK

Gtwy_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.16 CTD Gateway Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Gateway (DCS Connection) Monthly Data
Name of fact table	CTD.F_GTWY_MONTH
Name of metric dimension table	CTD.D_GTWY_METRIC
Names of other dimension tables	CTD.D_GTWY
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.16.1 Fact table CTD.F GTWY MONTH

Gtwy_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.17 CTD Application Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Application Hourly Data
Name of fact table	CTD.F_APPL_HOUR
Name of metric dimension table	CTD.D_APPL_METRIC
Names of other dimension tables	CTD.D_APPL
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.17.1 Fact table CTD.F APPL HOUR

Appl_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.18 CTD Application Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Application Daily Data
Name of fact table	CTD.F_APPL_HOUR
Name of metric dimension table	CTD.D_APPL_METRIC
Names of other dimension tables	CTD.D_APPL
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.18.1 Fact table CTD.F APPL DAY

Appl_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.19 CTD Application Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Application Weekly Data
Name of fact table	CTD.F_APPL_HOUR
Name of metric dimension table	CTD.D_APPL_METRIC
Names of other dimension tables	CTD.D_APPL
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.19.1 Fact table CTD.F APPL WEEK

Appl_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.20 CTD Application Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Application Monthly Data
Name of fact table	CTD.F_APPL_HOUR
Name of metric dimension table	CTD.D_APPL_METRIC
Names of other dimension tables	CTD.D_APPL
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.20.1 Fact table CTD.F APPL MONTH

Appl_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.21 CTD Table Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table Hourly Data
Name of fact table	CTD.F_TABLE_HOUR
Name of metric dimension table	CTD.D_TABLE_METRIC
Names of other dimension tables	CTD.D_TABLE
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.21.1 Fact table CTD.F TABLE HOUR

Table_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.22 CTD Table Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table Daily Data
Name of fact table	CTD.F_TABLE_DAY
Name of metric dimension table	CTD.D_TABLE_METRIC
Names of other dimension tables	CTD.D_TABLE
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.22.1 Fact table CTD.F TABLE DAY

Table_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.23 CTD Table Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table Weekly Data
Name of fact table	CTD.F_TABLE_WEEK
Name of metric dimension table	CTD.D_TABLE_METRIC
Names of other dimension tables	CTD.D_TABLE
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.23.1 Fact table CTD.F TABLE WEEK

Table_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.24 CTD Table Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table Monthly Data
Name of fact table	CTD.F_TABLE_HOUR
Name of metric dimension table	CTD.D_TABLE_METRIC
Names of other dimension tables	CTD.D_TABLE
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.24.1 Fact table CTD.F TABLE MONTH

Table_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.25 CTD Tablespace Partition Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Tablespace Partition Hourly Data
Name of fact table	CTD.F_TBSP_ND_HOUR
Name of metric dimension table	CTD.D_TBSP_ND_METRIC
Names of other dimension tables	CTD.D_TBSP_ND
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.25.1 Fact table CTD.F TBSP ND HOUR

TBSP_ND_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.26 CTD Tablespace Partition Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Tablespace Partition Daily Data
Name of fact table	CTD.F_TBSP_ND_DAY
Name of metric dimension table	CTD.D_TBSP_ND_METRIC
Names of other dimension tables	CTD.D_TBSP_ND
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.26.1 Fact table CTD.F_TBSP_ND_DAY

TBSP_ND_ID INTEGER	DB_ID INTEGER	Inst_ID INTEGER	Inst_Vers_ID INTEGER	Host_ID INTEGER	Host_IP_AD INTEGER	Metric_ID INTEGER	Meas_date TIMESTAMP	Min_value DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_count DOUBLE
-----------------------	------------------	--------------------	-------------------------	--------------------	-----------------------	----------------------	------------------------	---------------------	---------------------	---------------------	-----------------------	------------------------

8.1.27 CTD Tablespace Partition Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Tablespace Partition Weekly Data
Name of fact table	CTD.F_TBSP_ND_WEEK
Name of metric dimension table	CTD.D_TBSP_ND_METRIC
Names of other dimension tables	CTD.D_TBSP_ND
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.27.1 Fact table CTD.F_TBSP_ND_WEEK

TBSP_ND_ID INTEGER	DB_ID INTEGER	Inst_ID INTEGER	Inst_Vers_ID INTEGER	Host_ID INTEGER	Host_IP_AD INTEGER	Metric_ID INTEGER	Meas_hour TIMESTAMP	Min_value DOUBLE	Max_value DOUBLE	Avg_value DOUBLE	Total_value DOUBLE	Sample_count DOUBLE
-----------------------	------------------	--------------------	-------------------------	--------------------	-----------------------	----------------------	------------------------	---------------------	---------------------	---------------------	-----------------------	------------------------

8.1.28 CTD Tablespace Partition Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Tablespace Partition Monthly Data
Name of fact table	CTD.F_TBSP_ND_MONTH
Name of metric dimension table	CTD.D_TBSP_ND_METRIC
Names of other dimension tables	CTD.D_TBSP_ND
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.28.1 Fact table CTD.F TBSP ND MONTH

TBSP _ND_ ID INTE GER	DB_I D INTE GER	Inst_ID INTE GER	Inst_V ers_ID INTE GER	Host_I D INTE GER	Host_I P_AD INTE GER	Metric _ID INTE GER	Meas_d ate TIMES TAMP	Min_va lue DOUB LE	Max_v alue DOUBL E	Avg_va lue DOUB LE	Total_v alue DOUBL E	Sample _count DOUBL E
-----------------------------------	--------------------------	------------------------	---------------------------------	----------------------------	-------------------------------	------------------------------	--------------------------------	-----------------------------	-----------------------------	-----------------------------	-------------------------------	--------------------------------

8.1.29 CTD Apply Program Star Hourly Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Apply Program Daily Data
Name of fact table	CTD.F_APPLY_HOUR
Name of metric dimension table	CTD.D_APPLY_METRIC
Names of other dimension tables	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.29.1 Fact table CTD.F APPLY HOUR

Apply_I D INTE GER	DB_ID INTE GER	Inst_ID INTE GER	Inst_Ve rs_ID INTE GER	Host_ID INTE GER	Host_IP _AD INTE GER	Metric_I D INTE GER	Meas_h our TIMES TAMP	Min_valu e DOUBLE	Max_valu e DOUBLE	Avg_val ue DOUBLE	Total_val ue DOUBLE	S c D
-----------------------------	----------------------	------------------------	---------------------------------	------------------------	-------------------------------	------------------------------	--------------------------------	-------------------------	-------------------------	-------------------------	---------------------------	-------------

8.1.30 CTD Apply Program Star Daily Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Apply Program Daily Data
Name of fact table	CTD.F_APPLY_DAY
Name of metric dimension table	CTD.D_APPLY_METRIC
Names of other dimension tables	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.30.1 Fact table CTD.F APPLY DAY

Apply_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Scaled
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.31 CTD Apply Program Star Weekly Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Apply Program Weekly Data
Name of fact table	CTD.F_APPLY_WEEK
Name of metric dimension table	CTD.D_APPLY_METRIC
Names of other dimension tables	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.31.1 Fact table CTD.F APPLY WEEK

Apply_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Scaled
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.32 CTD Apply Program Star Monthly Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Apply Program Monthly Data
Name of fact table	CTD.F_APPLY_MONTH
Name of metric dimension table	CTD.D_APPLY_METRIC
Names of other dimension tables	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.32.1 Fact table CTD.F APPLY MONTH

Apply_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_date	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.33 CTD Table (Apply Level) Hourly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table (Apply Level) Hourly Data
Name of fact table	CTD.F_TAPPLY_HOUR
Name of metric dimension table	CTD.D_TAPPLY_METRIC
Names of other dimension tables	CTD.D_TAPPLY
	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.33.1 Fact table CTD.F TAPPLY HOUR

Tappl_y_ID	Apply_ID	DB_ID	Inst_ID	Inst_Vers_ID	Host_ID	Host_IP_AD	Metric_ID	Meas_hour	Min_value	Max_value	Avg_value	Total_value	Sample_count
INTEGER	INTEGER	INTEGER	INTEGER	TIMESTAMP	INTEGER	INTEGER	INTEGER	TIMESTAMP	DOUBLE	DOUBLE	DOUBLE	DOUBLE	DOUBLE

8.1.34 CTD Table (Apply Level) Daily Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table (Apply Level) Daily Data
Name of fact table	CTD.F_TAPPLY_DAY
Name of metric dimension table	CTD.D_TAPPLY_METRIC
Names of other dimension tables	CTD.D_TAPPLY
	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.34.1 Fact table CTD.F TAPPLY DAY

Tappl y_ID	Apply _ID	DB_I D	Inst_I D	Inst_Ve rs_ID	Host_I P_AD	Host_I D	Metric _ID	Meas_ date	Min_va lue	Max_va lue	Avg_v alue	Total_ value	Sampl e_cou nt
INTE GER	INTE GER	INTE GER	INTE GER	TIMES TAMP	INTE GER	INTE GER	INTEG ER	TIMES TAMP	DOUB LE	DOUBL E	DOUBL E	DOUBL E	DOUBL E

8.1.35 CTD Table (Apply Level) Weekly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table (Apply Level) Weekly Data
Name of fact table	CTD.F_TAPPLY_WEEK
Name of metric dimension table	CTD.D_TAPPLY_METRIC
Names of other dimension tables	CTD.D_TAPPLY
	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.35.1 Fact table CTD.F TAPPLY WEEK

Tappl y_ID	Apply _ID	DB_I D	Inst_I D	Inst_Ve rs_ID	Host_I D	Host_I P_AD	Metric _ID	Meas_ date	Min_va lue	Max_va lue	Avg_v alue	Total_ value	Sampl e_cou nt
INTE GER	INTE GER	INTE GER	INTE GER	TIMES TAMP	INTE GER	INTEG ER	INTEG ER	TIMES TAMP	DOUB LE	DOUBL E	DOUBL E	DOUBL E	DOUBL E

8.1.36 CTD Table (Apply Level) Monthly Star Schema

The following table defines the star schema. The description of the star schema is translated.

Description of star schema (in IWH_STARSHEMA)	DB2 Table (Apply Level) Monthly Data
Name of fact table	CTD.F_TAPPLY_MONTH
Name of metric dimension table	CTD.D_TAPPLY_METRIC
Names of other dimension tables	CTD.D_TAPPLY
	CTD.D_APPLY
	CTD.D_DB
	CTD.D_INST
	CTD.D_INST_VERS
	CTD.D_HOST
	CTD.D_HOST_IP_AD

8.1.36.1 Fact table CTD.F TAPPLY MONTH

Tappl y_ID	Apply _ID	DB_I D	Inst_I D	Inst_Ve rs_ID	Host_I D	Metric _ID	Meas_ date	Min_va lue	Max_va lue	Avg_v alue	Total_ value	Sampl e_cou nt
INTE GER	INTE GER	INTE GER	INTE GER	TIMES TAMP	INTE GER	INTEG ER	TIMES TAMP	DOUB LE	DOUBL E	DOUBL E	DOUBL E	DOUBL E

8.2 Metric dimension tables

This section describes the metric dimension tables used by the star schemas in this warehouse pack. Shaded columns indicate text that is translated. These column headings are also marked with an asterisk.

8.2.1 CTD.D_INST_METRIC

metric_I D	met_nam e *	met_desc *	met_un its *	met_catego ry *	msrc_n m *	min_e xists	max_ex ists	ave_exi sts ir	total exist s
INTEGER	VARCHA R(254)	VARCHAR(254)	VARCH AR(254)	VARCHAR(254)	VARCH AR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHA R(1)

8.2.2 CTD.D_DB_METRIC

metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.3 CTD.D_NODE_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.4 CTD.D_GTWY_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.5 CTD.D_APPL_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.6 CTD.D_APPLY_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.7 CTD.D_TBSP_ND_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.8 CTD.D_TABLE_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.2.9 CTD.D_TAPPLY_METRIC

Metric_ID	met_name *	met_desc *	met_units *	met_category *	msrc_nm *	min_exists	max_exists	ave_exists	total_exists
INTEGER	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	VARCHAR(254)	CHAR(1)	CHAR(1)	CHAR(1)	CHAR(1)

8.3 Dimension tables

The following sections describe the dimension tables (other than metric dimension tables) used by the star schemas in this warehouse pack.

8.3.1 Dimension table CTD.D_HOST

The following columns are used in this dimension table:

- Host_id
- Host_name
- Cust_nm
- Centr_nm

8.3.2 Dimension table CTD.D_HOST_IP_AD

The following columns are used in this dimension table:

- Host_ip_ad_id
- Host_ip_ad

8.3.3 Dimension table CTD.D_INST

The following columns are used in this dimension table:

- Inst_id
- Inst_nm
- isMPP

8.3.4 Dimension table CTD.D_INST_VERS

The following columns are used in this dimension table:

- Inst_Vers_id
- Inst_Vers

8.3.5 Dimension table CTD.D_DB

The following columns are used in this dimension table:

- DB_id
- DB_Nm

8.3.6 Dimension table CTD.D_NODE

The following columns are used in this dimension table:

- Node_Id
- Node_Number
- Node_Port_Number
- Node_Hostname

8.3.7 Dimension table CTD.D_ND_LGPTH

The following columns are used in this dimension table:

- Nd_LGPTH_id
- Nd_LGPTH

8.3.8 Dimension table CTD.D_TABLE

The following columns are used in this dimension table:

- Table_id
- Table_owner
- Table_nm

8.3.9 Dimension table CTD.D_TBSP_ND

The following columns are used in this dimension table:

- Tbsp_nd_id
- Tbsp_nd_nm

8.3.10 Dimension table CTD.D_GTWY

The following columns are used in this dimension table:

- gtwy_id
- gtwy_nm

8.3.11 Dimension table CTD.D_APPL

The following columns are used in this dimension table:

- Appl_id
- Appl_nm

8.3.12 Dimension table CTD.D_APPLY

The following columns are used in this dimension table:

- apply_id
- applyid

8.3.13 Dimension table CTD.D_TAPPLY

The following columns are used in this dimension table:

- Tbapply_id
- Tapply_owner
- Tapply_Name

8.4 Data marts and reports

This warehouse pack provides the following data marts.

8.4.1 DB2 Data Mart

The DB2 Data Mart contains all the DB2 Star Schemas:

8.4.2 Reports

This data mart provides the following prepackaged reports.

8.4.2.1 DB2 Buffer Pool Activity at the Application Level (Daily Data) - Health Check

This health check report examines the trend of buffer pool activity in terms of buffer pool hit ratio and index hit ratio across for each DB2 applications across all DB2 databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value), meas_date from CTD.D_APPL, CTD.D_APPL_METRIC, CTD.D_DB, CTD.D_HOST,
CTD.D_HOST_IP_AD, CTD.D_INST, CTD.D_INST_VERS, CTD.F_APPL_DAY where
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_APPL_DAY.INST_VERS_ID AND CTD.D_HOST.HOST_ID =
CTD.F_APPL_DAY.HOST_ID AND CTD.D_DB.DB_ID = CTD.F_APPL_DAY.DB_ID AND
CTD.D_APPL_METRIC.METRIC_ID = CTD.F_APPL_DAY.METRIC_ID AND CTD.D_INST.INST_ID =
CTD.F_APPL_DAY.INST_ID AND CTD.D_APPL.APPL_ID = CTD.F_APPL_DAY.APPL_ID AND
CTD.D_HOST_IP_AD.HOST_IP_AD_ID = CTD.F_APPL_DAY.HOST_IP_AD_ID AND
CTD.D_APPL_METRIC.met_name = 'PctBufferPoolHits' AND CTD.F_APPL_DAY.meas_date >=
(timestamp(current date - 14 days, '00.00.00')) and CTD.F_APPL_DAY.meas_date < (timestamp(current date,
'00.00.00')) group by meas_date order by meas_date asc;
```

```
select avg(avg_value), meas_date from CTD.D_APPL, CTD.D_APPL_METRIC, CTD.D_DB, CTD.D_HOST,
CTD.D_HOST_IP_AD, CTD.D_INST, CTD.D_INST_VERS, CTD.F_APPL_DAY where
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_APPL_DAY.INST_VERS_ID AND CTD.D_HOST.HOST_ID =
CTD.F_APPL_DAY.HOST_ID AND CTD.D_DB.DB_ID = CTD.F_APPL_DAY.DB_ID AND
CTD.D_APPL_METRIC.METRIC_ID = CTD.F_APPL_DAY.METRIC_ID AND CTD.D_INST.INST_ID =
CTD.F_APPL_DAY.INST_ID AND CTD.D_APPL.APPL_ID = CTD.F_APPL_DAY.APPL_ID AND
CTD.D_HOST_IP_AD.HOST_IP_AD_ID = CTD.F_APPL_DAY.HOST_IP_AD_ID AND
CTD.D_APPL_METRIC.met_name = 'PctIndexHits' AND CTD.F_APPL_DAY.meas_date >= (timestamp(current
```


date - 14 days, '00.00.00')) and CTD.F_APPL_DAY.meas_date < (timestamp(current date, '00.00.00')) group by meas_date order by meas_date asc;

8.4.2.2 DB2 Buffer Pool Activity at the Database Level (Daily Data) - Health Check

This health check report examines the trend of buffer pool activity in terms of buffer pool hit ratio and index hit across per DB2 databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_DB_METRIC, CTD.D_HOST, CTD.D_HOST_IP_AD,
CTD.D_INST, CTD.D_INST_VERS, CTD.F_DB_DAY where CTD.D_HOST.HOST_ID =
CTD.F_DB_DAY.HOST_ID AND CTD.D_INST.INST_ID = CTD.F_DB_DAY.INST_ID AND
CTD.D_HOST_IP_AD.HOST_IP_AD_ID = CTD.F_DB_DAY.HOST_IP_AD_ID AND
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_DB_DAY.INST_VERS_ID AND CTD.D_DB_METRIC.METRIC_ID
= CTD.F_DB_DAY.METRIC_ID AND CTD.D_DB.DB_ID = CTD.F_DB_DAY.DB_ID AND
CTD.D_DB_METRIC.met_name = 'PctBufferPoolHits' AND CTD.F_DB_DAY.meas_date >= (timestamp(current
date - 14 days, '00.00.00')) and CTD.F_DB_DAY.meas_date < (timestamp(current date, '00.00.00')) group by
meas_date order by meas_date asc;
```

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_DB_METRIC, CTD.D_HOST, CTD.D_HOST_IP_AD,
CTD.D_INST, CTD.D_INST_VERS, CTD.F_DB_DAY where CTD.D_HOST.HOST_ID =
CTD.F_DB_DAY.HOST_ID AND CTD.D_INST.INST_ID = CTD.F_DB_DAY.INST_ID AND
CTD.D_HOST_IP_AD.HOST_IP_AD_ID = CTD.F_DB_DAY.HOST_IP_AD_ID AND
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_DB_DAY.INST_VERS_ID AND CTD.D_DB_METRIC.METRIC_ID
= CTD.F_DB_DAY.METRIC_ID AND CTD.D_DB.DB_ID = CTD.F_DB_DAY.DB_ID AND
CTD.D_DB_METRIC.met_name = 'PctIndexHits' AND CTD.F_DB_DAY.meas_date >= (timestamp(current date -
14 days, '00.00.00')) and CTD.F_DB_DAY.meas_date < (timestamp(current date, '00.00.00')) group by meas_date
order by meas_date asc;
```

8.4.2.3 DB2 Buffer Pool Activity at the Database Partition Level (Daily Data) - Health Check

This health check report examines the trend of buffer pool activity in terms of buffer pool hit ratio and index hit across for each DB2 database partition across all DB2 databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_HOST, CTD.D_HOST_IP_AD, CTD.D_INST,
CTD.D_INST_VERS, CTD.D_ND_LGPTH, CTD.D_NODE, CTD.D_NODE_METRIC, CTD.F_NODE_DAY where
CTD.D_NODE.NODE_ID = CTD.F_NODE_DAY.NODE_ID AND CTD.D_HOST_IP_AD.HOST_IP_AD_ID =
CTD.F_NODE_DAY.HOST_IP_AD_ID AND CTD.D_NODE_METRIC.METRIC_ID =
CTD.F_NODE_DAY.METRIC_ID AND CTD.D_INST.INST_ID = CTD.F_NODE_DAY.INST_ID AND
CTD.F_NODE_DAY.ND_LGPTH_ID = CTD.D_ND_LGPTH.ND_LGPTH_ID AND CTD.F_NODE_DAY.HOST_ID =
CTD.D_HOST.HOST_ID AND CTD.D_INST_VERS.INST_VERS_ID = CTD.F_NODE_DAY.INST_VERS_ID AND
CTD.D_DB.DB_ID = CTD.F_NODE_DAY.DB_ID AND CTD.D_NODE_METRIC.met_name = 'PctBufferPoolHits'
AND CTD.F_NODE_DAY.meas_date >= (timestamp(current date - 14 days, '00.00.00')) and
CTD.F_NODE_DAY.meas_date < (timestamp(current date, '00.00.00')) group by meas_date order by meas_date
asc;
```

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_HOST, CTD.D_HOST_IP_AD, CTD.D_INST,
CTD.D_INST_VERS, CTD.D_ND_LGPTH, CTD.D_NODE, CTD.D_NODE_METRIC, CTD.F_NODE_DAY where
CTD.D_NODE.NODE_ID = CTD.F_NODE_DAY.NODE_ID AND CTD.D_HOST_IP_AD.HOST_IP_AD_ID =
CTD.F_NODE_DAY.HOST_IP_AD_ID AND CTD.D_NODE_METRIC.METRIC_ID =
CTD.F_NODE_DAY.METRIC_ID AND CTD.D_INST.INST_ID = CTD.F_NODE_DAY.INST_ID AND
CTD.F_NODE_DAY.ND_LGPTH_ID = CTD.D_ND_LGPTH.ND_LGPTH_ID AND CTD.F_NODE_DAY.HOST_ID =
CTD.D_HOST.HOST_ID AND CTD.D_INST_VERS.INST_VERS_ID = CTD.F_NODE_DAY.INST_VERS_ID AND
CTD.D_DB.DB_ID = CTD.F_NODE_DAY.DB_ID AND CTD.D_NODE_METRIC.met_name = 'PctIndexHits' AND
```

```
CTD.F_NODE_DAY.meas_date >= (timestamp(current date - 14 days, '00.00.00')) and
CTD.F_NODE_DAY.meas_date < (timestamp(current date, '00.00.00')) group by meas_date order by meas_date
asc;
```

8.4.2.4 DB2 Buffer Pool Activity at the Tablespace Partition Level (Daily Data) - Health Check

This health check report examines the trend of buffer pool activity in terms of buffer pool hit ratio and index hit across for each DB2 tablespace partition across all DB2 databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_HOST, CTD.D_HOST_IP_AD, CTD.D_INST,
CTD.D_INST_VERS, CTD.D_ND_LGPTH, CTD.D_NODE, CTD.D_TBSP_ND, CTD.D_TBSP_ND_METRIC,
CTD.F_TBSP_ND_DAY where CTD.D_TBSP_ND.TBSP_ND_ID = CTD.F_TBSP_ND_DAY.TBSP_ND_ID AND
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_TBSP_ND_DAY.INST_VERS_ID AND CTD.D_INST.INST_ID =
CTD.F_TBSP_ND_DAY.INST_ID AND CTD.D_HOST_IP_AD.HOST_IP_AD_ID =
CTD.F_TBSP_ND_DAY.HOST_IP_AD_ID AND CTD.D_DB.DB_ID = CTD.F_TBSP_ND_DAY.DB_ID AND
CTD.D_HOST.HOST_ID = CTD.F_TBSP_ND_DAY.HOST_ID AND CTD.D_TBSP_ND_METRIC.METRIC_ID =
CTD.F_TBSP_ND_DAY.METRIC_ID AND CTD.D_ND_LGPTH.ND_LGPTH_ID =
CTD.F_TBSP_ND_DAY.ND_LGPTH_ID AND CTD.D_NODE.NODE_ID = CTD.F_TBSP_ND_DAY.NODE_ID
AND CTD.D_TBSP_ND_METRIC.met_name = 'PctBufferPoolHits' AND CTD.F_TBSP_ND_DAY.meas_date >=
(timestamp(current date - 14 days, '00.00.00')) and CTD.F_TBSP_ND_DAY.meas_date < (timestamp(current
date, '00.00.00')) group by meas_date order by meas_date asc;
```

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_HOST, CTD.D_HOST_IP_AD, CTD.D_INST,
CTD.D_INST_VERS, CTD.D_ND_LGPTH, CTD.D_NODE, CTD.D_TBSP_ND, CTD.D_TBSP_ND_METRIC,
CTD.F_TBSP_ND_DAY where CTD.D_TBSP_ND.TBSP_ND_ID = CTD.F_TBSP_ND_DAY.TBSP_ND_ID AND
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_TBSP_ND_DAY.INST_VERS_ID AND CTD.D_INST.INST_ID =
CTD.F_TBSP_ND_DAY.INST_ID AND CTD.D_HOST_IP_AD.HOST_IP_AD_ID =
CTD.F_TBSP_ND_DAY.HOST_IP_AD_ID AND CTD.D_DB.DB_ID = CTD.F_TBSP_ND_DAY.DB_ID AND
CTD.D_HOST.HOST_ID = CTD.F_TBSP_ND_DAY.HOST_ID AND CTD.D_TBSP_ND_METRIC.METRIC_ID =
CTD.F_TBSP_ND_DAY.METRIC_ID AND CTD.D_ND_LGPTH.ND_LGPTH_ID =
CTD.F_TBSP_ND_DAY.ND_LGPTH_ID AND CTD.D_NODE.NODE_ID = CTD.F_TBSP_ND_DAY.NODE_ID
AND CTD.D_TBSP_ND_METRIC.met_name = 'PctIndexHits' AND CTD.F_TBSP_ND_DAY.meas_date >=
(timestamp(current date - 14 days, '00.00.00')) and CTD.F_TBSP_ND_DAY.meas_date < (timestamp(current
date, '00.00.00')) group by meas_date order by meas_date asc;
```

8.4.2.5 DB2 Connections Used (Daily Data) - Summary

This summary report lists the average percentage of connections used for all database partitions for all DB2 Databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value) as avgofavg_value, CTD.D_NODE.NODE_HOSTNAME, CTD.D_NODE.NODE_NUMBER,
CTD.D_DB.DB_NM, CTD.D_INST.INST_NM, CTD.D_HOST.HOST_NAME, CTD.D_NODE_METRIC.met_name
from CTD.D_DB, CTD.D_HOST, CTD.D_HOST_IP_AD, CTD.D_INST, CTD.D_INST_VERS, CTD.D_ND_LGPTH,
CTD.D_NODE, CTD.D_NODE_METRIC, CTD.F_NODE_DAY where CTD.D_NODE.NODE_ID =
CTD.F_NODE_DAY.NODE_ID AND CTD.D_HOST.HOST_ID = CTD.F_NODE_DAY.HOST_ID AND
CTD.D_ND_LGPTH.ND_LGPTH_ID = CTD.F_NODE_DAY.ND_LGPTH_ID AND
CTD.D_NODE_METRIC.METRIC_ID = CTD.F_NODE_DAY.METRIC_ID AND CTD.D_INST.INST_ID =
CTD.F_NODE_DAY.INST_ID AND CTD.D_HOST_IP_AD.HOST_IP_AD_ID =
CTD.F_NODE_DAY.HOST_IP_AD_ID AND CTD.D_INST_VERS.INST_VERS_ID =
CTD.F_NODE_DAY.INST_VERS_ID AND CTD.D_DB.DB_ID = CTD.F_NODE_DAY.DB_ID AND
CTD.D_NODE_METRIC.met_name in ('PctConnectionsUsed') AND CTD.F_NODE_DAY.meas_date >=
(timestamp(current date - 14 days, '00.00.00')) and CTD.F_NODE_DAY.meas_date < (timestamp(current date,
'00.00.00')) group by cube(CTD.D_NODE.NODE_HOSTNAME, CTD.D_NODE.NODE_NUMBER,
```

CTD.D_DB.DB_NM, CTD.D_INST.INST_NM, CTD.D_HOST.HOST_NAME, CTD.D_NODE_METRIC.met_name)
 order by CTD.D_NODE.NODE_HOSTNAME asc, CTD.D_NODE.NODE_NUMBER asc, CTD.D_DB.DB_NM asc,
 CTD.D_INST.INST_NM asc, CTD.D_HOST.HOST_NAME asc, CTD.D_NODE_METRIC.met_name;

8.4.2.6 DB2 Database Primary Log Usage (Daily Data) - Health Check

This health check report examines the trends in terms of percentage of log space used by the primary log across all DB2 databases for the last 14 days.

The SQL Query is:

```
select avg(avg_value), meas_date from CTD.D_DB, CTD.D_DB_METRIC, CTD.D_HOST, CTD.D_HOST_IP_AD,
CTD.D_INST, CTD.D_INST_VERS, CTD.F_DB_DAY where CTD.D_HOST.HOST_ID =
CTD.F_DB_DAY.HOST_ID AND CTD.D_INST.INST_ID = CTD.F_DB_DAY.INST_ID AND
CTD.D_HOST_IP_AD.HOST_IP_AD_ID = CTD.F_DB_DAY.HOST_IP_AD_ID AND
CTD.D_INST_VERS.INST_VERS_ID = CTD.F_DB_DAY.INST_VERS_ID AND CTD.D_DB_METRIC.METRIC_ID
= CTD.F_DB_DAY.METRIC_ID AND CTD.D_DB.DB_ID = CTD.F_DB_DAY.DB_ID AND
CTD.D_DB_METRIC.met_name = 'PctLogSpaceUsedByPrimary' AND CTD.F_DB_DAY.meas_date >=
(timestamp(current date - 14 days, '00.00.00')) and CTD.F_DB_DAY.meas_date < (timestamp(current date,
'00.00.00')) group by meas_date order by meas_date asc;
```

8.5 How to enable new data in Tivoli Enterprise Data Warehouse when creating new resource models in IBM Tivoli Monitoring

Some metrics are declared in the CIM classes but are not used in any pre-packaged Tivoli resource models. If you are using these metrics in your own resource model, this is what you should do to enable them in Tivoli Enterprise Data Warehouse.

1. Go to the directory where Tivoli Enterprise Data Warehouse is installed, then go to ctd\v110\cdw\dml, and edit the file called ctd_cdw_data.sql. Find the metric that you are declaring in your new resource model and uncomment all the lines that relate to that metric.
2. Connect to the central data warehouse database and run this script (db2 -stvf ctd_cdw_data.sql). The metric should exist in the following tables TWG.MSMTTYP, TWG.MSMTRUL, TWG.MGRPMBR (MAX_E group and/or MIN_E group and/or AVG_E and/or TOT_E group).
3. Enable the new resource model for Tivoli Enterprise Data Warehouse and distribute it to the adequate endpoints.