IBM Tivoli Workload Scheduler, Version 8.3
Warehouse Enablement Pack, Version 8.3.0
Implementation Guide

for Tivoli Data Warehouse, Version 1.2

SC32-2259-00

Note:

Before using this information and the product it supports, read the information in Notices on page 77.

First Edition (April 2006)

This edition applies to IBM Tivoli Workload Scheduler, Version 8.3 (program number 5698-WSH) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

1 About this guide

This document describes the warehouse enablement pack, version 8.3.0 for IBM® Tivoli® Workload Scheduler, Version 8.3. This warehouse enablement pack is created for Tivoli Data Warehouse, Version 1.2.

It covers the following topics:

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

With this warehouse pack, you can enable Tivoli Workload Scheduler to save scheduling execution history data into the Tivoli Data Warehouse database in order to be retrieved and correlated with other enterprise level data to produce summaries and statistics about the scheduling activity in your enterprise.

1.1 Who should read this guide

This guide is for people who do any of the following activities:

- Plan for and install the warehouse pack
- Use and maintain the warehouse pack and its reports
- Create new reports
- Create additional warehouse packs that use data from this warehouse pack

Administrators and installers should have the following knowledge or experience:

- Basic system administration and file management of the operating systems on which the components of Tivoli Data Warehouse are installed
- An understanding of the basic concepts of relational database management
- Experience administering IBM DB2 Universal Database

Additionally, report designers and warehouse pack creators should have the following knowledge or experience:

- An understanding of the source data and application
- Data warehouse information and design, extract, transform, and load (ETL) processes, and online analytical processing (OLAP)

1.2 Publications

This section lists publications in the Tivoli Data Warehouse library and other related documents. It also describes how to access Tivoli publications online and how to order Tivoli publications.

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

- IBM Tivoli Workload Scheduler
- Tivoli Data Warehouse
- IBM DB2, DB2 Data Warehouse Center, and DB2 Warehouse Manager
- IBM Redbooks

1.2.1 IBM Tivoli Workload Scheduler library

The following documents are available in the Tivoli Data Warehouse library:

- *IBM Tivoli Workload Scheduling Suite Version 8.3 General Information, SC32-1256*

  Provides general information about all Tivoli Workload Scheduler products. It gives an overview of how they can be used together to provide workload management solutions for your whole enterprise.

- *IBM Tivoli Workload Scheduler Version 8.3 Job Scheduling Console Release Notes, SC32-1258*

  Provides late-breaking information about the job scheduling console.

- *IBM Tivoli Workload Scheduler Version 8.3 Job Scheduling Console User's Guide, SC32-1257*

  Describes how to work with Tivoli Workload Scheduler, regardless of operating system, using a common GUI called the job scheduling console.

- *IBM Tivoli Workload Scheduler Version 8.3 Release Notes, SC32-1277-03*

  Provides late-breaking information about Tivoli Workload Scheduler on operating systems other than z/OS.

- *IBM Tivoli Workload Scheduler Version 8.3 Planning and Installation Guide, SC32-1273*

  Describes how to plan for and install Tivoli Workload Scheduler on operating systems other than z/OS, and how to integrate Tivoli Workload Scheduler with NetView®, Tivoli Data Warehouse, and IBM Tivoli Business Systems Manager.

- *IBM Tivoli Workload Scheduler Version 8.3 Reference manual, SC32-1274*

  Describes the Tivoli Workload Scheduler command line used on operating systems other than z/OS, and how extended and network agents work.

- *IBM Tivoli Workload Scheduler Version 8.3 Administration and Troubleshooting, SC32-1275*

  Provides information about how to administer Tivoli Workload Scheduler on operating systems other than z/OS, and what to do if things go wrong. It includes help on many messages generated by the main components of Tivoli Workload Scheduler.

- *IBM Tivoli Workload Scheduler Version 8.3 Database Reference, SC32-1502*

  Describes a set of views created by running queries on a specific set of tables in the IBM Tivoli Workload Scheduler database and displaying the obtained information in tabular form.

- *IBM Tivoli Workload Scheduler Information Roadmap, GI11-6455*

  Provides a guide to where specific topics are described within the documentation set for distributed operating systems.

1.2.2 Tivoli Data Warehouse library

The following documents are available in the Tivoli Data Warehouse library. The library is available on the Tivoli Data Warehouse Documentation CD as well as online, as described in "Accessing publications online" on page 4.

- *Tivoli Data Warehouse Release Notes*, SC32-1399

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

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

  Describes how Tivoli Data Warehouse fits into your enterprise, explains how to plan for its deployment, and gives installation and configuration instructions. It contains maintenance procedures and troubleshooting information.

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

  Provides information about connecting an application to Tivoli Data Warehouse. This book is for application programmers who use Tivoli Data Warehouse to store and report on their application data, data warehousing experts who import Tivoli Data Warehouse data into business intelligence applications, and customers who put their local data in Tivoli Data Warehouse. This document is available only from the IBM Web site.

- *Tivoli Data Warehouse Messages*, SC09-7776

  Lists the messages generated by Tivoli Data Warehouse, and describes the corrective actions you should take.

1.2.3 Related publications

The following sections describe additional publications to help you understand and use Tivoli Data Warehouse.

1.2.3.1 IBM Redbooks

IBM Redbooks are developed and published by the IBM International Technical Support Organization, the ITSO. They explore integration, implementation, and operation of realistic customer scenarios. The following Redbooks contain information about Tivoli Data Warehouse:

- *Introduction to Tivoli Enterprise Data Warehouse*, SG24-6607

  Provides a broad understanding of Tivoli Data Warehouse. Some of the topics that are covered are concepts, architecture, writing your own extract, transform, and load processes (ETLs), and best practices in creating data marts.

- *Planning a Tivoli Enterprise Data Warehouse Project*, SG24-6608

  Describes the necessary planning you must complete before you can deploy Tivoli Data Warehouse. The guide shows how to apply these planning steps in a real-life deployment of a warehouse pack using IBM Tivoli Monitoring. It also contains frequently used Tivoli and DB2 commands and lists troubleshooting tips for Tivoli Data Warehouse.

1.2.3.2 IBM DB2, DB2 Data Warehouse Center, and DB2 Warehouse Manager library

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, which is available on the IBM Web site:

http://www-3.ibm.com/software/data/db2/library/

After you install DB2, its library is also available on your system.

The following DB2 documents are particularly relevant for people working with Tivoli 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, which is 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 DB2 Data Warehouse Center.

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

Provides information on how 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 of a platform-specific DB2 client. 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 DB2 Data Warehouse Center, and describes the actions you should take.

1.2.4 Accessing publications online

The publications CD or product CD contains the publications that are in the product library. The format of the publications is PDF, HTML, or both.

IBM posts publications for this and all other Tivoli products, as they become available and whenever they are updated, to the Tivoli Software Information Center Web site. The Tivoli Software Information Center is located at the following Web address:

http://publib.boulder.ibm.com/tividd/td/tdprodlist.html

Note: If you print PDF documents on other than letter-sized paper, select the **Fit to page** check box in the Adobe Acrobat Print dialog. This option is available when you click **File** → **Print**. **Fit to page** ensures that the full dimensions of a letter-sized page print on the paper that you are using.

1.2.5 Ordering publications

You can order many Tivoli publications online at the following Web site:

http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi

You can also order by telephone by calling one of these numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968
- In other countries, for a list of telephone numbers, see the following Web site:

  http://www.ibm.com/software/tivoli/order-lit/

1.3 Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. For the warehouse pack, you use the interfaces of IBM DB2 and the reporting tool. See those documentation sets for accessibility information.

1.4 Contacting software support

If you have a problem with a Tivoli product, refer to the following IBM Software Support Web site:

http://www.ibm.com/software/sysmgmt/products/support/

If you want to contact customer support, see the IBM Software Support Guide at the following Web site:

http://techsupport.services.ibm.com/guides/handbook.html

The guide provides information about how to contact IBM Software Support, depending on the severity of your problem, and the following information:

- Registration and eligibility
- Telephone numbers, depending on the country in which you are located
- Information you must have before contacting IBM Software Support

1.5 Participating in newsgroups

User groups provide software professionals with a forum for communicating ideas, technical expertise, and experiences related to the product. They are located on the Internet, and are available using standard news reader programs. These groups are primarily intended for user-to-user communication, and are not a replacement for formal support. You can use Web browsers like Netscape Navigator or Microsoft Internet Explorer to view these newsgroups:

Tivoli Data Warehouse

news://news.software.ibm.com/ibm.software.tivoli.enterprise-data-warehouse

1.6 Typeface conventions

This guide uses the following typeface conventions:

Bold

- Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as **Tip** and **Operating system considerations**)
- Column headings in a table
- Keywords and parameters in text

Italic

- Citations (titles of books, diskettes, and CDs)
- Words defined in text
- Emphasis of words (words as words)
- Letters as letters
- New terms in text (except in a definition list)
- Variables and values you must provide

`Monospace`

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

2 Overview

The following sections provide an overview of Tivoli Data Warehouse and the warehouse pack for IBM Tivoli Workload Scheduler.

2.1 Overview of Tivoli Data Warehouse

Tivoli 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 reports

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

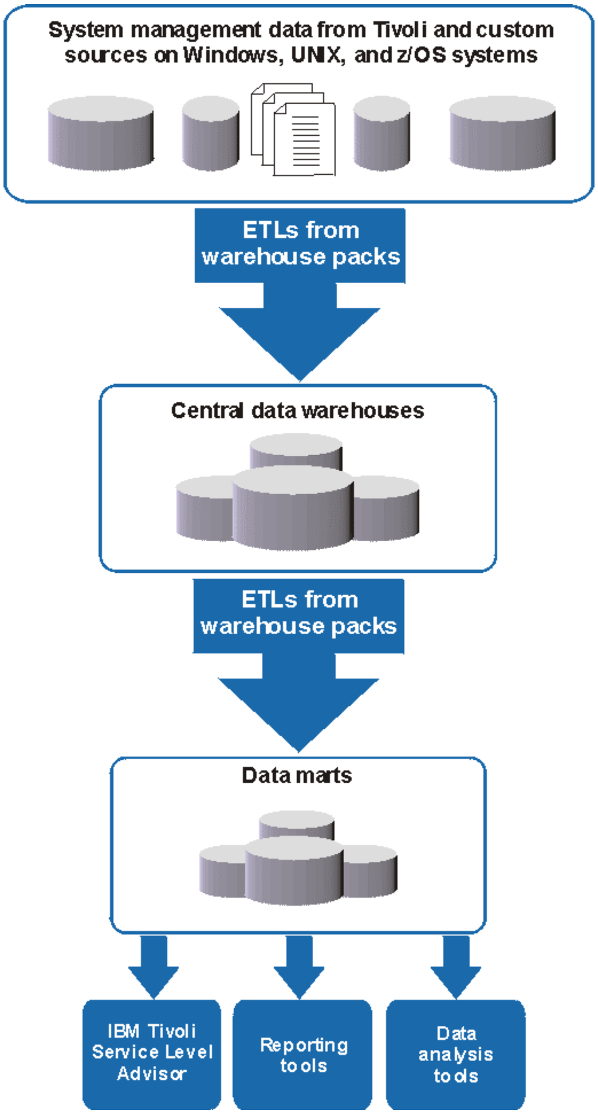

System management data from Tivoli and custom sources on Windows, UNIX, and z/OS systems
ETLs from warehouse packs
Central data warehouses
ETLs from warehouse packs
Data marts
IBM Tivoli Service Level Advisor
Reporting tools
Data analysis tools

Figure 1. Tivoli Data Warehouse basic architecture

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

A *data mart* is a subset of a data warehouse that contains data that is 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 simply *warehouse pack*.

The ETLs are typically scheduled to run periodically, usually during non-peak hours. If an ETL encounters data that it cannot correctly transform, it creates an entry in an exception table. Exception tables are described on page 68.

2.2 Overview of Tivoli Workload Scheduler Warehouse Pack

Tivoli Workload Scheduler helps in managing the workload on all the distributed and z/OS systems by planning and organizing every phase of workload production; you use it to plan which jobs to run, resolve interdependencies, launch and track each job.

The Tivoli Workload Scheduler master component creates a production plan containing the work to be done over a period of time and then distributes this plan to the Tivoli Workload Scheduler agents to run. The plan is updated with the status of all the jobs that run in the Tivoli Workload Scheduler environment. At the end of the production period, the plan is archived and a new plan is created and distributed to the Tivoli Workload Scheduler agents.

The Tivoli Workload Scheduler production plan is physically mapped into a binary file (named Symphony) that contains the scheduling activities to be performed in the period. When a new production plan is created, a new Symphony file is created, and all the activities that have not been completed in the previous plan are carried forward into the new Symphony file. The old Symphony file plan is archived in the schedlog Tivoli Workload Scheduler directory.

The archived Symphony files contain the history for all the jobs that have run during the previous production plans. The Tivoli Workload Scheduler **archiver** processes the archived Symphony files and fills the following flat files:

- Cpus
- Jobs
- Scheds

The **import** command then processes these flat files and imports the data into the following staging tables contained in the AWS schema:

- TWS_WORKSTATION_P
- TWS_JOBSTREAM_P
- TWS_JOB_P

In detail:

TWS_WORKSTATION_P

Column name	Data type	Length	Key	Description
Name	VARCHAR	96	*	The name of the workstation (CPU)

Column name	Data type	Length	Key	Description
Node	VARCHAR	255		The node of the CPU is installed
WorkstationType	INTEGER			The type of the CPU (1...6)
OperSys	VARCHAR	6		The system operating of the CPU (WNT, UNIX, OTHR)
DomainName	VARCHAR	96		The name of the domain of the CPU
MasterName	VARCHAR	96	*	The name of the Master of the CPU
HostName	VARCHAR	96		The name of the CPU if the type is XA
Sympohny_Date	TIMESTAMP		*	The data of the Symphony

TWS_JOBSTREAM_P

Column name	Data type	Length	Key	Description
Name	VARCHAR	96	*	The name of the job stream
Priority	INTEGER			The priority of the job stream
Status	INTEGER			The exit status of the job stream
Planned_StartTime	TIMESTAMP			The timestamp of the planned start
Actual_StartTime	TIMESTAMP		*	The timestamp of the start
Actual_Duration	INTEGER			The duration, in minutes, of the job stream
Cpu_Time	INTEGER			The time of the CPU, in sec, of the job stream
MasterName	VARCHAR	96	*	The name of the master
Jobstream_Id	VARCHAR	96	*	Job stream Internal Identifier
Jobstream_IA	TIMESTAMP		*	Input Arrival Time
Jobstream_CPU	VARCHAR	96	*	The name of the job stream CPU
DeadLine_Time	TIMESTAMP			The time within which a job or job stream is considered late in the plan
LateStart_Time	TIMESTAMP			The latest time a job or job stream is launched

TWS_JOB_P

Column name	**Data type**	**Length**	**Key**	**Description**
Name	VARCHAR	96	*	The name of the job
Priority	INTEGER			The priority of the job
Login	VARCHAR	48		The user login
Status	INTEGER			The exit status of the jobs
Planned_StartTime	TIMESTAMP			The timestamp of the planned start
Actual_StartTime	TIMESTAMP		*	The timestamp of the start
Actual_Duration	INTEGER			The duration, in min, of the jobs
Cpu_Time	INTEGER			The time of the CPU, in sec, of the job
WorkstationName	VARCHAR	96	*	The name of the job CPU
JobstreamName	VARCHAR	96	*	The name of the job stream
Jobstream_CPU	VARCHAR	96	*	The name of the job stream CPU
Command	VARCHAR	254		The command of the job
MasterName	VARCHAR	96	*	The name of the master
Jobstream_Id	VARCHAR	96	*	Job stream Internal Identifier
Jobstream_IA	TIMESTAMP		*	Input Arrival Time
DeadLine_Time	TIMESTAMP			The time within which a job or job stream is considered late in the plan
LateStart_Time	TIMESTAMP			The latest time a job or job stream is launched

The tws_launch_archiver perl script, provided with Tivoli Workload Scheduler, calls both the **archiver** process and the **import** command. This command generates a log file that will contain warning and error messages that occur during import, if the file already exists, the information is appended.

The tws_launch_archiver script can be scheduled on the Tivoli Workload Scheduler Master Domain Manager using Tivoli Workload Scheduler.

To run the tws_launch_archiver perl script use Perl level V5.8.0 or higher.

The tws_launch_archiver perl script can be run repeatedly, typically after a new Tivoli Workload Scheduler plan is created, and it can be scheduled to run using Tivoli Workload Scheduler itself. In this way, the Central Data Warehouse is updated with the new Tivoli Workload Scheduler data based on the jobs and job streams that have been run by Tivoli Workload Scheduler during the running of the previous Production Plan.

The picture below describes the typical Tivoli Workload Scheduler processing and the interaction with the Tivoli Data Warehouse.

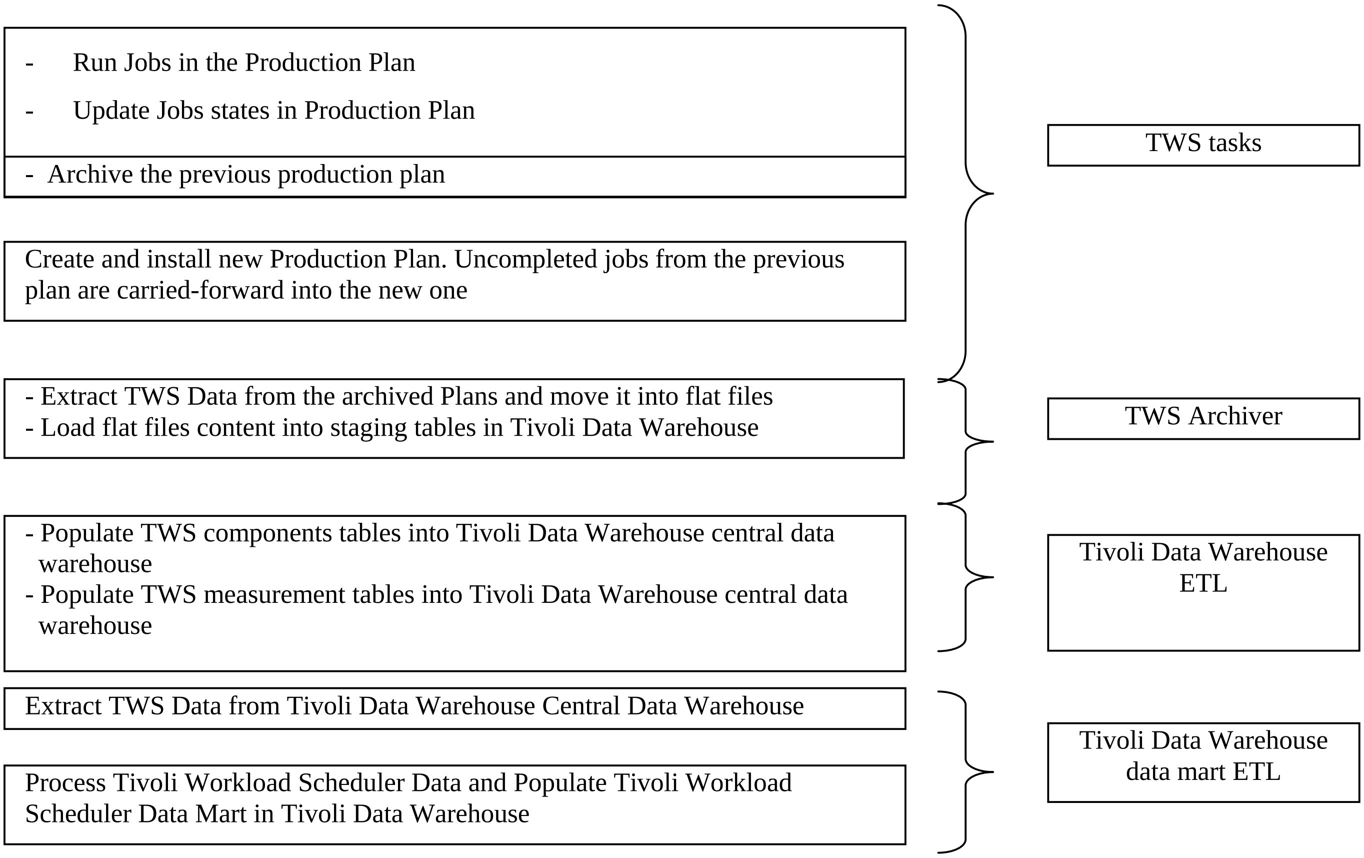

- Run Jobs in the Production Plan
- Update Jobs states in Production Plan
- Archive the previous production plan
Create and install new Production Plan. Uncompleted jobs from the previous plan are carried-forward into the new one
TWS tasks
- Extract TWS Data from the archived Plans and move it into flat files
- Load flat files content into staging tables in Tivoli Data Warehouse
TWS Archiver
- Populate TWS components tables into Tivoli Data Warehouse central data warehouse
- Populate TWS measurement tables into Tivoli Data Warehouse central data warehouse
Tivoli Data Warehouse ETL
Extract TWS Data from Tivoli Data Warehouse Central Data Warehouse
Process Tivoli Workload Scheduler Data and Populate Tivoli Workload Scheduler Data Mart in Tivoli Data Warehouse
Tivoli Data Warehouse data mart ETL

Figure 2. Tivoli Workload Scheduler processing

The Tivoli Workload Scheduler warehouse pack for Tivoli Data Warehouse is an implementation of the IBM DB2 Data Warehouse Center based on the data collected by Tivoli Workload Scheduler, version 8.3.0. As such, it provides a Data Warehouse Subject Area named AWS_Tivoli_Workload_Scheduler_v8.3.0_Subject_Area.

Within this Subject Area, there is the following process:

- AWS_c05_WorkloadScheduler_Process
- AWS_m05_WorkloadScheduler_Process

The Tivoli Workload Scheduler warehouse pack will extract the scheduling data only from the archived Production plans and not from the currently executing production plan.

3 Reports

This section provides information about the predefined reports provided by the warehouse pack.

The following information is provided:

- A list of the reports
- A description of the information contained in the reports
- The names of the data mart tables that are used to create the reports
- SQL queries for modifying a report or creating a new report based on this one

Report name	**Description**	**Table names**	**SQL queries**

Jobs with highest number of unsuccessful runs	This extreme case report shows jobs with the highest number of days when it have at least one or more unsuccessful runs (that is the job is abended or failed) during the time range selected.	AWS.D_RUNSTATES_METRIC
		AWS.F_RUNSTATES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNSTATES_DAY"."TOTAL_VALUE", "F_RUNSTATES_DAY"."MEAS_DATE", "F_RUNSTATES_DAY"."FACT_ID", "D_JOB"."JOB_NAME", "D_ENGINE"."ENGINE_NAME", "D_RUNSTATES_METRIC"."MET_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_RUNSTATES_METRIC" "D_RUNSTATES_METRIC" INNER JOIN "AWS"."F_RUNSTATES_DAY" "F_RUNSTATES_DAY" ON "D_RUNSTATES_METRIC"."METRIC_ID"="F_RUNSTATES_DAY"."METRIC_ID") INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNSTATES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID"
			WHERE "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unsuccessful Runs'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Workstations with highest number of unsuccessful runs	This extreme case report shows workstations with the highest number of unsuccessful jobs run on those workstations during the time range selected. This number is the sum of the days when the job is abended or failed.	AWS.D_RUNSTATES_METRIC
		AWS.F_RUNSTATES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "D_RUNSTATES_METRIC"."MET_NAME", "F_RUNSTATES_DAY"."TOTAL_VALUE", "F_RUNSTATES_DAY"."MEAS_DATE", "F_RUNSTATES_DAY"."FACT_ID", "D_WORKSTATION"."WORKSTATION_NAME", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_RUNSTATES_METRIC" "D_RUNSTATES_METRIC" INNER JOIN "AWS"."F_RUNSTATES_DAY" "F_RUNSTATES_DAY" ON "D_RUNSTATES_METRIC"."METRIC_ID"="F_RUNSTATES_DAY"."METRIC_ID") INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "F_RUNSTATES_DAY"."WORKSTATION_ID"="D_WORKSTATION"."WORKSTATION_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_WORKSTATION"."ENGINE_ID"="D_ENGINE"."ENGINE_ID"
			WHERE "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unsuccessful Runs'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Run states statistics for all jobs	This health check report compares the number of successful, unsuccessful, cancelled and unknown runs for all the jobs that were run	AWS.D_RUNSTATES_METRIC
		AWS.F_RUNSTATES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "D_RUNSTATES_METRIC"."MET_NAME", "F_RUNSTATES_DAY"."MEAS_DATE", "F_RUNSTATES_DAY"."FACT_ID", "F_RUNSTATES_DAY"."TOTAL_VALUE", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_RUNSTATES_METRIC" "D_RUNSTATES_METRIC" INNER JOIN "AWS"."F_RUNSTATES_DAY" "F_RUNSTATES_DAY" ON "D_RUNSTATES_METRIC"."METRIC_ID"="F_RUNSTATES_DAY

Report name	Description	Table names	SQL queries

	during the day.		"."METRIC_ID") INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNSTATES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID"
			WHERE ("D_RUNSTATES_METRIC"."MET_NAME"='Number of Cancelled Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Successful Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unknown Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unsuccessful Runs')
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Jobs with the highest average duration time	This extreme case report shows jobs with the highest average run duration time.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_JOB"."JOB_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_RUNTIMES_METRIC"."MET_NAME", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNTIMES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Duration'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Workstations with highest CPU utilization	This extreme case report shows workstations with the highest average CPU utilization.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_WORKSTATION"."WORKSTATION_NAME", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_WORKSTATION" "D_WORKSTATION" INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_WORKSTATION"."ENGINE_ID"="D_ENGINE"."ENGINE_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='CPU Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

CPU utilization time statistics for all jobs	This health check report shows the average CPU utilization time measurement for all the jobs that run during the day.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."FACT_ID", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_JOB" "D_JOB" INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_JOB"."JOB_ID"="F_RUNTIMES_DAY"."JOB_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON

Report name	Description	Table names	SQL queries

			"F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='CPU Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Duration time statistics for all jobs	This health check report shows the average duration time measurement for all the jobs that run during the day.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."FACT_ID", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."D_JOB" "D_JOB" INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_JOB"."JOB_ID"="F_RUNTIMES_DAY"."JOB_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Duration'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Start Delay statistics for all jobs	This health check report shows the average delay time measurement for all the jobs that run during the day.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM (("AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNTIMES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID") INNER JOIN "AWS"."D_ENGINE" "D_ENGINE" ON "D_JOB"."ENGINE_ID"="D_ENGINE"."ENGINE_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Start Delay'
			ORDER BY "D_ENGINE"."ENGINE_NAME"

Run states statistics for all workstations	This health check report compares the number of successful, unsuccessful, cancelled and unknown runs for all the jobs that were run in the Tivoli Workload Scheduler workstations.	AWS.D_RUNSTATES_METRIC
		AWS.F_RUNSTATES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "D_RUNSTATES_METRIC"."MET_NAME", "F_RUNSTATES_DAY"."MEAS_DATE", "F_RUNSTATES_DAY"."FACT_ID", "F_RUNSTATES_DAY"."TOTAL_VALUE", "D_ENGINE"."ENGINE_NAME", "D_ENGINE"."DOMAIN_NAME", "D_WORKSTATION"."WORKSTATION_NAME"
			FROM ("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN ("AWS"."D_RUNSTATES_METRIC" "D_RUNSTATES_METRIC" INNER JOIN "AWS"."F_RUNSTATES_DAY" "F_RUNSTATES_DAY" ON "D_RUNSTATES_METRIC"."METRIC_ID"="F_RUNSTATES_DAY"."METRIC_ID") ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNSTATES_DAY"."WORKSTATION_ID"
			WHERE ("D_RUNSTATES_METRIC"."MET_NAME"='Number of Cancelled Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Successful Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unknown Runs' OR "D_RUNSTATES_METRIC"."MET_NAME"='Number of Unsuccessful Runs')

Report name	Description	Table names	SQL queries
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

CPU utilization time statistics for all workstations	This health check report shows the average CPU utilization time measurement for all the jobs that run in the workstations.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."FACT_ID", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "D_ENGINE"."DOMAIN_NAME", "F_RUNTIMES_DAY"."JOB_ID"
			FROM ("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN ("AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID") ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='CPU Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Duration time statistics for all workstations	This health check report shows the average run duration time measurement for all the jobs that run in the workstations.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."FACT_ID", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "D_ENGINE"."DOMAIN_NAME", "F_RUNTIMES_DAY"."JOB_ID"
			FROM ("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN ("AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID") ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Duration'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Start Delay statistics for all workstations	This health check report shows the average delay time measurement for all the jobs that run in the workstations.	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_RUNTIMES_METRIC"."MET_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "D_ENGINE"."DOMAIN_NAME"
			FROM ("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN ("AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID") ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID"

Report name	Description	Table names	SQL queries

			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Start Delay'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Missed Deadline statistics for all jobs	This extreme case report shows the missed deadline statistics for all jobs	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_JOB
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_JOB"."JOB_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_RUNTIMES_METRIC"."MET_NAME", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "F_RUNTIMES_DAY"."MIN_VALUE", "F_RUNTIMES_DAY"."MAX_VALUE"
			FROM ((("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID") INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNTIMES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Deadline Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Missed Deadline statistics for all job streams	This extreme case report shows the missed deadline statistics for all job streams	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_JOBSTREAM
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "F_RUNTIMES_DAY"."AVG_VALUE", "D_RUNTIMES_METRIC"."MET_NAME", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "F_RUNTIMES_DAY"."MIN_VALUE", "F_RUNTIMES_DAY"."MAX_VALUE", "D_JOBSTREAM"."JOBSTREAM_NAME", "F_RUNTIMES_DAY"."JOB_ID"
			FROM ("AWS"."D_JOBSTREAM" "D_JOBSTREAM" INNER JOIN (("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID") ON "D_JOBSTREAM"."JOBSTREAM_ID"="F_RUNTIMES_DAY"."JOBSTREAM_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "F_RUNTIMES_DAY"."JOB_ID" IS NULL AND "D_RUNTIMES_METRIC"."MET_NAME"='Deadline Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Late Start statistics for all jobs	This extreme case report shows the late start statistics	AWS.D_RUNTIMES_METRIC AWS.F_RUNTIMES_DAY	SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "D_JOB"."JOB_NAME", "F_RUNTIMES_DAY"."AVG_VALUE", "D_RUNTIMES_METRIC"."MET_NAME",

Report name	Description	Table names	SQL queries

	for all jobs	AWS.D_WORKSTATION
		AWS.D_JOB
		AWS.D_ENGINE
			"D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "F_RUNTIMES_DAY"."MIN_VALUE", "F_RUNTIMES_DAY"."MAX_VALUE"
			FROM ((("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID") INNER JOIN "AWS"."D_JOB" "D_JOB" ON "F_RUNTIMES_DAY"."JOB_ID"="D_JOB"."JOB_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "D_RUNTIMES_METRIC"."MET_NAME"='Late Start Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

Late Start statistics for all job streams	This extreme case report shows the late start statistics for all job streams	AWS.D_RUNTIMES_METRIC
		AWS.F_RUNTIMES_DAY
		AWS.D_WORKSTATION
		AWS.D_JOBSTREAM
		AWS.D_ENGINE
			SELECT "F_RUNTIMES_DAY"."FACT_ID", "F_RUNTIMES_DAY"."MEAS_DATE", "F_RUNTIMES_DAY"."AVG_VALUE", "D_RUNTIMES_METRIC"."MET_NAME", "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME", "F_RUNTIMES_DAY"."MIN_VALUE", "F_RUNTIMES_DAY"."MAX_VALUE", "D_JOBSTREAM"."JOBSTREAM_NAME", "F_RUNTIMES_DAY"."JOB_ID"
			FROM ("AWS"."D_JOBSTREAM" "D_JOBSTREAM" INNER JOIN (("AWS"."D_ENGINE" "D_ENGINE" INNER JOIN "AWS"."D_WORKSTATION" "D_WORKSTATION" ON "D_ENGINE"."ENGINE_ID"="D_WORKSTATION"."ENGINE_ID") INNER JOIN "AWS"."F_RUNTIMES_DAY" "F_RUNTIMES_DAY" ON "D_WORKSTATION"."WORKSTATION_ID"="F_RUNTIMES_DAY"."WORKSTATION_ID") ON "D_JOBSTREAM"."JOBSTREAM_ID"="F_RUNTIMES_DAY"."JOBSTREAM_ID") INNER JOIN "AWS"."D_RUNTIMES_METRIC" "D_RUNTIMES_METRIC" ON "F_RUNTIMES_DAY"."METRIC_ID"="D_RUNTIMES_METRIC"."METRIC_ID"
			WHERE "F_RUNTIMES_DAY"."JOB_ID" IS NULL AND "D_RUNTIMES_METRIC"."MET_NAME"='Late Start Time'
			ORDER BY "D_ENGINE"."ENGINE_NAME", "D_WORKSTATION"."WORKSTATION_NAME"

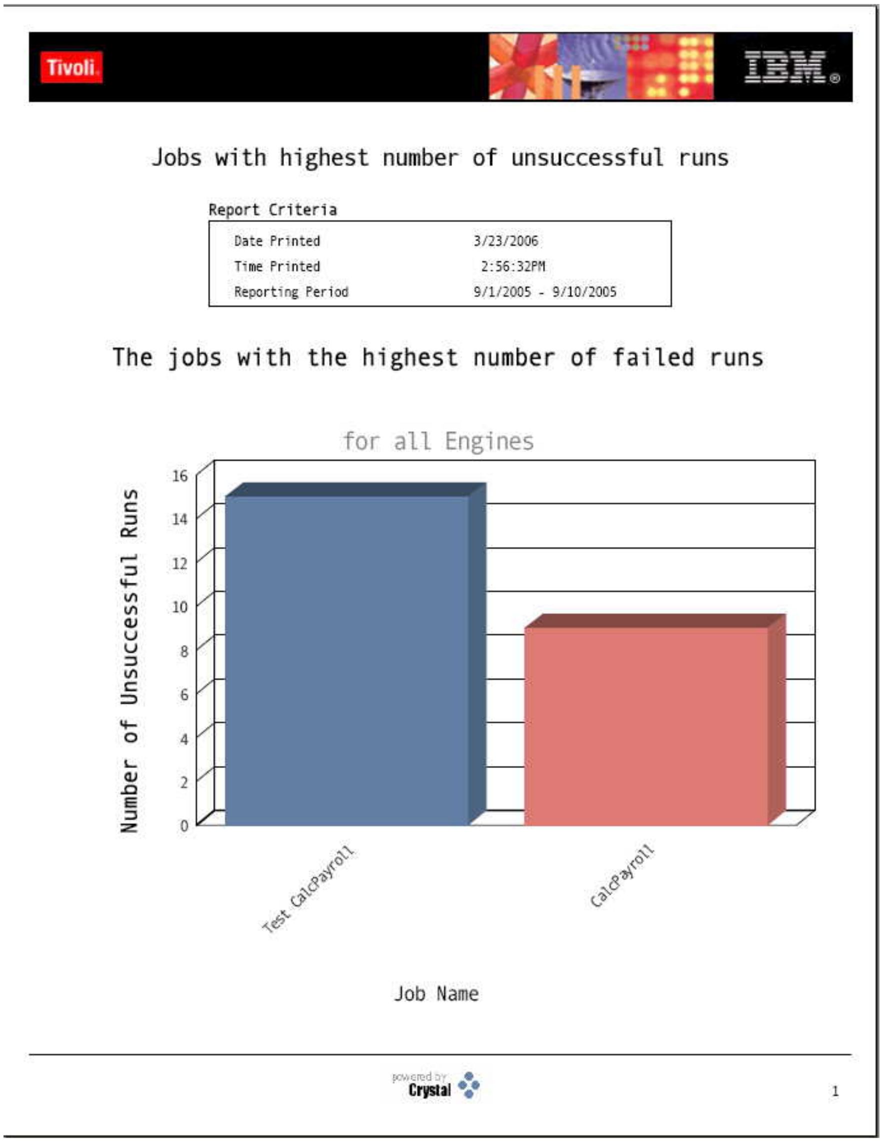
All reports are two input parameters:

- Start Date (Type: Date)
- End Date (Type: Date)

These parameters delimit the time range of the information that is shown in the reports. The following sections give more information about the reports.

3.1 Jobs with highest number of unsuccessful runs

This report shows a stacked bar chart with 3D visual effect. The first chart shows the top 10 jobs, for all Tivoli Workload Scheduler engines, with the highest number of unsuccessful runs (that is the job abended or failed). The other charts show the top 10 jobs grouped by Tivoli Workload Scheduler engine. Bars display the highest number of unsuccessful runs for each job during the time range selected.



Jobs with highest number of unsuccessful runs

Report Criteria

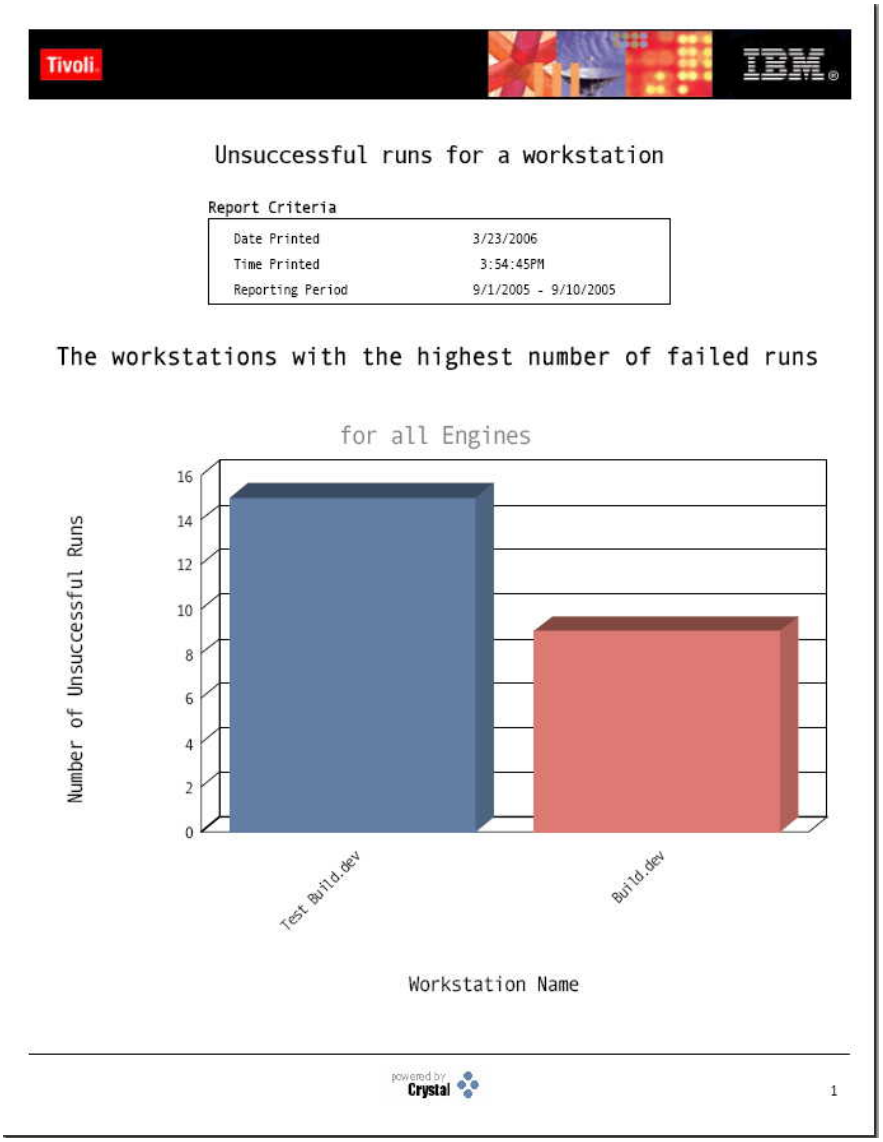
Date Printed	3/23/2006
Time Printed	2:56:32PM
Reporting Period	9/1/2005 - 9/10/2005

The jobs with the highest number of failed runs

Figure 3. Jobs with highest number of unsuccessful runs

3.2 Workstations with highest number of unsuccessful runs

This report shows a stacked bar chart with 3D visual effect. The first chart shows the top 10 workstations, for all Tivoli Workload Scheduler engines, with the highest number of unsuccessful run (that is the job abended or failed). The other charts show the top 10 workstations grouped by Tivoli Workload Scheduler engine. Bars display the highest number of unsuccessful runs on those workstations during the time range selected.



Unsuccessful runs for a workstation

Report Criteria

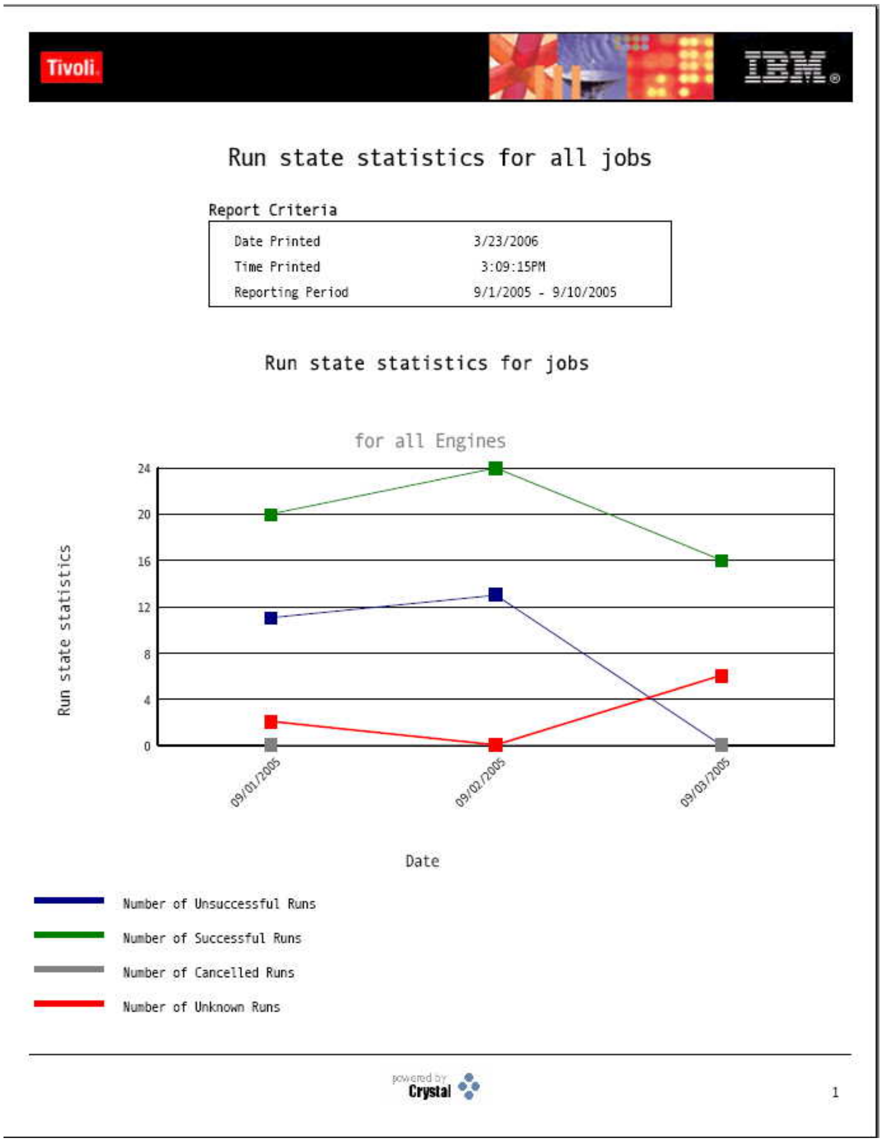
Date Printed	3/23/2006
Time Printed	3:54:45PM
Reporting Period	9/1/2005 - 9/10/2005

The workstations with the highest number of failed runs

Figure 4. Workstations with highest number of unsuccessful runs

3.3 Run states statistics for all jobs

This report shows a stacked line charts. The first chart shows the jobs statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the jobs grouped by Tivoli Workload Scheduler engine. The x-axis display the data (day-by-day) during the time range selected and the y-axis compares the number of successful, unsuccessful, cancelled and unknown runs for all the jobs that were run during the day.



Run state statistics for all jobs

Report Criteria

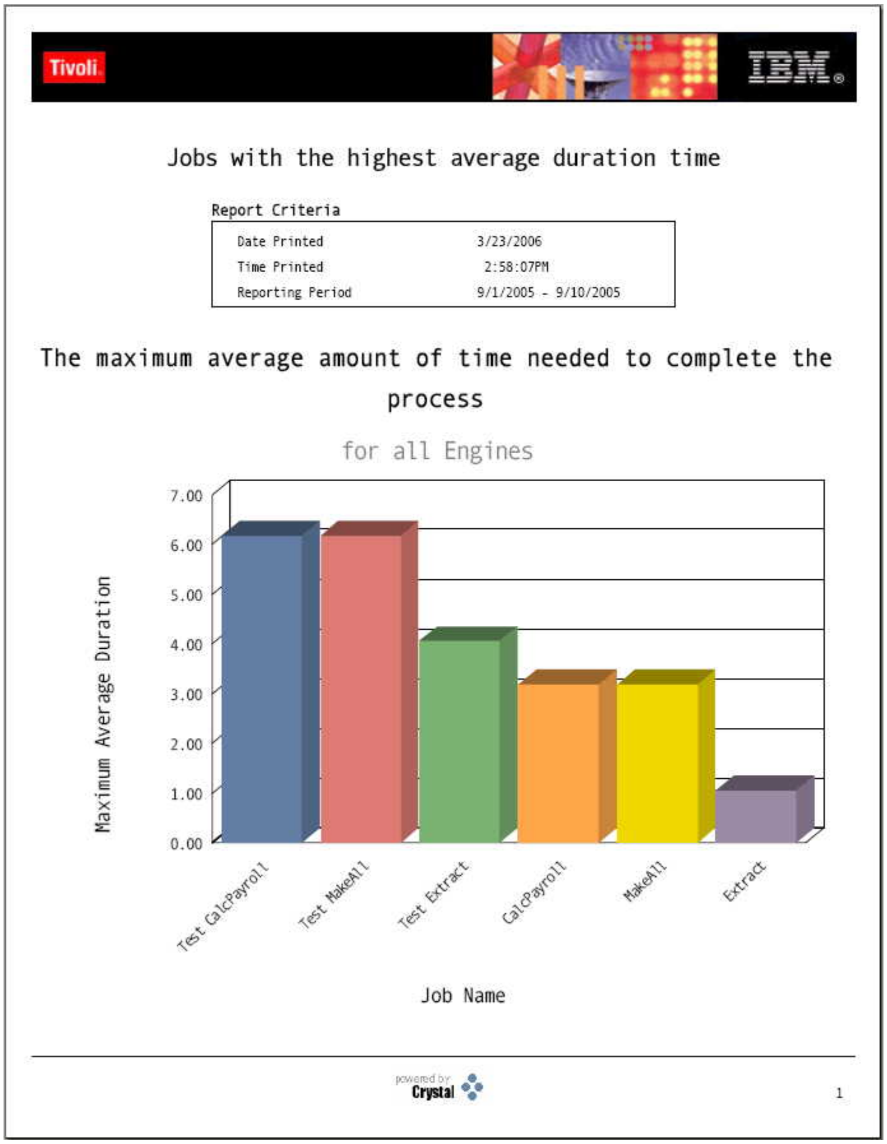
Date Printed	3/23/2006
Time Printed	3:09:15PM
Reporting Period	9/1/2005 - 9/10/2005

Run state statistics for jobs

Figure 5. Run states statistics for all job

3.4 Jobs with the highest average duration time

This report shows a stacked bar chart with 3D visual effect. The first chart shows the top 10 jobs, for all Tivoli Workload Scheduler engines, with the highest average duration time. The other charts show the top 10 jobs grouped by Tivoli Workload Scheduler engine. Bars display the highest average duration time for each job during the time range selected.



Jobs with the highest average duration time

Report Criteria

Date Printed	3/23/2006
Time Printed	2:58:07PM
Reporting Period	9/1/2005 - 9/10/2005

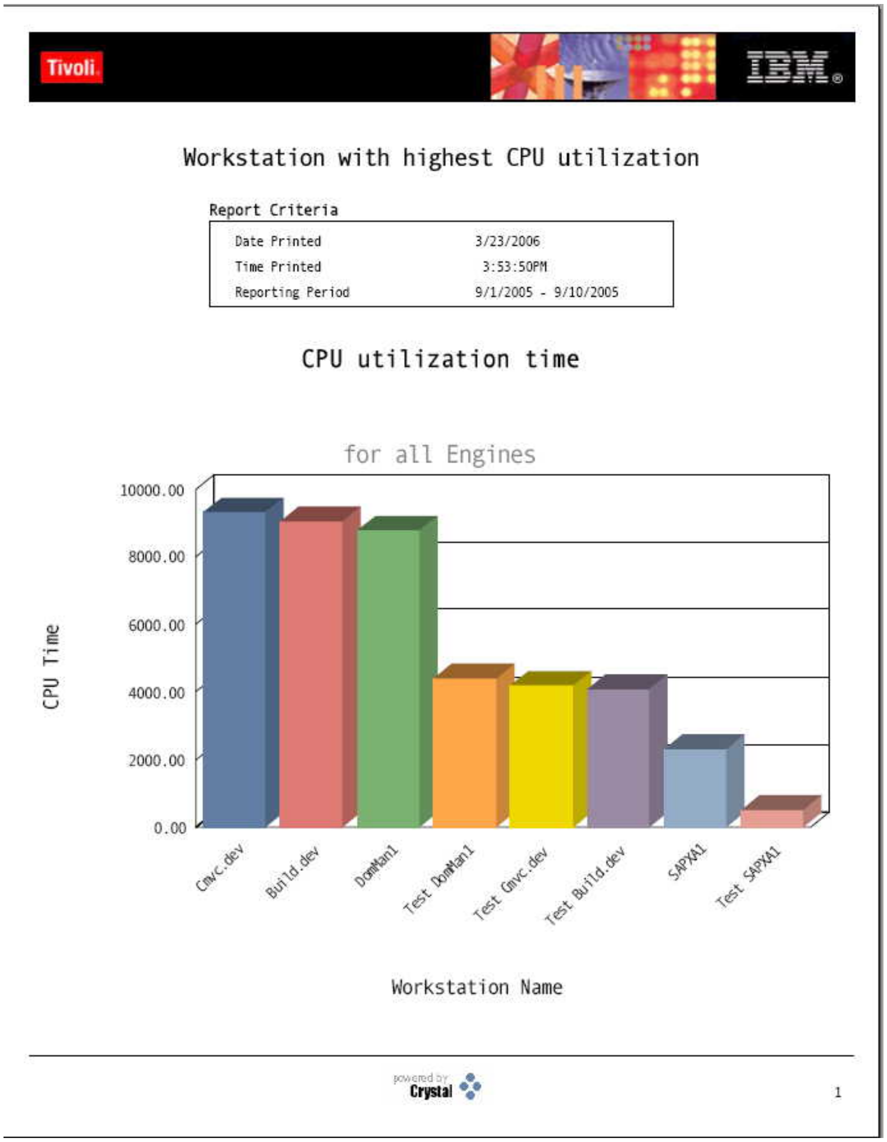
The maximum average amount of time needed to complete the process

for all Engines

Figure 6. Jobs with the highest average duration time

3.5 Workstations with highest CPU utilization

This report shows a stacked bar chart with 3D visual effect. The first chart shows the top 10 workstations, for all Tivoli Workload Scheduler engines, with the highest CPU utilization. The other charts show the top 10 workstations grouped by Tivoli Workload Scheduler engine. Bars display the highest CPU utilization of runs on those workstations during the time range selected.



Workstation with highest CPU utilization

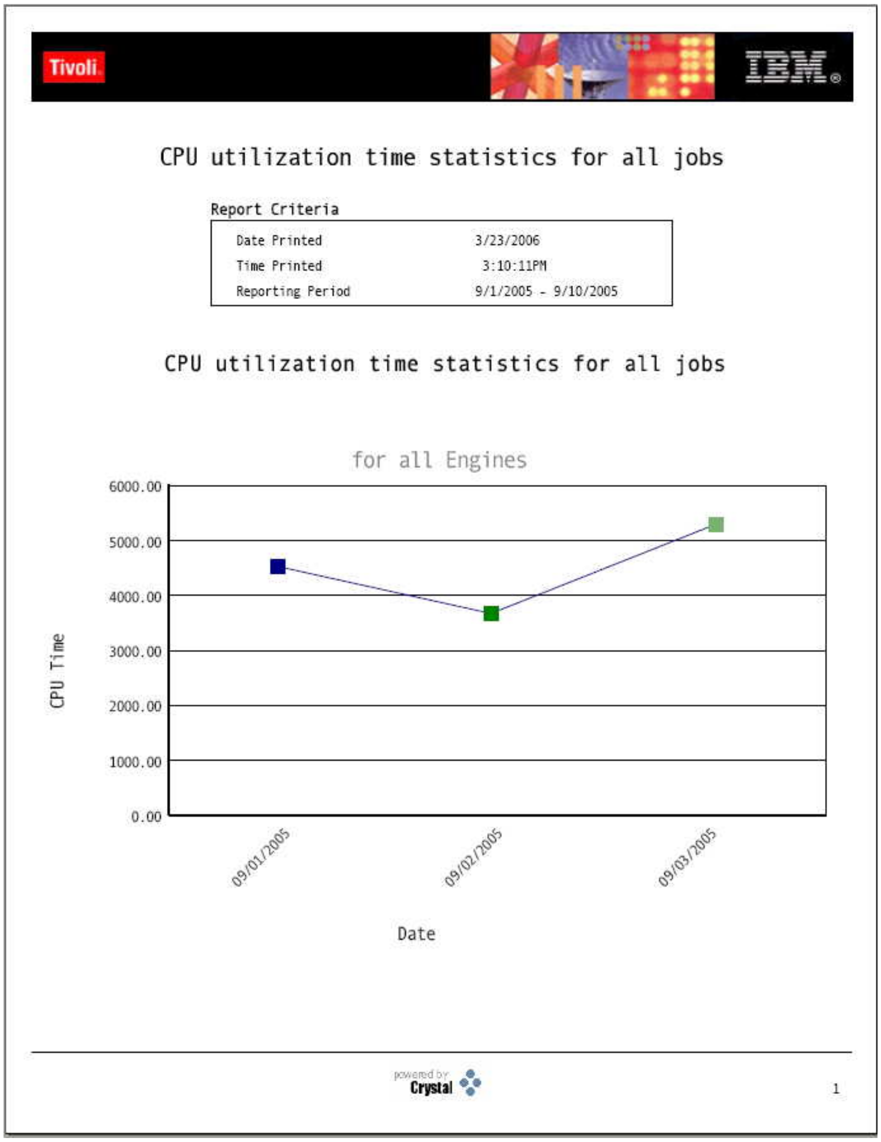
Report Criteria

Date Printed	3/23/2006
Time Printed	3:53:50PM
Reporting Period	9/1/2005 - 9/10/2005

Figure 7. Workstations with highest CPU utilization

3.6 CPU utilization time statistics for all jobs

This report shows a stacked line charts. The first chart shows the jobs CPU utilization time statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the jobs grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average CPU utilization time measurement for all the jobs that run during the day.



CPU utilization time statistics for all jobs

Report Criteria

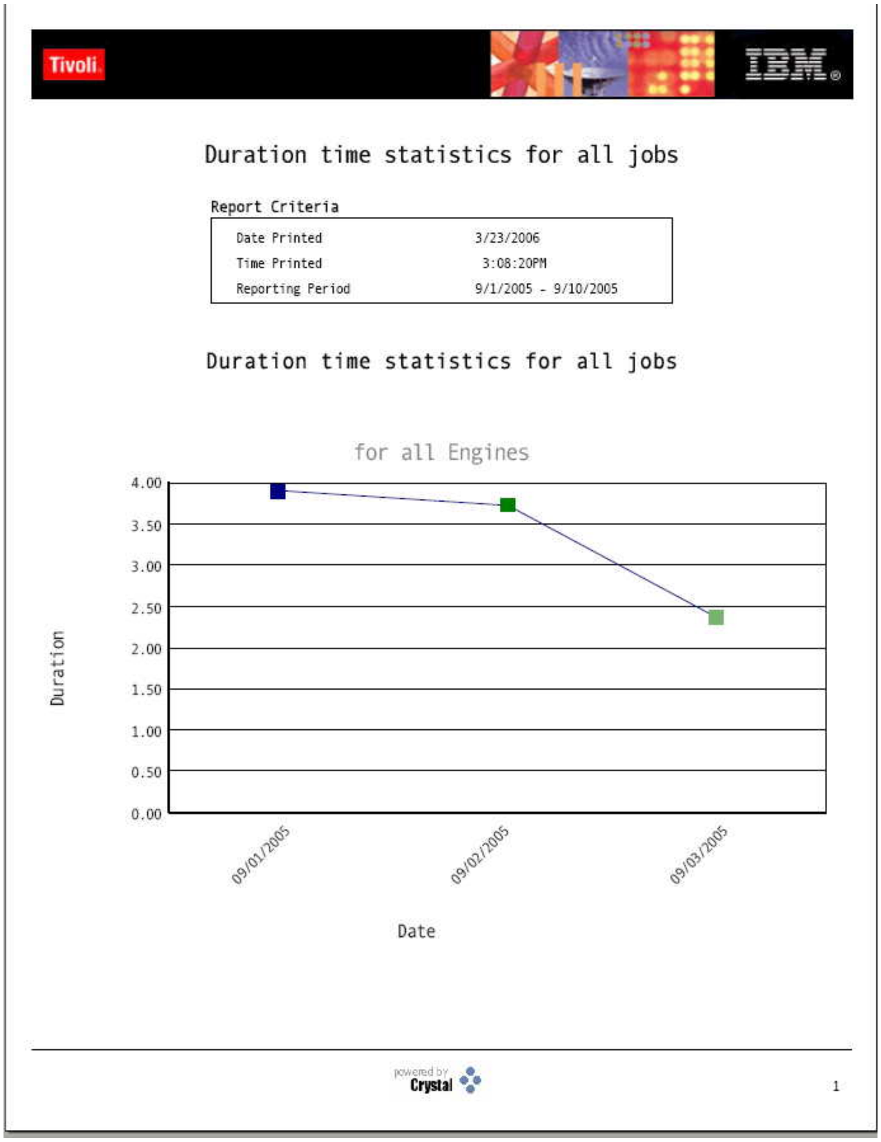
Date Printed	3/23/2006
Time Printed	3:10:11PM
Reporting Period	9/1/2005 - 9/10/2005

CPU utilization time statistics for all jobs

Figure 8. CPU utilization time statistics for all jobs

3.7 Duration time statistics for all jobs

This report shows a stacked line charts. The first chart shows the jobs duration time statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the jobs grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average run duration time measurement for all the jobs that run during the day.



Duration time statistics for all jobs

Report Criteria

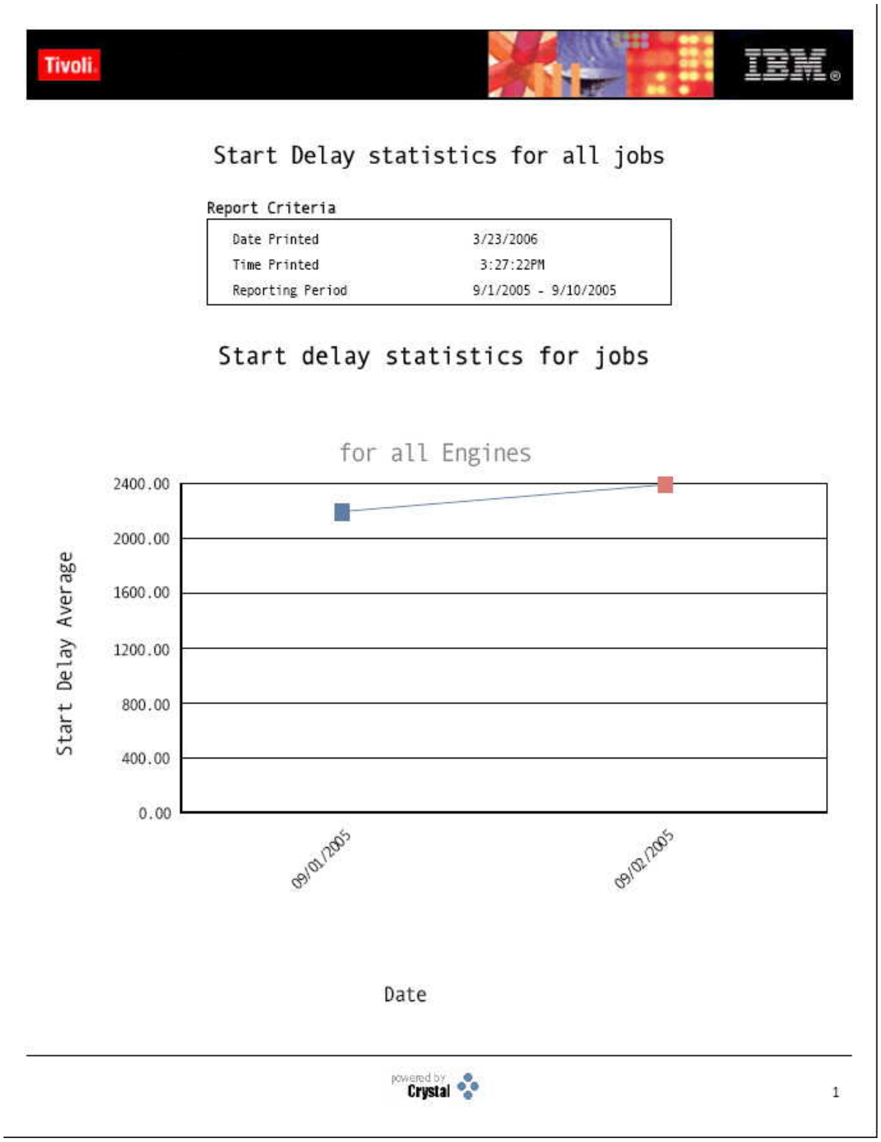
Date Printed	3/23/2006
Time Printed	3:08:20PM
Reporting Period	9/1/2005 - 9/10/2005

Duration time statistics for all jobs

Figure 9, Duration time statistics for all jobs

3.8 Start Delay statistics for all jobs

This report shows a stacked line charts. The first chart shows the jobs statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the jobs grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average delay time measurement for all the jobs that run during the day.



Start Delay statistics for all jobs

Report Criteria

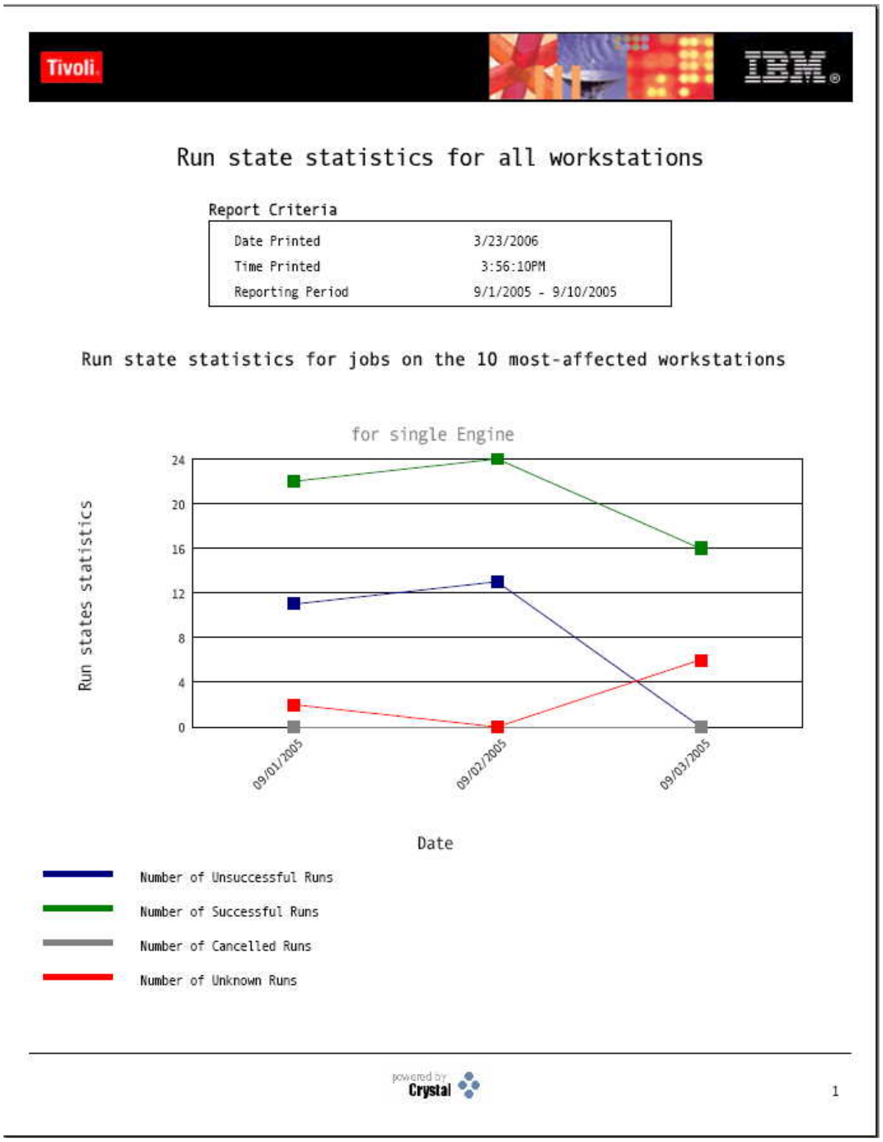
Date Printed	3/23/2006
Time Printed	3:27:22PM
Reporting Period	9/1/2005 - 9/10/2005

Start delay statistics for jobs

Figure 10. Start Delay statistics for all jobs

3.9 Run state statistics for all workstations

This report shows a stacked line charts. The first chart shows the workstations statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the workstations grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis compares the number of successful, unsuccessful, cancelled and unknown runs for all the jobs that run on the workstations.



Run state statistics for all workstations

Report Criteria

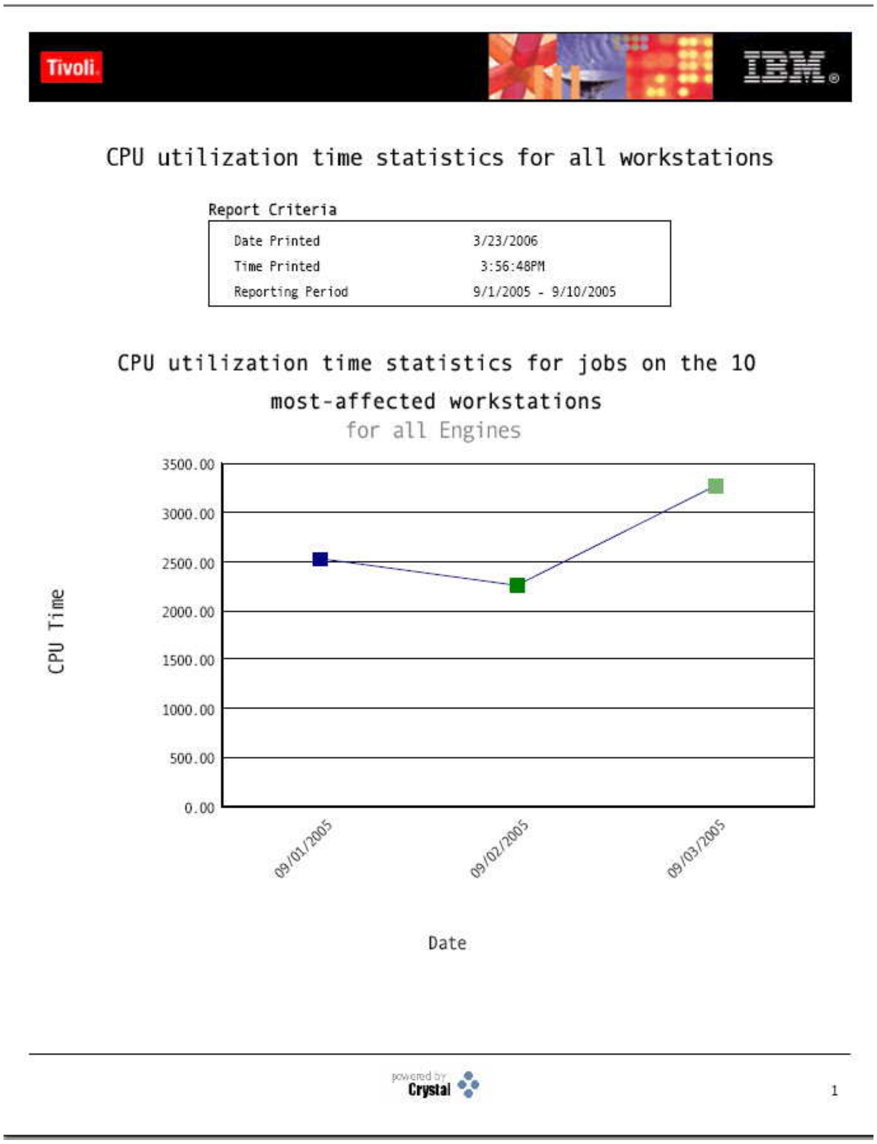
Date Printed	3/23/2006
Time Printed	3:56:10PM
Reporting Period	9/1/2005 - 9/10/2005

Run state statistics for jobs on the 10 most-affected workstations

Figure 11. Run state statistics for all workstations

3.10 CPU utilization time statistics for all workstations

This report shows a stacked line charts. The first chart shows the workstations CPU utilization time statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the workstations grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average CPU utilization time measurement for all the jobs that run on the workstations.



CPU utilization time statistics for all workstations

Report Criteria

Date Printed	3/23/2006
Time Printed	3:56:48PM
Reporting Period	9/1/2005 - 9/10/2005

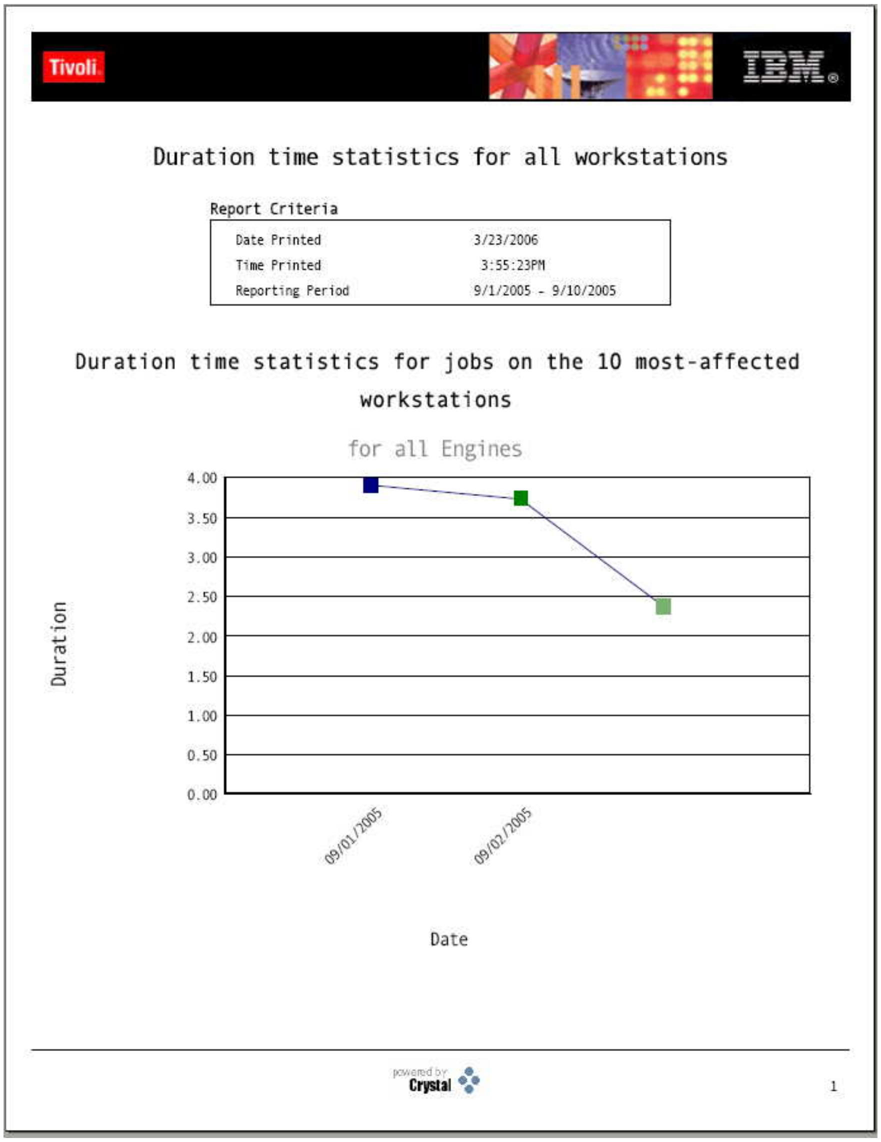
CPU utilization time statistics for jobs on the 10 most-affected workstations

for all Engines

Figure 12. CPU utilization time statistics for all workstations

3.11 Duration time statistics for all workstations

This report shows a stacked line charts. The first chart shows the workstations duration time statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the workstations grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average run duration time measurement for all the jobs that run on the workstations.



Duration time statistics for all workstations

Report Criteria

Date Printed	3/23/2006
Time Printed	3:55:23PM
Reporting Period	9/1/2005 - 9/10/2005

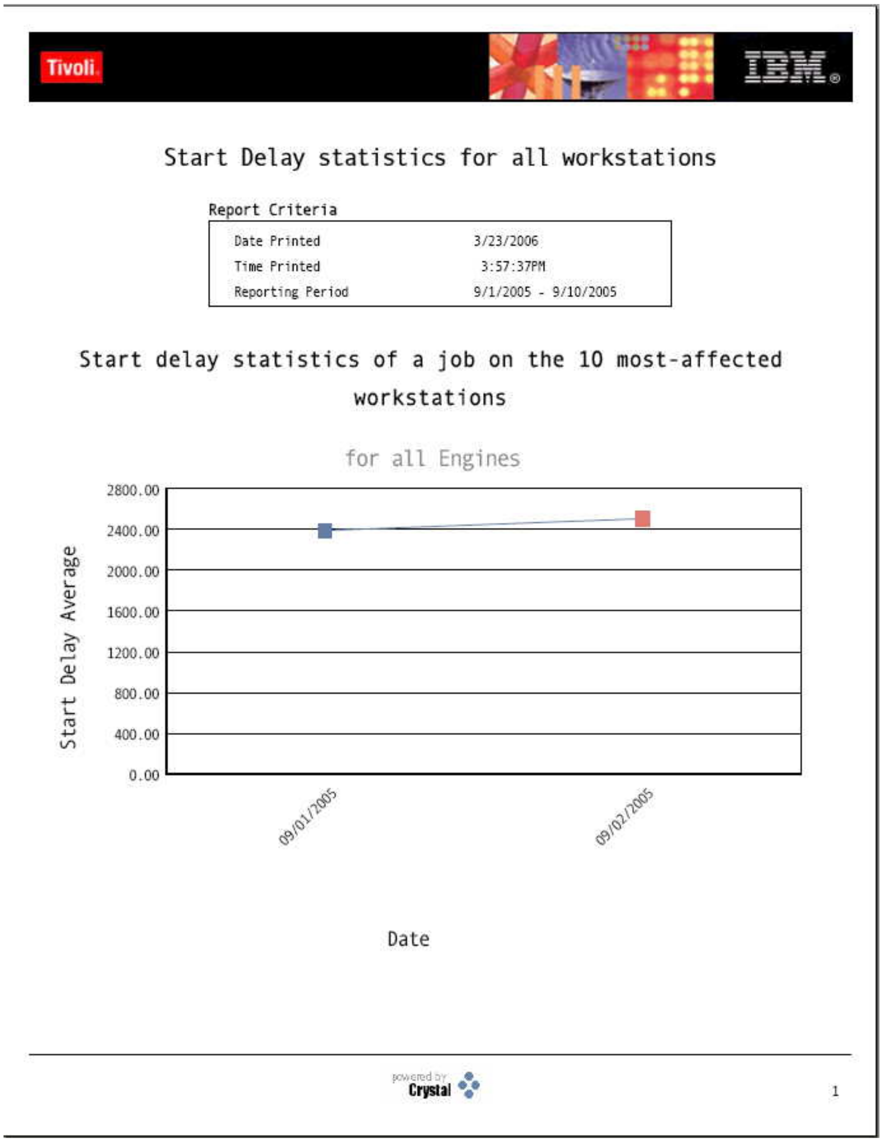
Duration time statistics for jobs on the 10 most-affected workstations

for all Engines

Figure 13, Duration time statistics for all workstations

3.12 Start Delay statistics for all workstations

This report shows a stacked line charts. The first chart shows the workstations statistics for all Tivoli Workload Scheduler engines, whereas the other charts show the statistics for the workstations grouped by Tivoli Workload Scheduler engine. The x-axis displays the data (day-by-day) during the time range selected and the y-axis shows the average delay time measurement for all the jobs that run on the workstations.



Start Delay statistics for all workstations

Report Criteria

Date Printed	3/23/2006
Time Printed	3:57:37PM
Reporting Period	9/1/2005 - 9/10/2005

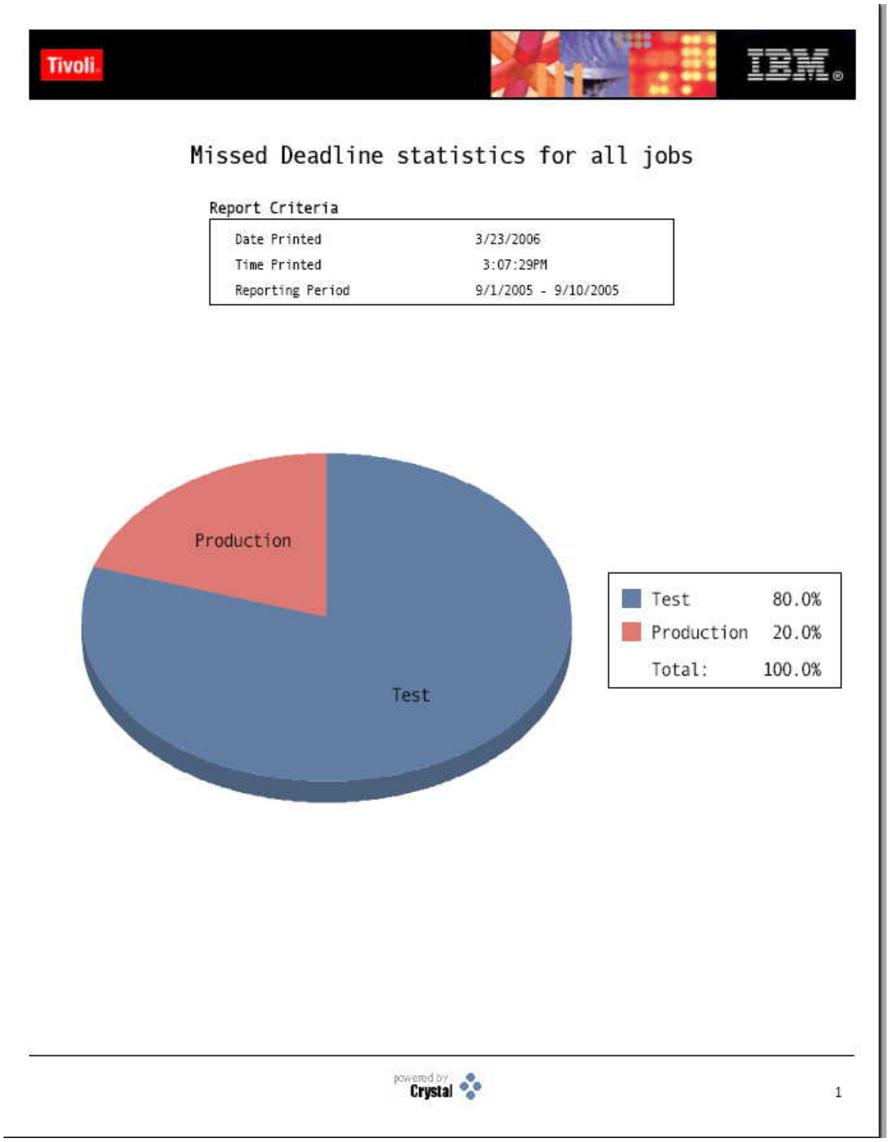
Start delay statistics of a job on the 10 most-affected workstations

for all Engines

Figure 14, Start Delay statistics for all workstations

3.13 Missed Deadline statistics for all jobs

This report shows stacked charts (pie and bar), with 3D visual effect. The first pie chart shows the percentage distributions of the missed deadline statistics for all jobs of the top 10 Tivoli Workload Scheduler engines. The second pie chart, for each engine, shows the percentage distributions of the missed deadline statistics for all jobs of the top 10 workstations. The following bar charts display, for each workstation, the maximum, average and minimum time, in minutes, of the missed deadline statistics for all jobs.



Missed Deadline statistics for all jobs

Report Criteria

Date Printed	3/23/2006
Time Printed	3:07:29PM
Reporting Period	9/1/2005 - 9/10/2005

🟦	Test	80.0%
🟥	Production	20.0%
	Total:	100.0%

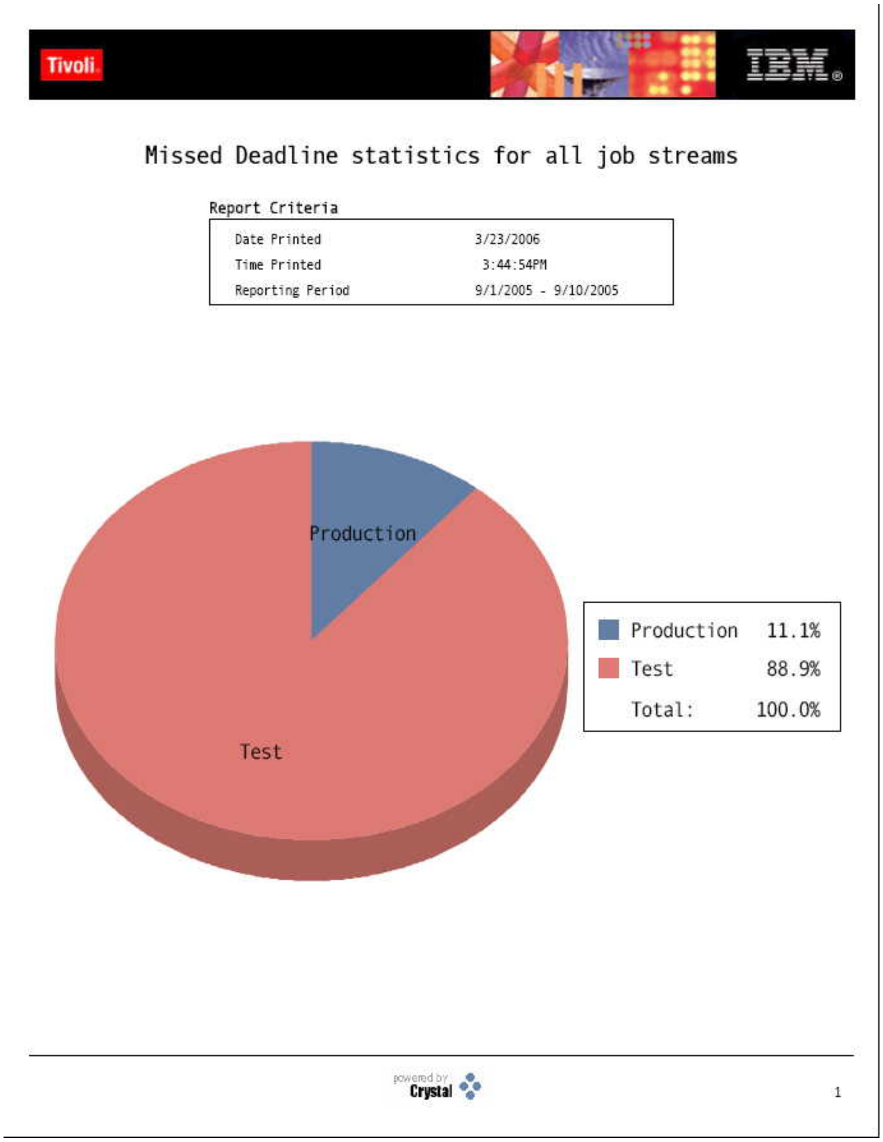
powered by Crystal

1

Figure 15, Missed Deadline statistics for all jobs

3.14 Missed Deadline statistics for all job streams

This report shows stacked charts (pie and bar), with 3D visual effect. The first pie chart shows the percentage distributions of the missed deadline statistics for all job streams of the top 10 Tivoli Workload Scheduler engines. The second pie chart, for each engine, shows the percentage distributions of the missed deadline statistics for all job streams of the top 10 workstations. The following bar charts display, for each workstation, the maximum, average and minimum time, in minutes, of the missed deadline statistics for all job streams.



Missed Deadline statistics for all job streams

Report Criteria

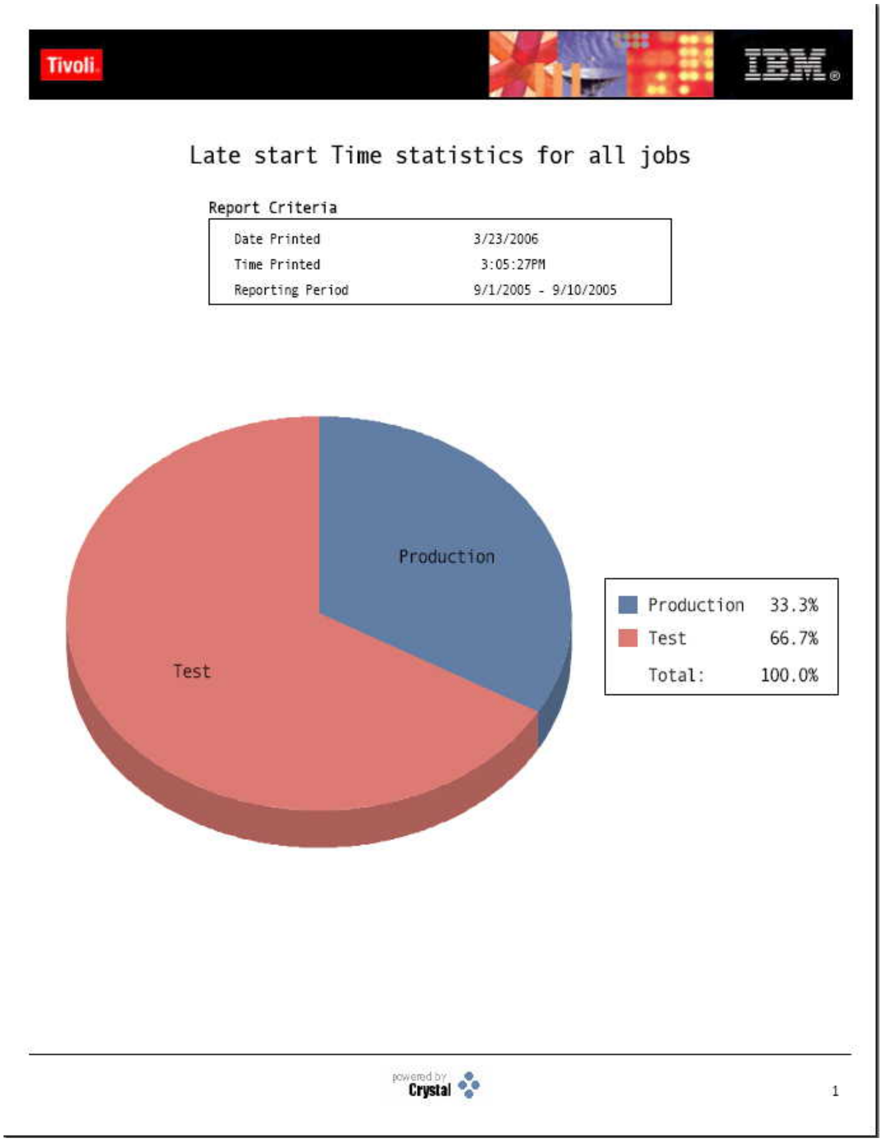
Date Printed	3/23/2006
Time Printed	3:44:54PM
Reporting Period	9/1/2005 - 9/10/2005

Production	11.1%
Test	88.9%
Total:	100.0%

Figure 16, Missed Deadline statistics for all job streams

3.15 Late Start statistics for all jobs

This report shows stacked charts (pie and bar), with 3D visual effect. The first pie chart shows the percentage distributions of the late start statistics for all jobs of the top 10 Tivoli Workload Scheduler engines. The second pie chart, for each engine, shows the percentage distributions of the late start statistics for all jobs of the top 10 workstations. The following bar charts display, for each workstation, the maximum, average and minimum time, in minutes, of the late start statistics for all jobs.



Late start Time statistics for all jobs

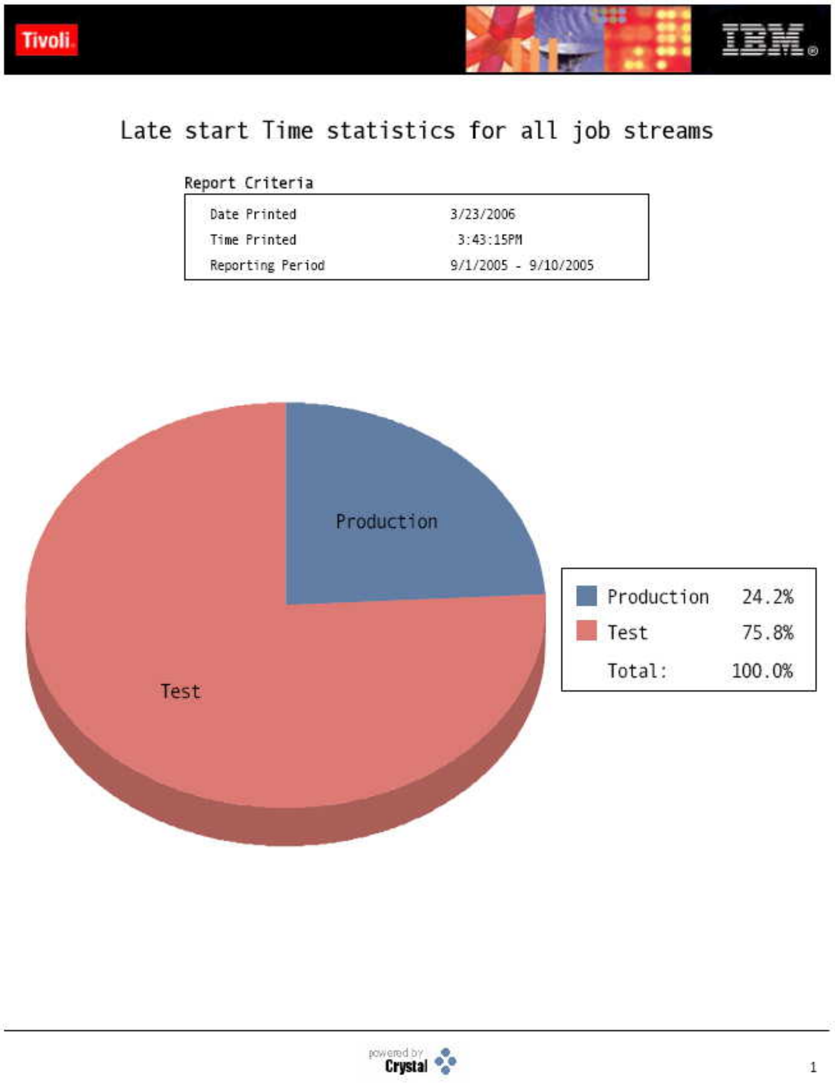
Report Criteria

Date Printed	3/23/2006
Time Printed	3:05:27PM
Reporting Period	9/1/2005 - 9/10/2005

Figure 17, Late Start statistics for all jobs

3.16 Late Start statistics for all job streams

This report shows stacked charts (pie and bar), with 3D visual effect. The first pie chart shows the percentage distributions of the late start statistics for all job streams of the top 10 Tivoli Workload Scheduler engines. The second pie chart, for each engine, shows the percentage distributions of the late start statistics for all job streams of the top 10 workstations. The following bar charts display, for each workstation, the maximum, average and minimum time, in minutes, of the late start statistics for all job streams.



Late start Time statistics for all job streams

Report Criteria

Date Printed	3/23/2006
Time Printed	3:43:15PM
Reporting Period	9/1/2005 - 9/10/2005

Figure 18, Late Start statistics for all jobs

4 Installing and configuring the warehouse pack

This section describes the installation and configuration of the warehouse pack.

4.1 Prerequisite hardware and software

Before installing the warehouse pack for IBM Tivoli Workload Scheduler, you must install the following software:

- IBM Tivoli Workload Scheduler, Version 8.3
- Perl level V5.8.0 or higher
- IBM Tivoli Data Warehouse, Version 1.2 or Version 1.3 and its fix pack and prerequisites

This warehouse pack supports central data warehouses on DB2 UDB for Windows and UNIX systems, this warehouse pack supports data marts on DB2 UDB for Windows and UNIX systems.

The Monotype Sans Duospace WT font must be installed on the double byte system that runs the Crystal Enterprise Server used for either Tivoli Data Warehouse historical reports or the operational reports. The font Monotype Sans Duospace WTJ (file name mtsansdj.ttf) is currently available on the Host On-Demand Support page at Web site:

http://www.ibm.com/software/webservers/hostondemand/support.html

Refer to the *Tivoli Data Warehouse Release Notes* and *IBM Tivoli Workload Scheduler Version 8.3 Release Notes* for specific information about hardware prerequisites, database and operating system support, and product prerequisites. For late-breaking news about prerequisites, refer to the following IBM Software Support Web site:

http://www.ibm.com/software/sysmgmt/products/support/

4.2 Product notes and limitations

The correct order in which to load data, from Tivoli Workload Scheduler Symphony files archived in the central data warehouse and then in the mart databases, is the following: Achiever, ETL1, ETL2. It is important to not accumulate more than one archived Symphony file before launching ETL processes to avoid performance problems.

This Warehouse Enablement Pack supports several Tivoli Workload Scheduler networks, each one identified by the Master Name.

When the configuration is composed of more than one Tivoli Workload Scheduler network, it is important to verify, before launching the ETL processes, that all the Archiver procedures running on every engine, have completed their work.

The ETLs use only the data center and the customer standard. For multiple customers you can use the following workaround: you use the name of Tivoli Workload Scheduler Engine such as NameCustomer.NameEngine.

In the migration from IBM Tivoli Workload Scheduler Warehouse Enablement Pack 8.2.x, because there is no related data, the job stream is linked to the master workstation, if it exists, or to the workstation linked to the job related to the job stream.

The "Workstations with highest CPU utilization" and the "Run times statistics for all jobs" reports work for the jobs running on any platforms except Microsoft; in fact, as described in the Tivoli Workload Scheduler documentation, Tivoli Workload Scheduler does not provide a value for the CPU time utilization of jobs running on Microsoft platforms. Due to this limitation, jobs running on Microsoft platforms are never included in these reports.

4.3 Database sizing considerations

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

Tivoli Manager for DB2	Number of components	Number of MsmtTyps	Components	Measurements per day
Total		**8**	**31,300**	**26,000**
Engine	1	0	1	0
Workstation	2	0	300	0
Job stream	1	8	1,000	900
Job	1	8	30,000	25000

Database	Fixed Storage Requirement	Variable Storage Requirement	Estimated Total Storage (Fixed + Variable)
Central data warehouse	10 MB	1.3 GB	1.4 GB
Data mart database	10 MB	94 MB	Approx 110 MB
Control database	10 MB	0	10 MB

4.4 Pre-installation procedures

You are advised to back up the following databases:

1. Central Data Warehouse database (TWH_CDW).
2. Mart database (TWH_MART).
3. Meta Data database (TWH_MD).

4.5 Installation of the warehouse pack

Before installing the warehouse pack, record the user IDs, passwords, and alias in the following table. You need this information to follow the installation procedures that are described in *Installing and Configuring Tivoli Data Warehouse*.

ODBC source	User ID	Password	Database type	Server name, alias, or file path
ITWS The default data source name for the ODBC connection is TWH_CDW.	db2admin, db2inst1		DB2 UDB	TWH_CDW

Install the warehouse pack as described in *Installing and Configuring Tivoli Data Warehouse*, using the installation properties file (twh_install_props.cfg file):

Location of the twh_install_props.cfg file	In the *tdw_weps/aws/v8300* directory of the installation media for the warehouse pack

4.6 Post-installation procedures

After you install the warehouse pack, you must determine how you want to schedule the ETLs. You can schedule only AWS_c05_s010_WorkloadScheduler_Pre_Extract to schedule the initialization AWS_c05_WorkloadScheduler_Process process. See the information about installing warehouse packs in *Installing and Configuring Tivoli Data Warehouse* for the procedure to schedule ETLs using these processes:

Initialization process AWS_c05_WorkloadScheduler_Process

Process dependencies Located in the AWS_Tivoli_Workload_Scheduler_V8.3.0_Subject_Area

The processes should be run in the following order:

1. AWS_c05_WorkloadScheduler_Process
2. AWS_m05_WorkloadScheduler_Process

The **tws_launch_archiver.pl** file contains the following archive process parameters:

- Installation configuration session:
  - $TWS_INSTALL_DIR specifies information related to Tivoli Workload Scheduler Installation Directory, such as, for a Windows environment: $TWS_INSTALL_DIR="E:/TWS"
- Database configuration session, specifies information related to database configuration:
  - $DB2_DIR specifies DB2 Local Installation directory, such as, for a Windows environment: $DB2_DIR="C:/Program Files/SQLLIB"
  - $DB2PROFILE_DIR specifies DB2 Local profile directory for a UNIX environment, such as DB2PROFILE_DIR="/home/db2inst1/sqllib"
  - $TWH_CDW specifies DB2 TWH CDW database instance, such as $TWH_CDW="TWH_CDW"
  - $DB2USER specifies DB2 TWH CDW User name, such as $DB2USER="db2admin"
  - $DB2PASSWORD specifies DB2 TWH CDW User password, such as $DB2PASSWORD="db2admin"

4.7 Migration from a previous release of the warehouse pack

The following database objects have changed since the previous release of the warehouse pack.

Added objects:

- AWS.D_CUST table, see page 75.
- AWS_c05_WorkloadScheduler_Process process, see page 40.
- AWS_m05_WorkloadScheduler_Process process, see page 40.
- aws_cdw_migrate.db2 script, see page 36.

Changed objects:

- AWS.F_RUNSTATES_DAY table, see page 71.
- AWS. F_RUNTIMES_DAY table, see page 72.
- AWS.D_WORKSTATION table, see page 74.
- AWS.D_JOBSTREAM table, see page 74.
- AWS.D_JOB table, see page 74.

Removed objects:

- AWS_c10_ETL_Process.

In order for the warehouse pack to function correctly, you must perform the following procedure:

1. Back up the following databases:
   - Central Data Warehouse database (TWH_CDW).
   - Mart database (TWH_MART).
   - Meta Data database (TWH_MD).
2. Perform the aws_update_start_delay_cdw.db2 script, located in *<TWH_INSTALL_DIR>/apps/aws/v8300/misc* directory to update the Tivoli Workload Scheduler 8.2.x start delay data.

   Note: If you migrate from IBM Tivoli Workload Scheduler Warehouse Enablement Pack 8.2.0 fix pack 2 or IBM Tivoli Workload Scheduler Warehouse Enablement Pack 8.2.1 fix pack 1, or if you had already performed this script, you must skip this step. In any case this script has to be run only once after the Warehouse Enablement Pack installation. The command is the following:

   db2 –f aws_update_start_delay_cdw.db2 –t –l output_startdelaycdw.txt.
3. Perform the aws_update_duration_cdw.db2 script, located in *<TWH_INSTALL_DIR>/apps/aws/v8300/misc* directory to update the Tivoli Workload Scheduler 8.2.x duration data.

   Note: If you migrate from IBM Tivoli Workload Scheduler Warehouse Enablement Pack 8.2.0 fix pack 3 or IBM Tivoli Workload Scheduler Warehouse Enablement Pack 8.2.1 fix pack 1, or if you had already performed this script, you must skip this step. In any case this script has to be run only once after the Warehouse Enablement Pack installation. The command is the following:

   db2 –f aws_duration_delay_cdw.db2 –t –l output_durationcdw.txt.
4. Perform the aws_migrate_cdw.db2 script, located in *<TWH_INSTALL_DIR>/apps/aws/v8300/misc* directory to migrate the Tivoli Workload Scheduler 8.2.x data.

   Note: The time required to migrate data from Tivoli Workload Scheduler 8.2.x to Tivoli Workload Scheduler 8.3 format, strictly depends on the amount of data to migrate. This script could require a lot of time (several hours) for completion. It must be run only once after the Warehouse Enablement Pack installation. The command is the following:

   db2 –f aws_migrate_cdw.db2 –t –l output_migratecdw.txt.
5. Perform the AWS_m05_WorkloadScheduler_Process process to retrieve all migration data from the Mart database.
6. Respecify the dependencies between processes and re-schedule the processes that are to run automatically. See “Post-installation procedures” on page 35.

Note: The migration process of Tivoli Workload Scheduler historical data does not supports duplication of Master and its Domain names. If there are more than one Tivoli Workload Scheduler network with the same names, they will be considered as a unique Tivoli Workload Scheduler network.

4.8 Uninstallation of the warehouse pack

Perform the following step to uninstall the warehouse pack:

1. Uninstall the warehouse pack as described in *Installing and Configuring Tivoli Data Warehouse*.

The following staging tables in the central data warehouse are removed:

- Aws.Cust_Lookup
- Aws.Centr_Lookup
- Aws.Tws_Workstation_P

- Aws.Tws_Jobstream_P
- Aws.Tws_Job_P

When the warehouse pack is uninstalled, all tables of both star schemas and all staging tables are removed, but the data in the central data warehouse remains and is still useable by other applications.

In order for the warehouse pack to function correctly, you must perform the following step:

1. Remove the old version of reports displayed in the Crystal Management console:
    a) Connect to the Crystal Management console. Log on as administrator.
    b) Click **Manage folders**.
    c) Select **Tivoli reports**, then **Tivoli Workload Scheduler**.
    d) Click **Delete** to remove the reports.

4.9 Multiple data centers

After you install the warehouse pack, you can configure Tivoli Data Warehouse to separate data for multiple data centers. To set this up, you must create SQL scripts with the following values:

Information for scripts	**Value or location**
Field in source data	Fully qualified host name
Name of lookup table	Aws.Centr_lookup table
Name of center list	TWG.Centr

For the procedural instructions and example SQL statements, see the information about warehouse pack installation in the *Installing and Configuring Tivoli Data Warehouse* guide.

After the initial configuration for multiple data centers, you must modify the tables when data centers are added and removed.

4.10 Multiple customer environments

After you install the warehouse pack, you can configure Tivoli Data Warehouse to separate data for the multiple customer environments. To set this up, you must create SQL scripts with the following values:

Information for scripts	**Value or location**
Field in source data	Fully qualified host name
Name of lookup table	Aws.Cust_lookup table
Column to use for lookup	Cust_Id
Name of customer list	TWG.Cust

For the procedural instructions and example SQL statements, see the information about warehouse pack installation in the *Installing and Configuring Tivoli Data Warehouse* guide.

After your initial configuration of the multiple customer environment, you must modify the tables when customers are added and removed.

5 Maintenance and problem determination

This section describes maintenance tasks for the warehouse pack.

5.1 Backing up and restoring

This section describes additional information about backing up and restoring data for the warehouse pack.

5.2 Pruning data

To manage the high volume of measurements data, use the Twg.Prune_Msmt_Control and Aws.Prune_Mart_Control tables to prune that data. The Twg.Prune_Msmt_Log and Aws. Prune_Msmt_Log tables keep a history of data pruning.

5.2.1 Central data warehouse

By default the data older than one year is pruned when the CDW_c05_Prune_and_Mark_Active process runs. This process is within the CDW_Tivoli_Data_Warehouse_v1.2.0_Subject_Area. By default, this process runs daily at 6:00 a.m.

5.2.1.1 Pruning event data (table Prune_Event_Ctrl)

Tivoli Data Warehouse uses the event age and the event date and time to find aged events. Aged events are ones that are older than the value specified in the Event_Age column. Then, the following data is pruned in this order:

1. The parent event that is an aged event (in the EventAttr table)
2. Either the source or target that is an aged event (in the EventReln table)
3. The relationship that involves an aged event (in the CEReln table)
4. The aged event is removed from the Event table

5.2.1.2 Pruning measurement data (table Prune_Msmt_Control)

Measurement data is pruned from the Msmt table every year. This is based on the age specified in the PMsmtC_Age_In_Days column.

5.2.2 Data mart

Pruning data from the fact tables is implemented in the AWS_m05_s040_WorkloadScheduler_Prune. The prune mart control table Aws.Prune_Mart_Control governs which data is pruned and contains a date duration value for the Aws.F_RunStates_Day and Aws.F_RunTimes_Day fact tables. By default, all hourly and daily fact data older than one year is pruned when the process step runs. Additionally, all weekly and monthly fact data older than one year is pruned. The AWS.hPrune_Mart_Log table keeps a history of data pruning.

5.3 Extraction control (table Extract_Control)

The extraction control table assists you in incrementally extracting data from a source database. For an example of incremental extraction, see the *Enabling an Application for Tivoli Data Warehouse* guide.

ExtCtl_Source VARCHAR (120)	ExtCtl_Target VARCHAR (120)	ExtCtl_From_RawSeq CHAR (10)	ExtCtl_to_RawSeq CHAR (10)	ExtCtl_From_IntSeq BIGINT	ExtCtl_To_IntSeq BIGINT	ExtCtl_From_DtTm TIMESTAMP	ExtCtl_To_DtTm TIMESTAMP	MSrc_Corr_Cd CHAR (6)
AWS.Stage_TWS_Archive	TWG.MSMT	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_RUNSTATES_METRIC	AWS.T_RUNSTATES_METRIC	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_RUNTIMES_METRIC	AWS.T_RUNTIMES_METRIC	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_CUST	AWS.T_CUST	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_ENGINE	AWS.T_ENGINE	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_WORKSTATION	AWS.D_WORKSTATION	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_JOB	AWS.D_JOB	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VD_JOBSTREAM	AWS.D_JOBSTREAM	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VF_RUNSTATES_DAY	AWS.F_RUNSTATES_DAY	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS
AWS.VF_RUNTIMES_DAY	AWS.F_RUNTIMES_DAY	x	x	0	0	1970-01-01-00.00.00.000	1970-01-01-00.00.00.000	AWS

5.4 Problem determination

For common problems and solutions, see the *Installing and Configuring Tivoli Data Warehouse* guide.

6 ETL processes

The warehouse pack has the following processes:

- Archiver
- AWS_c05_WorkloadScheduler_Process
- AWS_m05_WorkloadScheduler_Process

The Archiver process extracts Tivoli Workload Scheduler data from the archived plans and moves it into flat files and then loads the flat files content into staging tables in Tivoli Data Warehouse.

The ETL processes retrieve the Tivoli Workload Scheduler data from the Central Data Warehouse staging tables and load it first in the Central Data Warehouse database and then in the Data Mart tables. These two processes make up the entire IBM Tivoli Workload Scheduler ETL process, and the AWS_c05_WorkloadScheduler_Process must be performed first followed by the AWS_m05_WorkloadScheduler_Process. These processes are linked in such a way that the second starts if the first one is successful. Also the single steps are linked in such a way that each step must start if the previous one is successful. So, if the first step of the AWS_c05_WorkloadScheduler_Process process is scheduled, the IBM Tivoli Workload Scheduler Extarct Transform and Load process is also scheduled. If you want to perform only the ETL2 process, you can schedule only the first step of the AWS_m05_WorkloadScheduler_Process process.

For the procedure to schedule ETLs, see the information about configuring warehouse packs in *Installing and Configuring Tivoli Data Warehouse*.

6.1 AWS_c05_WorkloadScheduler_Process

This process, named ETL1, extracts a snapshot of the job workload scheduler data in your environment, based on the IBM Tivoli Workload Scheduler. Schedule this process to run daily. If this process is not run for a given day, measurements for that day will not be present in the central data warehouse and cannot be recovered later.

This process has the following steps:

1. AWS_c05_s010_WorkloadScheduler_Pre_Extract

   This step recreates the staging tables needed by the following steps

2. AWS_c05_s020_WorkloadScheduler_Extract

   This step extracts the IBM Tivoli Workload Scheduler data from the source Tivoli Workload Scheduler staging tables and stores it in the central data warehouse staging tables. This step is scheduled to run if the AWS_c05_s010_WorkloadScheduler_Pre_Extract is successful

3. AWS_c05_s030_WorkloadScheduler_Load

   This step loads data from the staging table to the TWH_CDW database. This step is scheduled to run if the AWS_c05_s020_WorkloadScheduler_Extract is successful

6.2 AWS_m05_WorkloadScheduler_Process

This process, named ETL2, extracts IBM Tivoli Workload Scheduler data from the central data warehouse and places the information into the data mart.

This process is scheduled to run if the AWS_c05_WorkloadScheduler_Process is successful.

This process has the following steps:

1. AWS_m05_s010_WorkloadScheduler_Pre_Extract

   This step recreates the staging tables needed by the following steps.

2. AWS_m05_s020_WorkloadScheduler_Extract

This step extracts the data from the TWH_CDW database to the translation tables. This step is scheduled to run if the AWS_m05_s010_WorkloadScheduler_Pre_Extract is successful.

3. AWS_m05_s030_WorkloadScheduler_Load

   This step loads the data from the translation tables to the TWH_MART database. This step is scheduled to run if the AWS_m05_s020_WorkloadScheduler_Extract is successful.

4. AWS_m05_s040_WorkloadScheduler_Prune

   This step prunes old data from the fact table. This step is scheduled to run if the AWS_s05_s030_WorkloadScheduler_Load is successful.

7 Central data warehouse information

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

This section provides an example of how information about IBM Tivoli Workload Scheduler data is stored in Tivoli Data Warehouse. The IBM Tivoli Workload Scheduler components and measurement values are based on this example.

The example uses the following information:

CPUS (Workstation)

Name	Node	WorkstationType	OperSys	DomainName	MasterName	HostName	Symphony_Date
ProdMaster	abc.abc.com	2	WNT	Account	ProdMaster	ProdMaster	2005-09-01-09.17.00
SAPXA1	db2.abc.com	4	UNIX	Account	ProdMaster	SAP1	2005-09-01-09.17.00
DomMan1	pay.xyz.com	1	UNIX	Finance	ProdMaster	DomMan1	2005-09-01-09.17.00
Cmvc.dev	aixl.abc.com	1	UNIX	Dev	ProdMaster	Cmvc.dev	2005-09-01-09.17.00
Build.dev	aixl.abc.com	6	UNIX	Dev	ProdMaster	Build.dev	2005-09-01-09.17.00

SCHEDS (Job stream)

Name	Priority	Status	Planned_StartTime	Actual_StartTime	Actual_Duration	Cpu_Time	MasterName	Jobstream_Id	Jobstream_IA	Jobstream_CPU	DeadLine_Time	LateStart_Time
Payroll	10	6	2005-09-01-09.00.00	2005-09-01-10.24.00	339	8781	ProdMaster	0AABBCCDDDDCCBBAA1	2005-09-01-09.00.00	DomMan1	2005-09-01-18.00.00	2005-09-01-10.00.00
Build	50	6	2005-09-01-11.30.00	2005-09-01-11.32.00	420	10236	ProdMaster	0AABBCCDDDDCCBBAA2	2005-09-01-11.30.00	Build.dev	2005-09-01-15.00.00	2005-09-01-11.45.00
Build	50	6	2005-09-01-20.00.00	2005-09-01-20.00.00	60	578	ProdMaster	0AABBCCDDDDCCBBAA2	2005-09-01-20.00.00	Build.dev	2005-09-01-23.00.00	2005-09-01-23.00.00

JOBS (Job)																
Name	**Priority**	**Login**	**Status**	**Planned_StartTime**	**Actual_StartTime**	**Actual_Duration**	**Cpu_Time**	**Workstation Name**	**JobstreamName**	**Jobstream_CPU**	**Command**	**MasterName**	**Jobstream_Id**	**Jobstrea_IA**	**DeadLine_Time**	**LateStart_Time**
Extract	01	Tws usr22	6	2005-09-01-09.00.00	2005-09-01-10.25.00	60	2312	SAPXA1	Payroll	DomMan1	Extract -db db2 -h ppp -v23; la -al	ProdMaster	0AABBCCDDDDCCBBAA1	2005-09-01-09.00.00	2005-09-01-14.00.00	2005-09-01-09.30.00
CalcPayroll	20	Tws usr33	6	2005-09-01-10.00.00	2005-09-01-11.45.00	180	5903	FTA1	Payroll	DomMan1	Calcpay -in extr.txt - out pav.form	ProdMaster	0AABBCCDDDDCCBBAA1	2005-09-01-09.00.00	2005-09-01-18.00.00	2005-09-01-10.00.00
MakeAll	10	Tws build1	6	2005-09-01-11.30.00	2005-09-01-11.32.05	300	9201	Build.dev	Build	Build.dev	Make 2>&amp1 | tee build.log	ProdMaster	0AABBCCDDDDCCBBAA2	2005-09-01-11.30.00	2005-09-01-15.00.00	2005-09-01-11.45.00
MakeAll	10	Tws build1	6	2005-09-01-20.00.00	2005-09-01-20.00.05	60	570	Build.dev	Build	Build.dev	Make 2>&amp1 | tee build.log	ProdMaster	0AABBCCDDDDCCBBAA2	2005-09-01-20.00.00	2005-09-01-23.00.00	2005-09-01-23.00.00

The following assumptions are used for this example:

- The archive ran on 1 September 2004 (the Symphony data has been collected).
- The ETL processes ran on 2 September 2004 at 12:00 a.m. (midnight).

The following timeline was used:

- All of the IBM Tivoli Workload Scheduler components listed above are discovered for the first time.
- The number of workstations (“ProdMaster, SAPXA1, DomMan1, Cmvc.dev, Build.dev”) discovered is 5.
- The number of job streams (“Payroll, Build”) in a group discovered is 3 (two time “Build”).

- The number of jobs (“Extract, CalcPayroll, MakeAll”) discovered is 4 (two time “MakeAll”).
- The run states measurement discovered are 10: Number or Runs and Number of Successful Runs 5, Number of Unsuccessful Runs, Number of Unknown Runs and Number of Cancelled Runs 0.
- The run times measurement discovered are 20: Duration, Start Delay and CPU Time 5, Deadline Time 2 and Late Start Time 3.

This section about the information in the central data warehouse is intended primarily for report designers and warehouse pack creators. For information about reports, see “Reports” on page 11.

The values for the “duration” fields are expressed in minutes in the Symphony file and in the Aws tables, and the corresponding values in the Central Data Warehouse are expressed in hours.

The values for the fields WorkstationType and OperSys of the Aws.Tws_Workstation_P table are stored in the central data warehouse as a descriptive phrase. The following tables show how the values of WorkstationType and OperSys in the Aws.Tws_Workstation_P table correspond to the values in the Twg.CompAttr table.

WorkstationType	**CompAttr_Val**
1	Domain Manager
2	Master
3	Host
4	X-Agent
5	Standard Agent
6	Fault Tolerant Agent

OperSys	**CompAttr_Val**
WNT	Windows
UNIX	Unix
OTHR	Other

Shaded columns in the following tables are translated. These columns are also marked with an asterisk (*) after the column name.

7.1 Component configuration

The following sections describe the component configuration.

7.1.1 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**	**MSrc_Corr_Cd CHAR (6)**
SCHED_ENGINE	NULL	Workload Scheduker Master	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	MODEL1

CompTyp_Cd CHAR (17)	CompTyp_Parent_Cd CHAR (17)	CompTyp_Nm * VARCHAR (120)	CompTyp_Strt_DtTm TIMESTAMP	CompTyp_End_DtTm TIMESTAMP	MSrc_Corr_Cd CHAR (6)
SCHED_WORKSTATION	NULL	Workload Scheduker Workstation	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	MODEL1
SCHED_JOB	NULL	Workload Scheduker Job	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	MODEL1
SCHED_JOBSTREAM	NULL	Workload Scheduker Job stream	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	MODEL1
SCHED_HOST	NULL	Workload Scheduker Host	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS

7.1.2 Component extension (table Comp_ext)

There is no component extension for this warehouse pack.

Comp_ID INTEGER	Comp_Long_Nm VARCHAR (3500)

7.1.3 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_DtTm TIMESTAMP	Comp_End_DtTm TIMESTAMP	Comp_Ds VARCHAR (254)	MSrc_Corr_Cd CHAR (6)
1	SCHED_ENGINE	CDW	1		ProdMaster	Account	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Master	AWS
2	SCHED_HOST	CDW	1		abc.abc.com		2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Host	AWS
3	SCHED_WORKSTATION	CDW	1		ProdMaster	ProdMaster#abc.abc.com	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Workstation	AWS
4	SCHED_HOST	CDW	1		db2.abc.com		2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Host	AWS
5	SCHED_WORKSTATION	CDW	1		SAPXA1	ProdMaster#db2.abc.com	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Workstation	AWS

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_DtTm TIMESTAMP	Comp_End_DtTm TIMESTAMP	Comp_Ds VARCHAR (254)	MSrc_Corr_Cd CHAR (6)
							00.00.00.000000	00.00.00.000000		
6	SCHED_HOST	CDW	1		pay.xyz.com		2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Host	AWS
7	SCHED_WORKSTATION	CDW	1		DomMan1	ProdMaster#pay.xyz.com	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Workstation	AWS
8	SCHED_HOST	CDW	1		aix1.abc.com		2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Host	AWS
9	SCHED_WORKSTATION	CDW	1		Cmvc.dev	ProdMaster#aix1.abc.com	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Workstation	AWS
10	SCHED_WORKSTATION	CDW	1		Build.dev	ProdMaster#aix1.abc.com	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Workstation	AWS
11	SCHED_JOBSTREAM	CDW	1		Payroll	ProdMaster#DomMan1	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Job Stream	AWS
12	SCHED_JOB	CDW	1		Extract	ProdMaster#SAPXA1#Payroll	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Job	AWS
13	SCHED_JOB	CDW	1		CalcPayroll	ProdMaster#DomMan1#Payroll	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Job	AWS
14	SCHED_JOBSTREAM	CDW	1		Build	ProdMaster#Build.dev	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Job Stream	AWS
15	SCHED_JOB	CDW	1		MakeAll	ProdMaster#Build.dev#Build	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Job	AWS

7.1.4 Component relationship type (table RelnTyp)

RelnTyp_Cd CHAR (6)	RelnTyp_Nm * VARCHAR (120)	MSrc_Corr_Cd CHAR (6)
PCHILD	Parent Child Relation	MODEL1

RelnTyp_Cd CHAR (6)	RelnTyp_Nm * VARCHAR (120)	MSrc_Corr_Cd CHAR (6)
MEMBER	Member Relation	MODEL1
OWNS	Owns Relation	MODEL1
CNTRLS	Controls Relation	MODEL1
NETWRK	Network Relation	MODEL1
RUNSON	Runs on Relation	MODEL1
INVOKE	Invoke Relation	MODEL1
DEPEND	Depend Relation	MODEL1
PCONT	Physical Containment Relation	MODEL1
MONITR	Monitoring Relation	MODEL1
INSTON	Installed on Relation	MODEL1
LCONT	Logical Containment Relation	MODEL1
DERIVE	Derive Relation	MODEL1
* This column is translated.		

7.1.5 Component relationship rule (table RelnRul)

CompTyp_Source_Cd CHAR (17)	CompTyp_Target_Cd CHAR (17)	RelnTyp_Cd CHAR (6)	RelnRul_Strt_DtTm TIMESTAMP	RelnRul_End_DtTm TIMESTAMP
SCHED_ENGINE	SCHED_WORKSTATION	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000
SCHED_ENGINE	SCHED_JOBSTREAM	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000
SCHED_ENGINE	SCHED_JOB	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000
SCHED_JOB	SCHED_JOBSTREAM	LCONT	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000
SCHED_JOB	SCHED_WORKSTATION	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000

CompTyp_Source_Cd CHAR (17)	CompTyp_Target_Cd CHAR (17)	RelnTyp_Cd CHAR (6)	RelnRul_Strt_DtTm TIMESTAMP	RelnRul_End_DtTm TIMESTAMP
SCHED_JOBSTREAM	SCHED_WORKSTATION	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000
SCHED_WORKSTATION	SCHED_HOST	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000

7.1.6 Component relationship (table CompReln)

CompReln_ID INTEGER	Comp_Source_ID INTEGER	Comp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	CompReln_Strt_DtTm TIMESTAMP	CompReln_End_DtTm TIMESTAMP	MSrc_Corr_Cd CHAR (6)
1	1	3	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
2	3	2	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
3	1	5	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
4	5	4	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
5	1	7	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
6	7	6	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
7	1	9	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
8	9	8	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
9	1	10	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
10	10	8	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
11	1	11	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
12	1	12	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
13	12	11	LCONT	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
14	12	5	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
15	1	13	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
16	13	11	LCONT	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS

CompReln_ID INTEGER	Comp_Source_ID INTEGER	Comp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	CompReln_Strt_DtTm TIMESTAMP	CompReln_End_DtTm TIMESTAMP	MSrc_Corr_Cd CHAR (6)
17	13	7	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
18	1	14	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
19	12	14	LCONT	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
20	12	10	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
21	1	15	PCHILD	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
22	15	14	LCONT	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
23	11	7	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
24	14	10	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS
25	15	10	RUNSON	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	AWS

7.1.7 Component type keyword (table CompTyp_Keyword)

There is no component type keyword for this warehouse pack.

Keyword_ID INTEGER	CompTyp_Cd CHAR (17)	Keyword_Nm VARCHAR (230)	Keyword_Parent_Nm VARCHAR (230)

7.1.8 Attribute type (table AttrTyp)

AttrTyp_Cd CHAR (17)	AttrTyp_Nm * VARCHAR (120)	MSrc_Corr_Cd CHAR (6)
LAST_IP_ADDRESS	Last IP Address	MODEL1
IP_HOSTNAME	IP Host Name	MODEL1
NAME	Name	MODEL1
OS_NAME	Operating System Name	MODEL1

AttrTyp_Cd CHAR (17)	AttrTyp_Nm * VARCHAR (120)	MSrc_Corr_Cd CHAR (6)
SCHED_WRKST_TYPE	Workload Scheduler Workstation Type	AWS
SCHED_DOMAIN	Workload Scheduler Domain	AWS
SCHED_HOST_NAME	Workload Scheduler Host Name	AWS
SCHED_PRIORITY	Workload Scheduler Priority	AWS
SCHED_LOGIN	Workload Scheduler Login User Name	AWS
SCHED_COMMAND	Workload Scheduler Command String	AWS
SCHED_JOBS_ID	Workload Scheduler Job stream Internal Identifier	AWS
SCHED_JOBS_IA	Workload Scheduler Job stream Input Arrival Time	AWS

* This column is translated.

7.1.9 Attribute rule (table AttrRul)

CompTyp_Cd CHAR (17)	AttrTyp_Cd CHAR (17)	AttrRul_Strt_DtTm TIMESTAMP	AttrRul_End_DtTm TIMESTAMP	AttrRul_Dom_Ind CHAR	AttrTyp_Multi_ Val CHAR (1)
SCHED_ENGINE	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_WORKSTATION	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_WORKSTATION	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_WORKSTATION	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_WORKSTATION	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_JOBSTREAM	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_JOBSTREAM	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	Y
SCHED_JOBSTREAM	SCHED_JOBS_IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	Y
SCHED_JOB	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N

CompTyp_Cd CHAR (17)	AttrTyp_Cd CHAR (17)	AttrRul_Strt_DtTm TIMESTAMP	AttrRul_End_DtTm TIMESTAMP	AttrRul_Dom_Ind CHAR	AttrTyp_Multi_ Val CHAR (1)
SCHED_JOB	SCHED_LOGIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_JOB	SCHED_COMMAND	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	N
SCHED_JOB	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	Y
SCHED_JOB	SCHED_JOBS_IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	N	Y

7.1.10 Attribute domain (table AttrDom)

There is no attribute domain for this warehouse pack.

AttrDom_ID INTEGER	CompTyp_Cd CHAR (17)	AttrTyp_Cd CHAR (17)	AttrDom_Strt_DtTm TIMESTAMP	AttrDom_End_DtTm TIMESTAMP	AttrDom_Val VARCHAR (254)	AttrDom_Ds VARCHAR (254)	MSrc_Corr_Cd CHAR (6)

7.1.11 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)	MSrc_Corr_Cd CHAR (6)
1	1	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Account	AWS
2	3	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Master	AWS
3	3	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Windows	AWS
4	3	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Account	AWS
5	3	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	ProdMaster	AWS
6	5	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	X-Agent	AWS
7	5	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Unix	AWS
8	5	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Account	AWS
9	5	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	SAP1	AWS

CompAttr_ID INTEGER	Comp_ID INTEGER	AttrTyp_Cd CHAR (17)	CompAttr_Strt_DtTm TIMESTAMP	CompAttr_End_DtTm TIMESTAMP	CompAttr_Val VARCHAR (254)	MSrc_Corr_Cd CHAR (6)
10	7	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Domain Manager	AWS
11	7	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Unix	AWS
12	7	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Finance	AWS
13	7	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	DomMan1	AWS
14	9	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Domain Manager	AWS
15	9	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Unix	AWS
16	9	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Dev	AWS
17	9	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Cmvc.dev	AWS
18	10	SCHED_WRKST_TYPE	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Fault Tolerant Agent	AWS
19	10	OS_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Unix	AWS
20	10	SCHED_DOMAIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Dev	AWS
21	10	SCHED_HOST_NAME	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Build.dev	AWS
22	11	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	10	AWS
23	11	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	0AABBCCDDDDCCBBAA1	AWS
24	11	SCHED_ JOBS _IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	2005-09-01-09.00.00	AWS
25	12	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	1	AWS
26	12	SCHED_LOGIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Twsusr22	AWS
27	12	SCHED_COMMAND	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Extract –db db2 –h ppp –v 23; ls -al	AWS
28	12	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	0AABBCCDDDDCCBBAA1	AWS
29	12	SCHED_ JOBS _IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	2005-09-01-09.00.00	AWS

CompAttr_ID INTEGER	Comp_ID INTEGER	AttrTyp_Cd CHAR (17)	CompAttr_Strt_DtTm TIMESTAMP	CompAttr_End_DtTm TIMESTAMP	CompAttr_Val VARCHAR (254)	MSrc_Corr_Cd CHAR (6)
30	13	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	20	AWS
31	13	SCHED_LOGIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Twsusr33	AWS
32	13	SCHED_COMMAND	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Calcpay –in extr.txt –out pay.form	AWS
33	13	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	0AABBCCDDDDCCBBAA1	AWS
34	13	SCHED_ JOBS _IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	2005-09-01-09.00.00	AWS
35	14	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	50	AWS
36	14	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	0AABBCCDDDDCCBBAA2	AWS
37	14	SCHED_ JOBS _IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	2005-09-01-11.30.00	AWS
38	15	SCHED_PRIORITY	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	10	AWS
39	15	SCHED_LOGIN	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Twsbuild1	AWS
40	15	SCHED_COMMAND	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	Make 2>&1 | tee build.log	AWS
41	15	SCHED_JOBS_ID	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	0AABBCCDDDDCCBBAA2	AWS
42	15	SCHED_ JOBS _IA	2005-09-02-00.00.00.000000	9999-01-01-00.00.00.000000	2005-09-01-11.30.00	AWS

7.1.12 Component type relationship (table CTypReln)

There is no component type relationship for this warehouse pack.

CTyp_Source_Cd CHAR (17)	CTyp_Target_Cd CHAR (17)	RelnTyp_Cd CHAR (6)	Reln_Strt_DtTm TIMESTAMP	Reln_End_DtTm TIMESTAMP

7.1.13 Component attribute type relationship (table ATypReln)

There is no component attribute type relationship for this warehouse pack.

ATyp_Source_Cd CHAR (17)	**ATyp_Target_Cd CHAR (17)**	**RelnTyp_Cd CHAR (6)**	**ATReln_Strt_DtTm TIMESTAMP**	**ATReln_End_DtTm TIMESTAMP**

7.2 Component measurement

The following sections describe the component measurement.

7.2.1 Measurement group type (table MGrpTyp)

MGrpTyp_Cd CHAR (6)	**MGrpTyp_Nm * VARCHAR (120)**
CATEG	Category
GROUP	Aggregate Types or Group Functions
STATE	State
TRANS	State Transition Groups
COLL	Data Collection Groups
* This column is translated.	

7.2.2 Measurement group (table MGrp)

MGrp_Cd CHAR (6)	**MGrpTyp_Cd CHAR (6)**	**MGrp_Parent_Cd CHAR (6)**	**MGrp_Nm * VARCHAR (120)**
PERF	CATEG	NULL	Performance
UTIL	CATEG	NULL	Utilization
AVL	CATEG	NULL	Availability
AVG_E	GROUP	NULL	Average Value Exists
MIN_E	GROUP	NULL	Minimum Value Exists

MGrp_Cd CHAR (6)	MGrpTyp_Cd CHAR (6)	MGrp_Parent_Cd CHAR (6)	MGrp_Nm * VARCHAR (120)
MAX_E	GROUP	NULL	Maximum Value Exists
TOT_E	GROUP	NULL	Total Value Exists
STORAG	CATEG	NULL	Storage
* This column is translated.			

7.2.3 Measurement group member (table MGrpMbr)

MGrp_Cd CHAR (6)	MGrpTyp_Cd CHAR (6)	MsmtTyp_ID INTEGER
AVG_E	GROUP	20
AVG_E	GROUP	103
AVG_E	GROUP	104
AVG_E	GROUP	105
AVG_E	GROUP	106
MIN_E	GROUP	20
MIN _E	GROUP	103
MIN _E	GROUP	104
MIN _E	GROUP	105
MIN _E	GROUP	106
MAX_E	GROUP	20
MAX_E	GROUP	103
MAX_E	GROUP	104

MGrp_Cd CHAR (6)	MGrpTyp_Cd CHAR (6)	MsmtTyp_ID INTEGER
MAX_E	GROUP	105
MAX_E	GROUP	106
TOT_E	GROUP	21
TOT_E	GROUP	22
TOT_E	GROUP	100
TOT_E	GROUP	101
TOT_E	GROUP	102

7.2.4 Measurement unit category (table MUnitCat)

MunitCat_Cd CHAR (6)	MunitCat_Nm * VARCHAR (120)
TM	Time Duration
QTY	Quantity
PRC	Percentage
RT	Rate
* This column is translated.	

7.2.5 Measurement unit (table MUnit)

MUnit_Cd CHAR (6)	MUnitCat_Cd CHAR (6)	Munit_Nm * VARCHAR (120)
PRC	PRC	Percentage
Bps	RT	Bytes per Second

MUnit_Cd CHAR (6)	MUnitCat_Cd CHAR (6)	Munit_Nm * VARCHAR (120)
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	4K Pages
* This column is translated.		

7.2.6 Measurement alias names (table MTypReln)

There is no measurement alias names for this warehouse pack.

MTyp_Source_ID INTEGER	MTyp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	MReln_Strt_DtTm TIMESTAMP	MReln_End_DtTm TIMESTAMP

7.2.7 Time summary (table TmSum)

The period over which a measurement can be summarized.

TmSum_Cd CHAR	TmSum_Nm * VARCHAR (120)
H	Hourly
D	Daily
W	Weekly
M	Monthly
Q	Quarterly
Y	Yearly
P	Point
* This column is translated.	

7.2.8 Measurement source (table MSrc)

MSrc_Cd CHAR (6)	MSrc_Parent_Cd CHAR (6)	MSrc_Nm VARCHAR (120)
Tivoli		Tivoli Application
MODEL1		Tivoli Common Data Model V1
AWS	Tivoli	Tivoli Workload Scheduler

7.2.9 Measurement source history (table MSrcHistory)

MSrc_Cd CHAR (6)	**MSrc_Nm VARCHAR (120)**	**MSrc_Strt_DtTm TIMESTAMP**	**MSrc_End_DtTm TIMESTAMP**
Tivoli	Tivoli Application	1970-01-01-00.00.00.000000	9999-01-01-00.00.00.000000
MODEL1	Tivoli Common Data Model V1	1970-01-01-00.00.00.000000	9999-01-01-00.00.00.000000
AWS	Tivoli Workload Scheduler	1970-01-01-00.00.00.000000	9999-01-01-00.00.00.000000

7.2.10 Measurement type (table MsmtTyp)

MsmtTyp_ID INTEGER	**MUnit_Cd CHAR (6)**	**MSrc_Cd CHAR (6)**	**MsmtTyp_Nm * VARCHAR (120)**	**MsmtTyp_Ds * VARCHAR (254)**
1	Min	MODEL1	Available	The amount of time that the resource is available
2	Min	MODEL1	Degrading	The amount of time that the resource is degrading
3	Min	MODEL1	Unavailable	The amount of time that the resource is unavailable
4	Min	MODEL1	Unreachable	The amount of time that the resource is unreachable
5	Min	MODEL1	Responding	The amount of time that it took to detect, isolate, and respond to a problem with the resource
6	Min	MODEL1	Repairing	The amount of time that it took to fix the problem associated with the resource
7	Min	MODEL1	Closing	The amount of time that it took to close the problem with the resource
8	Min	MODEL1	Unknown	The amount of time that the state of the resource is unknown
9	Min	MODEL1	Unmanaged	The amount of time that the resource is unmanaged
10	Msec	MODEL1	Response Time	The amount of time it took a process to respond
20	Sec	MODEL1	Duration	The amount of time needed to complete a process
21	QTY	MODEL1	Number of Unsuccessful Runs	Number of failed runs
22	QTY	MODEL1	Number or Runs	The number of runs
100	QTY	AWS	Number of Unknown	Number of unknown status runs for a job or job stream

MsmtTyp_ID INTEGER	**MUnit_Cd CHAR (6)**	**MSrc_Cd CHAR (6)**	**MsmtTyp_Nm * VARCHAR (120)**	**MsmtTyp_Ds * VARCHAR (254)**
			Runs	
101	QTY	AWS	Number of Successful Runs	Number of runs that were successful for a job or a job stream
102	QTY	AWS	Number of Cancelled Runs	Number of runs that were cancelled for a job or a job stream
103	Sec	AWS	Start Delay	Start delay that is the time elapsed between the planned start time and the effective start time of a job or job stream
104	Sec	AWS	CPU Time	CPU utilization time
105	Min	AWS	Deadline Time	The time within which a job or job stream is considered late in the plan
106	Min	AWS	Late Start Time	The latest time a job or job stream is launched
* This column is translated.				

7.2.11 Component measurement rule (table MsmtRul)

CompTyp_Cd CHAR (17)	**MsmtTyp_ID INTEGER**
SCHED_JOB	20
SCHED_JOB	21
SCHED_JOB	22
SCHED_JOB	100
SCHED_JOB	101
SCHED_JOB	102
SCHED_JOB	103
SCHED_JOB	104
SCHED_JOB	105
SCHED_JOB	106
SCHED_JOBSTREAM	20
SCHED_JOBSTREAM	21
SCHED_JOBSTREAM	22
SCHED_JOBSTREAM	100
SCHED_JOBSTREAM	101
SCHED_JOBSTREAM	102
SCHED_JOBSTREAM	103
SCHED_JOBSTREAM	104

CompTyp_Cd CHAR (17)	MsmtTyp_ID INTEGER
SCHED_JOBSTREAM	105
SCHED_JOBSTREAM	106

7.2.12 Measurement (table Msmt)

Msmt_ID BIGINT	Comp_ID INTEGER	MsmtTyp_ID INTEGER	TmSum_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 INTEGER	Msmt_Err_Cnt INTEGER	msmt_stddev_Val DOUBLE	MSrc_Corr_Cd CHAR (6)
1	13	22	D	2005-09-01	11:45:00				1				AWS
2	13	20	D	2005-09-01	11:45:00			3					AWS
3	13	101	D	2005-09-01	11:45:00				1				AWS
4	13	103	D	2005-09-01	11:45:00	6300	6300	6300					AWS
5	13	104	D	2005-09-01	11:45:00	5903	5903	5903					AWS
6	13	106	D	2005-09-01	11:45:00	105	105	105					AWS
7	12	22	D	2005-09-01	10:25:00				1				AWS

Msmt_ID BIGINT	Comp_ID INTEGER	MsmtTyp_ID INTEGER	TmSum_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 INTEGER	Msmt_Err_Cnt INTEGER	msmt_stddev_Val DOUBLE	MSrc_Corr_Cd CHAR (6)
8	12	20	D	2005-09-01	10:25:00	1	1	1					AWS
9	12	101	D	2005-09-01	10:25:00				1				AWS
10	12	103	D	2005-09-01	10:25:00	5100	5100	5100					AWS
11	12	104	D	2005-09-01	10:25:00	2312	2312	2312					AWS
12	12	106	D	2005-09-01	10:25:00	55	55	55					AWS
13	15	22	D	2005-09-01	20:00:00				2				AWS
14	15	20	D	2005-09-01	20:00:00	1	5	3					AWS
15	15	101	D	2005-09-01	20:00:00				2				AWS
16	15	103	D	2005-09-01	20:00:00	125	125	125					AWS
17	15	104	D	2005-09-01	20:00:00	570	9201	4885.5					AWS

Msmt_ID BIGINT	Comp_ID INTEGER	MsmtTyp_ID INTEGER	TmSum_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 INTEGER	Msmt_Err_Cnt INTEGER	msmt_stddev_Val DOUBLE	MSrc_Corr_Cd CHAR (6)
18	15	105	D	2005-09-01	20:00:00	92	92	92					AWS
19	14	22	D	2005-09-01	20:00:00				2				AWS
20	14	20	D	2005-09-01	20:00:00	1	7	4					AWS
21	14	101	D	2005-09-01	20:00:00				2				AWS
22	14	103	D	2005-09-01	20:00:00	120	120	120					AWS
23	14	104	D	2005-09-01	20:00:00	578	10236	5407					AWS
24	14	105	D	2005-09-01	20:00:00	212	212	212					AWS
25	11	22	D	2005-09-01	10:24:00				1				AWS
26	11	20	D	2005-09-01	10:24:00	5.65	5.65	5.65					AWS
27	11	101	D	2005-09-01	10:24:00				1				AWS

Msmt_ID BIGINT	Comp_ID INTEGER	MsmtTyp_ID INTEGER	TmSum_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 INTEGER	Msmt_Err_Cnt INTEGER	msmt_stddev_Val DOUBLE	MSrc_Corr_Cd CHAR (6)
28	11	103	D	2005-09-01	10:24:00	5040	5040	5040					AWS
29	11	104	D	2005-09-01	10:24:00	8781	8781	8781					AWS
30	11	106	D	2005-09-01	10:24:00	24	24	24					AWS

7.2.13 Threshold measurement objective (table Mobj)

There is no threshold measurement objective for this warehouse pack.

Mobj_ID INTEGER	MsmtTyp_ID INTEGER	CompTyp_Cd CHAR (17)	Centr_Cd CHAR (6)	Cust_ID INTEGER	Attrdom_ID INTEGER	MSrc_Cd CHAR (6)	Mobj_Strt_DtTm TIMESTAMP	Mobj_End_DtTm TIMESTAMP

7.2.14 Threshold measurement objective range (table MobjRng)

There is no threshold measurement objective range for this warehouse pack.

Mobjrng_ID INTEGER	Mobj_ID INTEGER	Sev_Cd CHAR	Mobjrng_Min_Val VARCHAR (254)	Mobjrng_Max_Val VARCHAR (254)	Mobjrng_Strt_Dow TIMESTAMP	Mobjrng_End_Dow TIMESTAMP	Mobjrng_Strt_Tm TIMESTAMP	Mobjrng_End_Tm TIMESTAMP

7.2.15 Threshold severity level (table SevLvl)

There is no threshold severity level for this warehouse pack.

Sev_Cd CHAR	MSrc_Cd CHAR (6)	Sev_Nm * VARCHAR (254)
* This column is translated.		

7.3 Component events

The following section describes the component events.

7.3.1 Event type (table EventTyp)

There is no this type data for this warehouse pack.

EventTyp_ID INTEGER	EventTyp_Nm * VARCHAR (254)	Msrc_Cd CHAR (6)	EventTyp_Ds * VARCHAR (254)
* This column is translated.			

7.3.2 Event (table Event)

There is no component type relationship for this warehouse pack.

Event_ID INTEGER	EventTyp_ID INTEGER	Event_DtTm TIMESTAMP	Tmsum_Cd CHAR (1)	Msrc_Cd CHAR (6)	Repeat_cnt INTEGER	Centr_Cd CHAR (6)	Cust_ID INTEGER	Event_Corr_ID INTEGER	Event_Corr_Val VARCHAR (254)

7.3.3 Event attribute (table EventAttr)

There is no this type data for this warehouse pack.

EventAttr_ID INTEGER	Event_ID INTEGER	EAttrTyp_Cd VARCHAR (254)	Msrc_Cd CHAR (6)	EventAttr_Val VARCHAR (3500)

EventAttr_ID INTEGER	Event_ID INTEGER	EAttrTyp_Cd VARCHAR (254)	Msrc_Cd CHAR (6)	EventAttr_Val VARCHAR (3500)

7.3.4 Event attribute type (table EAttrTyp)

There is no this type data for this warehouse pack.

EattrTyp_Cd VARCHAR (254)	EAttrTyp_Nm * VARCHAR (254)	Msrc_Cd CHAR (6)
* This column is translated.		

7.3.5 Event group (table EGrp)

There is no component type relationship for this warehouse pack.

EGrp_Cd CHAR (6)	EGrpTyp_Cd CHAR (6)	EGrp_Parent_Cd CHAR (6)	EGrp_Nm * VARCHAR (254)
* This column is translated.			

7.3.6 Event group member (table EGrpMbr)

There is no component type relationship for this warehouse pack.

EGrp_Cd CHAR (6)	EGrpTyp_Cd CHAR (6)	EventTyp_ID INTEGER

7.3.7 Event group type (table EGrpTyp)

There is no component type relationship for this warehouse pack.

EGrpTyp_Cd CHAR (6)	EGrpTyp_Nm * VARCHAR (254)
* This column is translated.	

7.3.8 Event type relationship (table ETypReln)

There is no this type data for this warehouse pack.

ETyp_Source_ID INTEGER	ETyp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	Reln_Strt_DtTm TIMESTAMP	Reln_End_DtTm TIMESTAMP

7.3.9 Event relationship (table EventReln)

There is no component type relationship for this warehouse pack.

EventReln_ID INTEGER	Events_Source_ID INTEGER	Event_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	Msrc_Cd CHAR (6)

7.3.10 Component-event relationship (table CEReln)

There is no component type relationship for this warehouse pack.

CEReln_ID INTEGER	Event_ID INTEGER	Comp_ID INTEGER	RelnTyp_Cd CHAR (6)	Msrc_Cd CHAR (6)

7.3.11 Event rule relationship (table ERelnRul)

There is no component type relationship for this warehouse pack.

ETyp_Source_ID INTEGER	ETyp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	Rul_Strt_DtTm TIMESTAMP	Rul_End_DtTm TIMESTAMP

ETyp_Source_ID INTEGER	ETyp_Target_ID INTEGER	RelnTyp_Cd CHAR (6)	Rul_Strt_DtTm TIMESTAMP	Rul_End_DtTm TIMESTAMP

7.3.12 Component-event rule relationship (table CERelnRul)

There is no component type relationship for this warehouse pack.

EventTyp_ID INTEGER	CompTyp_Cd CHAR (17)	RelnTyp_Cd CHAR (6)	Rul_Strt_DtTm TIMESTAMP	Rul_End_DtTm TIMESTAMP

7.4 Helper tables

This warehouse pack does not have helper tables.

7.5 Exception tables

This warehouse pack does not have exception tables.

7.6 Incremental extraction

This warehouse pack uses incremental extraction to extract data from the source tables and to store it into the central data warehouse. It also 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.

Here is an example for this warehouse pack.

EXTCTL_SOURCE	EXTCTL_TARGET	EXTCTL_FROM_INTSEQ	EXTCTL_TO_INTSEQ	EXTCTL_FROM_DTTM	EXTCTL_TO_DTTM
AWS.Stage_TWS_Archive	TWG.MSMT	0	0	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_RUNSTATES_METRIC	AWS.T_RUNSTATES_METRIC	21	102	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_RUNTIMES_METRIC	AWS.T_RUNTIMES_METRIC	20	104	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_CUST	AWS.T_CUST	1	1	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000

EXTCTL_SOURCE	EXTCTL_TARGET	EXTCTL_FROM_INTSEQ	EXTCTL_TO_INTSEQ	EXTCTL_FROM_DTTM	EXTCTL_TO_DTTM
AWS.VD_ENGINE	AWS.T_ENGINE	1	1	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_WORKSTATION	AWS.D_WORKSTATION	3	10	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_JOB	AWS.D_JOB	12	15	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VD_JOBSTREAM	AWS.D_JOBSTREAM	11	14	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VF_RUNSTATES_DAY	AWS.F_RUNSTATES_DAY	25	32	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000
AWS.VF_RUNTIMES_DAY	AWS.F_RUNTIMES_DAY	22	31	2005-09-01-11.45.00.000	2005-09-02-10.32.00.000

8 Data mart schema information

The following sections contain the definition of star schemas, metric dimension tables, data marts, and reports provided with the warehouse pack. This section is intended primarily for report designers and warehouse pack creators. For information about reports, see “Reports” on page 11.

Shaded columns in the following tables are translated. These columns are also marked with an asterisk (*) after the column name.

8.1 Data mart AWS TWH_MART

This data mart uses the following star schemas:

- TWS Daily Run States Star Schema
- TWS Daily Run Times Star Schema

8.2 Star schemas

Before using this section, read about the star schemas in *Enabling an Application for Tivoli Data Warehouse*. That document defines the content of each table and explains the relationships between the tables in this document.

The warehouse pack provides the following star schemas.

8.2.1 TWS Daily Run States Star Schema

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

Description of star schema (in IWH_STARSCHEMA)	TWS Daily Run States Star Schema
Name of fact table	AWS.F_RUNSTATES_DAY
Name of metric dimension table	AWS.D_RUNSTATES_METRIC
Names of other dimension tables	AWS.D_WORKSTATION
	AWS.D_JOB
	AWS.D_JOBSTREAM
	AWS.D_ENGINE
	AWS.D_CUST

8.2.1.1 Fact table AWS.F_RunStates_Day

The following columns are used in the fact table:

- Metric_Id INTEGER
- Workstation_Id INTEGER
- Job_Id INTEGER
- JobStream_Id INTEGER
- Meas_Date TIMESTAMP
- Min_Value DOUBLE
- Max_Value DOUBLE
- Avg_Value DOUBLE
- Total_Value DOUBLE
- Sample_Count DOUBLE
- Fact_Id INTEGER
- Cdw_Id INTEGER

8.2.1.2 Metric dimension tables

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

8.2.1.2.1 Aws.D_RunStates_Metric

Metric_ID INTEGER	Met_Category * VARCHAR (254)	Met_Desc * VARCHAR (254)	Met_Name * VARCHAR (254)	Met_Units * VARCHAR (254)	Min_Exists CHAR (1)	Max_Exists CHAR (1)	Avg_Exists CHAR (1)	Total Exists CHAR (1)	Msrc_Nm * VARCHAR (254)
1	Not used	Number of successful runs for a job or a job stream	Number of Successful Runs	Number of job runs	N	N	N	Y	AWS
2	Not used	Number of runs that failed	Number of Unsuccessful Runs	Number of job runs	N	N	N	Y	MODEL1
3	Not used	Number of cancelled runs for a job or a job stream	Number of Cancelled Runs	Number of job runs	N	N	N	Y	AWS

Metric_ID INTEGER	Met_Category * VARCHAR (254)	Met_Desc * VARCHAR (254)	Met_Name * VARCHAR (254)	Met_Units * VARCHAR (254)	Min_Exists CHAR (1)	Max_Exists CHAR (1)	Avg_Exists CHAR (1)	Total Exists CHAR (1)	Msrc_Nm * VARCHAR (254)
4	Not used	Number of unknown status runs for a job or a job stream	Number of Unknown Runs	Number of job runs	N	N	N	Y	AWS
5	Not used	Total number of runs	Number of Runs	Number of job runs	N	N	N	Y	MODEL1
* This column is translated.									

8.2.2 TWS Daily Run Times Star Schema

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

Description of star schema (in IWH_STARSCHEMA)	TWS Daily Run Times Star Schema
Name of fact table	AWS.F_RUNTIMES_DAY
Name of metric dimension table	AWS.D_RUNTIMES_METRIC
Names of other dimension tables	AWS.D_WORKSTATION
	AWS.D_JOB
	AWS.D_JOBSTREAM
	AWS.D_ENGINE
	AWS.D_CUST

8.2.2.1 Fact table AWS.F_RunTimes_Day

The following columns are used in the fact table:

- Metric_Id INTEGER
- Workstation_Id INTEGER
- Job_Id INTEGER

- JobStream_Id INTEGER
- Meas_Date TIMESTAMP
- Min_Value DOUBLE
- Max_Value DOUBLE
- Avg_Value DOUBLE
- Total_Value DOUBLE
- Sample_Count DOUBLE
- Fact_Id INTEGER
- Cdw_Id INTEGER

8.2.2.2 Metric dimension tables

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

8.2.2.2.1 Aws.D_RunTimes_Metric

Metric_ID INTEGER	Met_Category * VARCHAR (254)	Met_Desc * VARCHAR (254)	Met_Name * VARCHAR (254)	Met_Units * VARCHAR (254)	Min_Exists CHAR (1)	Max_Exists CHAR (1)	Avg_Exists CHAR (1)	Total Exists CHAR (1)	Msrc_Nm * VARCHAR (254)
1	Not used	The amount of time needed to complete the process	Duration	Hours	Y	Y	Y	N	MODEL1
2	Not used	CPU utilization time	CPU Time	Seconds	Y	Y	Y	N	AWS
3	Not used	Start delay that is the time elapsed between the planned start time and the effective start time of a job or job stream	Start Delay	Seconds	Y	Y	Y	N	AWS
4	Not used	The time within which a job or job stream is considered late in the plan	Deadline Time	Minutes	Y	Y	Y	N	AWS
5	Not used	The latest time a job or job stream is launched	Late Start Time	Minutes	Y	Y	Y	N	AWS

* This column is translated.

8.2.3 Dimension tables

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

8.2.3.1 Dimension table Aws.D_Workstation

This dimension is related to the Tivoli Workload Scheduler CPU information.

The following columns are used in this dimension table:

- Workstation_Id INTEGER
- Workstation_Name VARCHAR
- Cust_Id INTEGER
- Engine_Id INTEGER
- OS_Name VARCHAR
- Ip_Host VARCHAR
- Domain_Name VARCHAR
- Host_Name VARCHAR
- Type_Name VARCHAR

8.2.3.2 Dimension table Aws.D_JobStream

This dimension is related to the Tivoli Workload Scheduler job stream information.

The following columns are used in this dimension table:

- Jobstream_Id INTEGER
- Jobstream_Name VARCHAR
- Cust_Id INTEGER
- Engine_Id INTEGER
- Priority_Name VARCHAR

8.2.3.3 Dimension table Aws.D_Job

This dimension is related to the Tivoli Workload Scheduler Job information.

The following columns are used in this dimension table:

- Job_Id INTEGER

- Job_Name VARCHAR
- Cust_Id INTEGER
- Engine_Id INTEGER
- Login_Name VARCHAR
- Command_Name VARCHAR
- Priority_Name VARCHAR

8.2.3.4 Dimension table Aws.D_Engine

This dimension is related to the engine, Tivoli Workload Scheduler Master, information.

The following columns are used in this dimension table:

- Engine_Id INTEGER
- Engine_Name VARCHAR
- Cust_Id INTEGER
- Domain_Name VARCHAR

8.2.3.5 Dimension table Aws.D_Cust

This dimension is related to the customer information.

The following columns are used in this dimension table:

- Cust_Id INTEGER
- Cust_Parent_Id INTEGER
- GeoArea_Cd VARCHAR
- TmZone_Id INTEGER
- Cust_Acct_Cd VARCHAR
- Cust_Nm VARCHAR

9 Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

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

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758   U.S.A.

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

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

AIX, DB2, IBM, the IBM logo, Tivoli and the Tivoli logo are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Intel, the Intel Inside logos, MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

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

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

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

Program Number: 5698-WSH

Printed in U.S.A.

SC32-2259-00