

# **PATROL<sup>®</sup> for Tivoli<sup>®</sup> Workload Scheduler by OTL Software**



## **User Guide**

**Version 1.4  
Document Revision 1  
May 17 2011**



Copyright ©2011 OTL Software Limited, as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. Tivoli Workload Scheduler is a registered trademark of International Business Machines Corporation or Tivoli Systems Inc.. All other third party logos and product/trade names are registered trademarks or trademarks of their respective companies.

THE USE AND CONTENTS OF THIS DOCUMENTATION ARE GOVERNED BY THE SOFTWARE LICENSE AGREEMENT FOUND AT [http://www.otl.co.nz/km/OTL\\_KM\\_Software\\_Licence\\_Agreement.pdf](http://www.otl.co.nz/km/OTL_KM_Software_Licence_Agreement.pdf).

## Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b), or DFARS 227.7202, as amended from time to time. Send any contract notices to OTL Software Limited.

---

## Contacting OTL Software

<b>Postal Address</b>	OTL Software Limited P O Box 68300 Newton Auckland 1145 New Zealand	<b>Web Site Address</b>	<a href="http://www.otl.co.nz">http://www.otl.co.nz</a>
<b>Telephone</b>	+64 9 373 9920	<b>E-mail (Sales)</b>	<a href="mailto:sales@otl.co.nz">sales@otl.co.nz</a>
<b>Fax</b>	+64 9 303 9129	<b>E-mail (Help Desk)</b>	<a href="mailto:helpdesk@otl.co.nz">helpdesk@otl.co.nz</a>
<b>Help Desk</b>	+64 9 303 9120		



# Customer Support

You can obtain technical support by using the Support page on the OTL Software web site or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting OTL Software,” below.

## Support Web Site

You can obtain technical support from OTL Software 24 hours a day, seven days a week by accessing the technical support web site at <http://www.otl.co.nz/support>. From this site, you can:

- read overview about support services and programs that OTL Software offers
- log a support call online
- check OTL Software contact information, including e-mail addresses, fax numbers, and telephone numbers

Click [here](#) to ensure you have the latest version of the OTL Software KMs.

## Support via Telephone or E-mail

If you need technical support, you can contact OTL Software by calling +64 9 303 9120 or by faxing +64 9 303 9129. You can also contact the OTL Software Helpdesk via e-mail at [helpdesk@otl.co.nz](mailto:helpdesk@otl.co.nz). For other contact methods, please refer to the OTL Software Web site at <http://www.otl.co.nz/support>.

## Before Contacting OTL Software

Before you contact OTL Software, please have the following information available so that a technical support analyst can begin working on your problem immediately:

- KM product information:
  - product name
  - product version
  - license serial number
- monitored application information:
  - Tivoli Workload Scheduler version
- operating system information:
  - machine type
  - operating system type, version, and service pack or patch details
  - system hardware configuration
- PATROL information:
  - PATROL Agent version
  - PATROL Console version and platform details
  - BMC ProactiveNet Performance Management Portal version and platform details
- sequence of events leading to the problem
- commands and options needed to reproduce the problem
- messages received:
  - product error messages
  - messages from monitored application
  - messages in PATROL Console system output window (SOW)



---

## Contents

<b>1</b>	<b>Introduction</b>	
	PATROL for Tivoli Workload Scheduler .....	1-2
	Features .....	1-2
	Supported Operating Systems .....	1-3
	Supported Versions .....	1-4
	Security Requirements .....	1-4
	Disk and Memory Usage .....	1-5
	Components .....	1-6
	Applications and Icons .....	1-8
	Hierarchical Structure .....	1-11
	Instance Naming .....	1-17
	InfoBoxes .....	1-20
	Where to Go from Here .....	1-25
<b>2</b>	<b>Getting Started</b>	
	Preparing to Use TWS KM .....	2-2
	KM Requirements .....	2-2
	License requirements for the KM .....	2-2
	Software Requirements for the KM .....	2-2
	Installing the KM .....	2-3
	Installing the KM (Using BMC Installation Utility) .....	2-3
	Preparing to Install or Upgrade (Using All in One Exe/Zip File) .....	2-4
	Installing the KM on a Unix Platform .....	2-5
	Installing the KM on a Microsoft Windows Platform .....	2-6
	Installing or Upgrading the PAR file on BPPM Portal .....	2-7
	Loading the KM .....	2-8
	Preparing to Load the KM .....	2-8
	Loading the KM on PATROL Console .....	2-8

Loading the KM on PATROL Central .....	2-9
Loading the KM on BPPM Portal .....	2-10
Configuring the KM .....	2-11
Licensing the KM .....	2-12
Agent and Console Re-start .....	2-12
Discovery Cycle .....	2-13
Help .....	2-13
Accessing Help .....	2-13
Where to Go from Here .....	2-14

### **3 Menu Summary**

Accessing Application Menus .....	3-2
Menu Summary .....	3-3
TWS_MAESTRO Application Menu .....	3-4
TWS_CPU_CONT Application Menu .....	3-9
TWS_CPU Application Menu .....	3-10
TWS_PROMPTS_CONT Application Menu .....	3-12
TWS_PROMPTS Application Menu .....	3-13
TWS_SCHEDULES_CONT Application Menu .....	3-14
TWS_SCHEDULES_CPU Application Menu .....	3-15
TWS_SCHEDULES Application Menu .....	3-16
TWS_JOBS_CONT Application Menu .....	3-18
TWS_JOBS Application Menu .....	3-19
Where to Go from Here .....	3-21

### **4 Parameter Summary**

Functional Parameter Summary .....	4-2
Parameter Default Values .....	4-8
Where to Go from Here .....	4-12

### **5 Monitoring TWS**

Overview .....	5-2
Objectives of the KM .....	5-2
Monitoring Server Availability .....	5-2
Monitoring CPU Availability .....	5-3
Schedule, Job, and Prompt Filters .....	5-3
Schedule Filter .....	5-4
Job Filter .....	5-5
Prompt Filter .....	5-6
Administering TWS KM .....	5-7

Starting the NETMAN Process . . . . .	5-7
Stopping the Process Tree . . . . .	5-8
Starting the Process Tree . . . . .	5-9
Stopping the Production Processes . . . . .	5-10
Launching Administration Utilities . . . . .	5-11
Reporting from TWS KM . . . . .	5-12
Tivoli Workload Scheduler Database Validation . . . . .	5-13
Debugging TWS KM . . . . .	5-14
Before You Begin . . . . .	5-14
To Enable or Disable Debugging . . . . .	5-14
Refreshing Parameters . . . . .	5-15
Before You Begin . . . . .	5-15
To Refresh All Parameters . . . . .	5-15
Displaying a Parameter Graph, Gauge, or Text Output Window . . .	5-16
Customizing Parameters . . . . .	5-16
Unloading the KM . . . . .	5-17
Unloading the KM from PATROL Agent . . . . .	5-17
Unloading the KM from PATROL Console . . . . .	5-18
Unloading the KM from PATROL Central Console . . . . .	5-19
Uninstalling the KM . . . . .	5-20
Uninstalling the KM from PATROL Agent . . . . .	5-20
Uninstalling the KM from PATROL Console . . . . .	5-21
Uninstalling the KM from PATROL Central Console . . . . .	5-23
Uninstalling the KM from PATROL Central Console Server . . .	5-23
Uninstalling the KM from PATROL Central Web Server . . . . .	5-24
Uninstalling the PAR File from the BPPM Portal . . . . .	5-24
Deleting PATROL Agent Configuration Variables . . . . .	5-25
Where to Go from Here . . . . .	5-25

## Index





---

## Figures

Figure 1-1	PATROL for Tivoli Workload Scheduler Application Icons . . .	1-7
Figure 1-2	TWS KM Top Level Application Classes . . . . .	1-11
Figure 1-3	TWS KM TWS_PROMPTS Application Class Hierarchy . . . . .	1-12
Figure 1-4	TWS KM TWS_PROCESS Application Class Hierarchy . . . . .	1-13
Figure 1-5	TWS KM TWS_CPU Application Class Hierarchy . . . . .	1-14
Figure 1-6	TWS KM TWS_SUMMARY Application Class Hierarchy . . .	1-15
Figure 1-7	TWS KM TWS_SCHEDULES Application Class Hierarchy . .	1-16
Figure 2-1	TWS_SETUP Icon . . . . .	2-9
Figure 2-2	TWS_SETUP Icon . . . . .	2-11
Figure 2-3	License TWS KM Menu . . . . .	2-12
Figure 3-1	Application Menus . . . . .	3-2
Figure 3-2	TWS_CPU_CONT Application Menu . . . . .	3-9
Figure 3-3	TWS_CPU Application Menu . . . . .	3-10
Figure 3-4	TWS_PROMPTS_CONT Application Menu . . . . .	3-12
Figure 3-5	TWS_PROMPTS Application Menu . . . . .	3-13
Figure 3-6	TWS_SCHEDULES_CONT Application Menu . . . . .	3-14
Figure 3-7	TWS_SCHEDULES_CPU Application Menu . . . . .	3-15
Figure 3-8	TWS_SCHEDULES Application Menu . . . . .	3-16
Figure 3-9	TWS_JOBS_CONT Application Menu . . . . .	3-18
Figure 3-10	TWS_JOBS Application Menu . . . . .	3-19
Figure 5-1	Schedule Filter Window . . . . .	5-4
Figure 5-2	Job Filter Window . . . . .	5-5
Figure 5-3	Prompts Filter Window . . . . .	5-6



---

## Tables

Table 1-1	PATROL for Tivoli Workload Scheduler Components . . . . .	1-6
Table 1-2	TWS KM Icons, Names, Application Classes, and Descriptions . . . . .	1-8
Table 1-3	TWS_SERVER Application InfoBox Items . . . . .	1-20
Table 1-4	TWS_CPU Application InfoBox Items . . . . .	1-21
Table 1-5	TWS_PROMPTS Application InfoBox Items . . . . .	1-22
Table 1-6	TWS_SCHEDULES Application InfoBox Items . . . . .	1-23
Table 1-7	TWS_JOBS Application InfoBox Items . . . . .	1-24
Table 2-2	Contents of the Distribution File . . . . .	2-4
Table 2-3	Unix Platform Installation Files and Extraction Paths. . . . .	2-5
Table 2-4	MS Windows Platform Installation Files and Extraction Paths. . . . .	2-6
Table 3-1	Menu Items for TWS_MAESTRO Application . . . . .	3-4
Table 3-2	Menu Items for TWS_PROMPTS_CONT Application . . . . .	3-9
Table 3-3	Menu Items for TWS_CPU Application . . . . .	3-11
Table 3-4	Menu Items for TWS_PROMPTS_CONT Application . . . . .	3-12
Table 3-5	Menu Items for TWS_PROMPTS Application . . . . .	3-13
Table 3-6	Menu Items for TWS_SCHEDULES_CONT Application . . . . .	3-14
Table 3-7	Menu Items for TWS_SCHEDULES_CPU Application . . . . .	3-15
Table 3-8	Menu Items for TWS_SCHEDULES Application . . . . .	3-16
Table 3-9	Menu Items for TWS_JOBS_CONT Application . . . . .	3-18
Table 3-10	Menu Items for TWS_JOBS Application . . . . .	3-19
Table 4-1	PATROL for Tivoli Workload Scheduler Parameter Summary . . . . .	4-2
Table 4-2	TWS KM Parameter Default Values . . . . .	4-9
Table 5-1	Uninstallation from the PATROL Agent . . . . .	5-21
Table 5-2	Uninstallation from PATROL Console . . . . .	5-22



---

## About this Guide

PATROL<sup>®</sup> for Tivoli<sup>™</sup> Workload Scheduler by OTL Software (TWS KM) User Guide contains detailed information about the applications, commands, and parameters that the TWS KM provides. The guide also contains instructions for loading and configuring the Knowledge Module (KM). For more detailed information, refer to the TWS KM online help.

This guide should be used with the appropriate PATROL user guide for your Console, which describes how to use PATROL to perform typical tasks.

This chapter discusses the following topics:

Who Should Read This Guide . . . . .	xiv
How This Guide Is Organised . . . . .	xiv
Related Publications . . . . .	xv
Documentation Sequence . . . . .	xviii
Where to Look for Information . . . . .	xxi
When Used with the PATROL Console for Unix . . . . .	xix
When Used with the PATROL Console for Windows . . . . .	xx
Where to Look for Information . . . . .	xxi
Conventions . . . . .	xxiii
Mouse Controls . . . . .	xxiv

## Who Should Read This Guide

This guide is intended for backup administrators, system administrators, and anyone who monitors backup systems. This guide assumes that you are familiar with your host operating system and TWS. You should know how to perform a basic set of actions in a window environment, including

- choosing menu commands
- moving and resizing windows
- opening icon windows
- dragging and dropping icons
- using mouse controls for your system

## How This Guide Is Organised

This manual is organized as follows:

<b>Chapter</b>	<b>Title</b>	<b>Purpose</b>
1	"Introduction"	provides an overview of the features and components of the KM.
2	"Getting Started"	provides information on setting up and accessing the KM and provides basic information about the KM.
3	"Menu Summary"	discusses the menus that the KM offers.
4	"Parameter Summary"	discusses the parameters that the KM offers.
5	"Monitoring TWS"	provides tasks that you perform using the KM.
Index	"Index"	lists index entries.

## Related Publications

PATROL product documentation consists of both hardcopy and online publications. PATROL hardcopy documentation is divided into the following categories based on function:

Category	Document	Purpose
PATROL Base Documents	PATROL for UNIX Getting Started	provides procedures and examples to introduce PATROL Console for Unix.
	PATROL for Agent Reference Manual	describes the PATROL Agent and explains how it interacts with other PATROL components. It also describes configuration utilities and Management Information Base (MIB) tables used with the Agent.
	PATROL for Unix User Guide	contains task-oriented information on how to fill out appropriate dialog boxes to manage the computers, applications, and parameters that PATROL is capable of managing using the PATROL Console for UNIX.
	PATROL for Windows User Guide (Volume 1)	introduces you to PATROL components, object hierarchy, and the GUI using the PATROL Console for Windows. It also contains task-oriented information about how to start PATROL components, load KMs, and discover applications.
	PATROL for Windows User Guide (Volume 2)	contains the task-oriented information about how to monitor and manage computers, applications, and parameters using the PATROL Console for Windows.
	PATROL for Windows User Guide (Volume 3)	describes how to customise your PATROL monitoring environment using the PATROL Console for Windows.
	PATROL Command Line Interfaces Reference Manual	describes the PATROL command line interfaces for the PATROL Agent and the PATROL Console.
	PATROL Console Charting Server for Unix Reference Manual	describes how you can collect and plot system and application data in a real-time chart or graph.
PATROL Installation Documents	PATROL Installation Guides	describe how to run the installation program to load the platform-specific PATROL Agents, PATROL Consoles, and PATROL KMs.

<b>Category</b>	<b>Document</b>	<b>Purpose</b>
PATROL Integration Documents	PATROLVIEW user guides	describe the PATROLVIEW products. PATROLVIEW allows you to fully integrate PATROL with leading enterprise management products.
	PATROLINK for CA-Unicenter Reference Manual	provides information about installing and configuring the PATROLINK product for your particular site. PATROLINK allows you to connect to PATROL from the CA-Unicenter console.
PATROL Event Manager (PEM) Documents	PATROL Event Manager Console for Unix User Guide	describes the stand-alone Event Manager Console for Unix provided with the PATROL product. The PEM Console is a graphical user interface that allows you to manage the events generated by PATROL as it monitors your applications.
	PATROLWATCH for Web Browsers User Guide	provides the ability to view PATROL monitored hosts and applications using the Internet and platform-specific browsing technology.
	PATROLWATCH for Windows User Guide	describes the standalone event manager for Windows.
PATROL Knowledge Module (KM) Documents	Specific PATROL Knowledge Module user guides	contain task-oriented information for loading and modifying individual PATROL KMs used in monitoring and managing operating systems, databases, Knowledge Modules, and applications.



Category	Document	Purpose
PATROL Software Development Kit (SDK) Documents	PATROL Script Language Reference Manual	describes the PATROL Script Language (PSL) data types, syntax, operators, statements, and built-in functions.
	PATROL Script Language Debugger for Unix Reference Manual	discusses the PSL debugger available through the PATROL Developer Console for Unix. The PSL debugger provides an interactive GUI environment for debugging PSL processes and scripts in the PATROL Agent.
	PATROL Online Help Developers Guide	provides guidelines and procedures for implementing a BMC Software Help File. The PATROL <i>Online Help Developers Guide</i> includes elements of style, design, and presentation.
	PATROL Knowledge Module Developers Style Guide	presents the objectives, methods and requirements of PATROL Knowledge Module development and includes these topics: <ul style="list-style-type: none"> <li>• KM Style</li> <li>• setup application</li> <li>• packaging and structure</li> <li>• efficiency and usage</li> </ul>
	PATROL API Reference Manual	describes the PATROL API, a series of functions defined in a C header file that allow a user-written non-PATROL program to connect to PATROL or read a PATROL event log circular file.
Utility Document	PATROL KM Migrator User Guide	describes how you can incorporate your KM customisations into the current version.
Supplemental Documents	Release Notes and Technical Bulletins	explain the latest updates to PATROL products.

These hardcopy publications can be requested from BMC Software, Inc., or can be viewed on BMC Software's Internet World Wide Web site (<http://www.bmc.com/>) when you have registered for Customer Support. Each PATROL Console and each KM come with an extensive online help facility that is available through the PATROL Console **Help** menu option. The online documentation contains reference information about PATROL Console features and options and about KM parameters.

# Documentation Sequence

The following tables provide the suggested sequence for using PATROL documentation. An asterisk denotes additional documentation that may be applicable to your job function.

# When Used with the PATROL Console for Unix

If you work as a...	then read these documents in the order shown:											
	PATROL Installation Guide - Specific	PATROL for Unix Getting Started	PATROL Agent Reference Manual	PATROL for Unix User Guide	PATROL Command Line Reference Manual	PATROL Charting Server Reference Manual	PATROL KM User Guide(s) - Specific	PATROL API Reference Manual	PATROL PSL Reference Manual	PATROL KM Developer's Style Guide	PATROLVIEW™ Guide(s) - Specific	PATROLWATCH™ Guides
<b>Project Engineer</b> - responsible for implementing PATROL and rollout	1	2	3	4							5	6
<b>Systems Administrator/Network Manager</b> - responsible for administering Unix or other operating systems and networks	1	2	3	4	5	6	7				8	9
<b>Database Administrator</b> - responsible for monitoring and administering databases		1		2			3				4	5
<b>Operator</b> - responsible for monitoring environments	1	2	3	4			5				6	7
<b>Help Desk Personnel</b> - responsible for troubleshooting user problems		1		2			3				4	5
<b>Applications Programmer/Developer</b> - responsible for developing KMs		1		2	3	4	5	6	7	8	9	10

# When Used with the PATROL Console for Windows

If you work as a...	then read these documents in the order shown:											
	PATROL Installation Guide - Specific	PATROL for Unix Getting Started	PATROL Agent Reference Manual	PATROL for Unix User Guide	PATROL Command Line Reference Manual	PATROL Charting Server Reference Manual	PATROL KM User Guide(s) - Specific	PATROL API Reference Manual	PATROL PSL Reference Manual	PATROL KM Developer's Style Guide	PATROLVIEW™ Guide(s) - Specific	PATROLWATCH™ Guides
<b>Project Engineer</b> - responsible for implementing PATROL and rollout	1	2	3	4							5	6
<b>Systems Administrator/Network Manager</b> - responsible for administering Unix or other operating systems and networks	1	2	3	4	5	6	7				8	9
<b>Database Administrator</b> - responsible for monitoring and administering databases		1		2			3				4	5
<b>Operator</b> - responsible for monitoring environments	1	2	3	4			5				6	7
<b>Help Desk Personnel</b> - responsible for troubleshooting user problems		1		2			3				4	5
<b>Applications Programmer/Developer</b> - responsible for developing KMs		1		2	3	4	5	6	7	8	9	10

# Where to Look for Information

The following table summarizes where to look for more information on using PATROL, Knowledge Modules, and PATROL integration products to perform typical tasks.

<b>If you want information about...</b>	<b>See the...</b>
adding computers to Patrol	<i>PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 1)</i>
changing the behavior of the PATROL console or the PATROL Agent by using a script or operating system command line	<i>PATROL Command Line Interfaces Reference Manual</i>
changing the PATROL Agent configuration	<i>PATROL Agent Reference Manual</i>
changing various parameters in a real-time environment	<i>PATROL Console Charting Server Reference Manual or the PATROL for Windows User Guide (Volume 2)</i>
connecting to PATROL from a network manager	<i>PATROLVIEW user guides and the PATROLINK for CA-Unicenter Reference Manual</i>
defining your monitoring environment	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1)</i>
KMs in general	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1)</i>
KM versioning and customizations	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3)</i>
managing monitored objects	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 2)</i>
specific applications	<i>appropriate Knowledge Module's user guide and online help</i>
specific menu commands	<i>appropriate Knowledge Module's user guide and online help</i>
specific parameters	<i>appropriate Knowledge Module's user guide and online help</i>
starting and stopping the PATROL Console	<i>PATROL installation guides, PATROL for Unix Getting Started, and the PATROL Windows User Guide (Volume 1)</i>
starting and stopping the PATROL Agent	<i>PATROL installation guides, PATROL for Unix Getting Started, and the PATROL Windows User Guide (Volume 1)</i>

<b>If you want information about...</b>	<b>See the...</b>
managing events	<i>PATROL for Unix User Guide, the PATROL Event Manager Console for Unix User Guide, or the PATROL for Windows User Guide (Volume 2)</i>
the PATROL interface	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1)</i>
the PATROL Script Language (PSL)	<i>PATROL Script Language Reference Manual</i>
working with menu commands	<i>PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 2)</i>
working with parameters	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 2)</i>
working with tasks	<i>PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 2)</i>
unloading the KM	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1)</i>
customizing commands (PATROL Developer Console required)	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3)</i>
customizing a computer class (PATROL Developer Console required)	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3)</i>
customizing an InfoBox (PATROL Developer Console required)	<i>PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 3)</i>
defining an application (PATROL Developer Console required)	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3)</i>
defining a parameter (PATROL Developer Console required)	<i>PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3)</i>
PSL commands and writing PSL scripts (PATROL Developer Console required)	<i>PATROL Script Language Reference Manual</i>
debugging your PSL scripts (PATROL Developer Console required)	<i>PATROL Script Language Debugger for Unix Reference Manual or the PATROL for Windows User Guide (Volume 2)</i>

# Conventions

This guide contains detailed procedures about using the PATROL<sup>®</sup> for Tivoli<sup>™</sup> Workload Scheduler by OTL Software with the PATROL Console for Unix and the PATROL Console for Windows. When instructions for the two Consoles differ, you'll see a heading such as “**With the PATROL Console for Unix**” or “**With the PATROL Console for Windows**”.

The following special elements have been used in this guide to make it easier for you to use:

---

## Note

---

Notes provide additional information about the current subject.

---

---

## Warning

---

Warnings alert you to situations that can cause problems, such as the loss of data, if you do not follow the instructions carefully.

---

All syntax, operating system terms, and literal examples are presented in this font.

*Italics in a command string* signify variables.

Text enclosed in angle brackets (< >) denotes variable information. Replace the variable information with the information it represents.

The word *choose* is used in instruction text in the context of carrying out a series of menu choices to invoke some function. For example, “Choose **File => Save.**”

In hardcopy documents, the symbol >> denotes one-step instructions.

# Mouse Controls

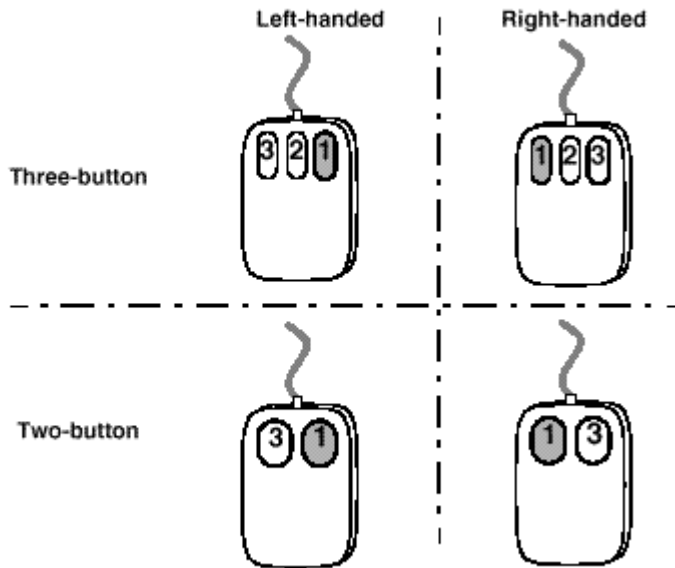
Please note the function of the mouse buttons in all PATROL windows using:

Unix		Windows		Function
Button	Action	Button	Action	
MB1	Click ... Double-Click ...	Left mouse button	Click ... Double-click ...	Selects an icon, menu command, or button; opens an icon's container.
MB2	Using MB2, click ...	-	-	Displays an icon's InfoBox.
MB3	Using MB3, click ...	Right mouse button	Right-click the ...	Displays an icon's pop-up menu.

## With the PATROL Console for Unix

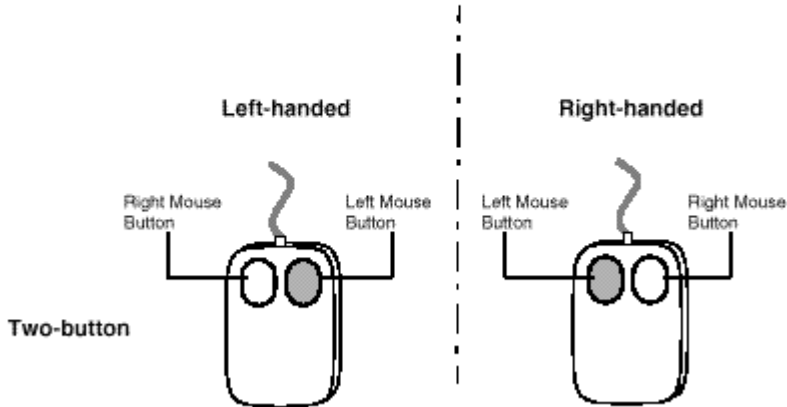
The following figure shows the names and positions of the buttons on right- and left-handed three- and two-button mouse devices when used with the PATROL Console for Unix. MB2 is simulated on a two-button mouse by simultaneously pressing the two buttons (MB1 and MB3).





### With the PATROL Console for Windows

The following figure shows the names and positions of the buttons on right- and left-handed two-button mouse devices when used with the PATROL Console for Windows.



### **With any PATROL Console**

One-button mouse devices such as those used by Apple Macintosh assign MB1 (or left mouse button) to the single mouse button and use a user-selectable combination of option and arrow keys to simulate MB2 and MB3 (or right mouse button). Refer to the documentation for the Macintosh X Window emulation software for details.

---

# Introduction

This chapter provides you with a brief overview of PATROL for Tivoli Workload Scheduler by OTL Software (also referred to as TWS KM). PATROL for Tivoli Workload Scheduler is also known as TWS, and formerly Unison Maestro. The following topics are discussed:

PATROL for Tivoli Workload Scheduler . . . . .	1-2
Features . . . . .	1-2
Supported Operating Systems . . . . .	1-3
Supported Versions . . . . .	1-4
Security Requirements . . . . .	1-4
Disk and Memory Usage . . . . .	1-5
Components . . . . .	1-6
Applications and Icons . . . . .	1-8
Hierarchical Structure . . . . .	1-11
Instance Naming . . . . .	1-17
InfoBoxes . . . . .	1-20
Where to Go from Here . . . . .	1-25

# PATROL for Tivoli Workload Scheduler

A Knowledge Module is a set of files that contain knowledge in the form of command descriptions, application, parameters, and recovery actions that PATROL can use to monitor Tivoli Workload Scheduler.

TWS KM parameters allow you to analyze TWS schedule and job status information quickly and easily to provide management of the production scheduling environment. You can clearly identify problems relating to jobs and schedules and view prompts awaiting a response.

By enabling you to detect problems, optimize systems, analyze trends, plan capacity, and manage multiple hosts simultaneously, TWS KM helps you ensure that your Tivoli Workload Scheduler installation runs efficiently 24 hours a day.

## Features

Key features of TWS KM include:

- Monitoring the system processes responsible for running Tivoli Workload Scheduler
- Monitoring status of the Master, Domain Managers, Fault Tolerant Agents, Standard Agents, and Extended Agent to ensure that they are linked and available
- Monitoring of outstanding prompts
- Provide statistics on the number of schedules in each of the states such as READY, HOLD, SUCC, etc
- Provide statistics on the number of jobs in each of the states such as READY, HOLD, SUCC, etc
- Monitor schedules that match the filter criteria
- Monitor jobs that match the filter criteria

- Ability to control schedules and jobs
- Launch X-Windows administration tools (UNIX only)
- Ability to validate database configurations
- Reports of production schedules

## Supported Operating Systems

PATROL for Tivoli Workload Scheduler can be run on the following operating systems:

- Unix - Solaris, HP-UX, Digital Unix, AIX, Linux
- Windows

TWS KM may run on other variants of Unix, on which Tivoli Workload Scheduler runs, but has not been tested.

On Windows platforms the Tivoli Workload Scheduler installation is identified by checking the Windows registry. Depending on the version of Tivoli Workload Scheduler, the registry entries can be found under one of the following registry paths:

*“HKEY\_LOCAL\_MACHINE\SOFTWARE\Unison Software, Inc.\Unison Maestro\”*

*“HKEY\_LOCAL\_MACHINE\SOFTWARE\IBM\Tivoli Workload Scheduler”*

*“HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\IBM\Tivoli Workload Scheduler”.*

Therefore the PATROL Agent user should have read access to the registry path.

---

### Note

64-bit installation of Tivoli Workload Scheduler may not be detected through the 32-bit PATROL Agent, unless the above registry path is copied under:

*HKEY\_LOCAL\_MACHINE\SOFTWARE\WOW6432NODE\.*

---

## Supported Versions

PATROL for Tivoli Workload Scheduler supports the following versions for PATROL and Tivoli Workload Scheduler:

### **Tivoli Workload Scheduler:**

- version 5.2 and above running on Solaris, HP-UX, Digital Unix, AIX, Linux or Windows platforms.

### **BMC PATROL Agent:**

- version 3.4.20 and above.

### **BMC PATROL Console:**

- version 3.4 and above.

### **BMC ProactiveNet Performance Management Portal:**

- version 2.5 and above.

## Security Requirements

PATROL for Tivoli Workload Scheduler does not require any specific security rights. The only base requirement is that the PATROL account should be able to run the **conman** and **cpuinfo** (UNIX only) commands to be able to view CPU, Schedule, Job, Prompt and TWS application status details. If for some reason the **Security** file in TWS has been configured to restrict access, then the PATROL account may need to be added to the file, to have display access to most components. Please refer the *PATROL for Tivoli Workload Scheduler Planning and Installation Guide*.

Specific menus do require the operators to enter the login and password of a valid user to perform TWS Operations.

## Disk and Memory Usage

When monitoring a standard installation of TWS using TWS KM, the PATROL Agent will consume approximately 800KB of additional system memory. A PATROL for Tivoli Workload Scheduler master machine configured with additional CPU's, or with filters for prompts, schedules, or jobs that create more instances, this will consume additional memory resources as per other KMs used by the PATROL Agent.

When monitoring a standard installation of TWS using TWS KM, the PATROL Agent will generate approximately 200K of history data per day. An installation configured with additional CPU's or modified filter rules will generate more history data as per other KMs used by the PATROL Agent.

---

### Note

---

The number of monitored TWS component instances can be reduced using the **Set Number of Instances** menu.

---

# Components

The TWS KM consists of the application classes described in Table 1-1:

**Table 1-1 PATROL for Tivoli Workload Scheduler Components**

<b>Application Class</b>	<b>Description</b>	<b>Parent/Child Relationship</b>
TWS_MAESTRO	displays the Maestro information	None (Top level)
TWS_SETUP	used to license the KM	None (Top level)
TWS_CPU_CONT	container for CPU information	Child to TWS_MAESTRO
TWS_CPU	displays CPU information	Child to TWS_MAESTRO
TWS_PROCESS_CONT	container for process information	Child to TWS_MAESTRO
TWS_PROCESS	displays information about key Maestro processes	Child to TWS_PROCESS_CONT
TWS_PROMPTS_CONT	container for prompt information	Child to TWS_MAESTRO
TWS_PROMPTS_CPU	container for prompts related to a specific cpu	Child to TWS_PROMPTS_CONT
TWS_PROMPTS	displays prompt information	Child to TWS_PROMPTS_CPU
TWS_SUMMARY	displays summary information about schedules and jobs	Child to TWS_MAESTRO or TWS_SCHEDULES_CPU
TWS_SCHEDULES_CONT	container for schedule and job details	Child to TWS_MAESTRO
TWS_SCHEDULES_CPU	container for schedules and jobs related to a specific cpu	Child to TWS_SCHEDULES_CONT
TWS_SCHEDULES	displays information about schedule	Child to TWS_SCHEDULES_CPU
TWS_SCHEDULES_STATUS	displays current status of the schedule	Child to TWS_SCHEDULES
TWS_JOBS_CONT	container for jobs	Child to TWS_SCHEDULES
TWS_JOBS	displays information about the job	Child to TWS_JOBS_CONT
TWS_JOBS_STATUS	displays current status of the job	Child to TWS_JOBS



Figure 1-1 displays the icons for each successfully configured TWS KM application class.

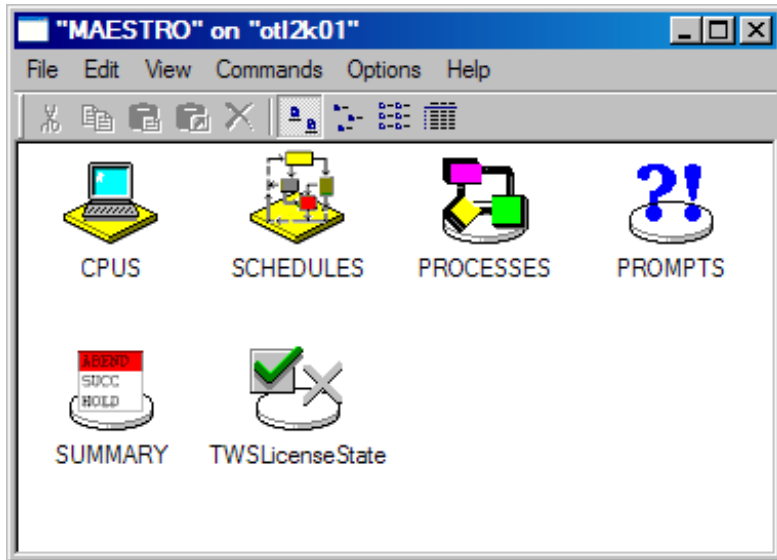







Figure 1-1 PATROL for Tivoli Workload Scheduler Application Icons






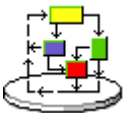
## Applications and Icons

Table 1-2 contains information on each application in the TWS KM. For information on parameter icons, refer to the *PATROL for Unix User Guide* or the *PATROL for Windows User Guide (Volume 2)*.


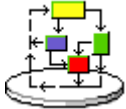




**Table 1-2 TWS KM Icons, Names, Application Classes, and Descriptions**

Icon and Name	Application Class	Description
 Tivoli Workload Scheduler	TWS_MAESTRO	Represents the TWS Master CPU
 TWS_SETUP	TWS_MAESTRO	Application class to configure licensing details for PATROL for Tivoli Workload Scheduler
 TWS_CPU_CONT	TWS_CPU_CONT	Represents container holding CPU information
 TWS_CPU	TWS_CPU	Represents CPU's configured within Maestro
 TWS_PROCESS_CONT	TWS_PROCESS_CONT	Represents container holding process information

**Table 1-2 TWS KM Icons, Names, Application Classes, and Descriptions**

Icon and Name	Application Class	Description
 TWS_PROCESS	TWS_PROCESS	Represents process being monitored
 TWS_PROMPTS_CONT	TWS_PROMPTS_CONT	Represents the container for holding prompts meeting the configured prompts filter
 TWS_PROMPTS_CPU	TWS_PROMPTS_CPU	Represents the cpu that the prompts belong to
 TWS_PROMPTS	TWS_PROMPTS	Represents a prompt matching the prompts filter configured. State can be one of ASKED, INACT, NO, or YES, depending on the filter.
 TWS_SUMMARY	TWS_SUMMARY	Represents the summary details about schedules and jobs
 TWS_SCHEDULES_CONT	TWS_SCHEDULES_CONT	Represents the container holding all schedules

**Table 1-2 TWS KM Icons, Names, Application Classes, and Descriptions**

Icon and Name	Application Class	Description
 <p>TWS_SCHEDULES_CPU</p>	TWS_SCHEDULES_CPU	Represents CPU that has schedules matching the filter rules
 <p>TWS_SCHEDULES</p>	TWS_SCHEDULES	Represents a specific schedule
 <p>TWS_SCHEDULES_STATUS</p>	TWS_SCHEDULES_STATUS	Represents the status of the current schedule
 <p>TWS_JOBS_CONT</p>	TWS_JOBS_CONT	Represents the container holding jobs
 <p>TWS_JOBS</p>	TWS_JOBS	Represents a specific job
 <p>TWS_JOBS_STATUS</p>	TWS_JOBS_STATUS	Represents the status of the current job

# Hierarchical Structure

PATROL for Tivoli Workload Scheduler is organized as groups of application classes. Figure 1-2 shows the top level icons from Table 1-2 in a graphical representation of TWS KM hierarchical structure. Subsequent figures represent the application icons and parameters at lower levels.

---

**Note**

The top level icon for TWS KM is a single instance of the TWS\_MAESTRO application class, labeled Tivoli Workload Scheduler. Only one installed version of TWS will be monitored. Automatic discovery will initially detect the currently active version of TWS.

---

By double-clicking an application class icon, you will find the parameters that monitor your TWS system. Instead of parameters, some application classes may contain additional application classes, application instances, or both. For example, for the TWS\_SCHEDULES application class, you will find one icon for each discovered schedule that matches the schedule filter. Within each of these discovered schedules will be an icon for each job that matches the job filter for each schedule.

Each of these are application classes. The icon for each job represents an instance of the TWS\_JOBS application class. From each job instance icon, you can find parameters.

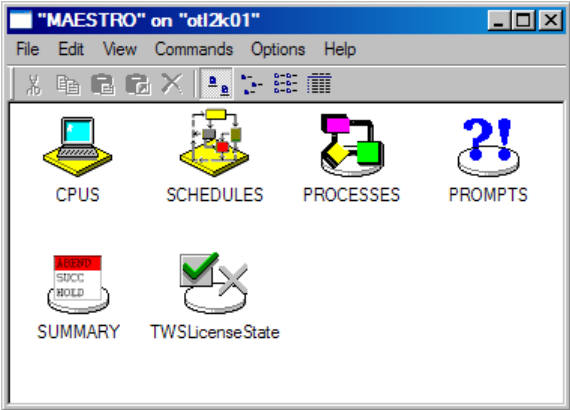


Figure 1-2 TWS KM Top Level Application Classes

The following series of figures break down each of the top level application classes. Below is the TWS\_PROMPTS application class hierarchy. Each instance is a prompt that matches the filter rule.

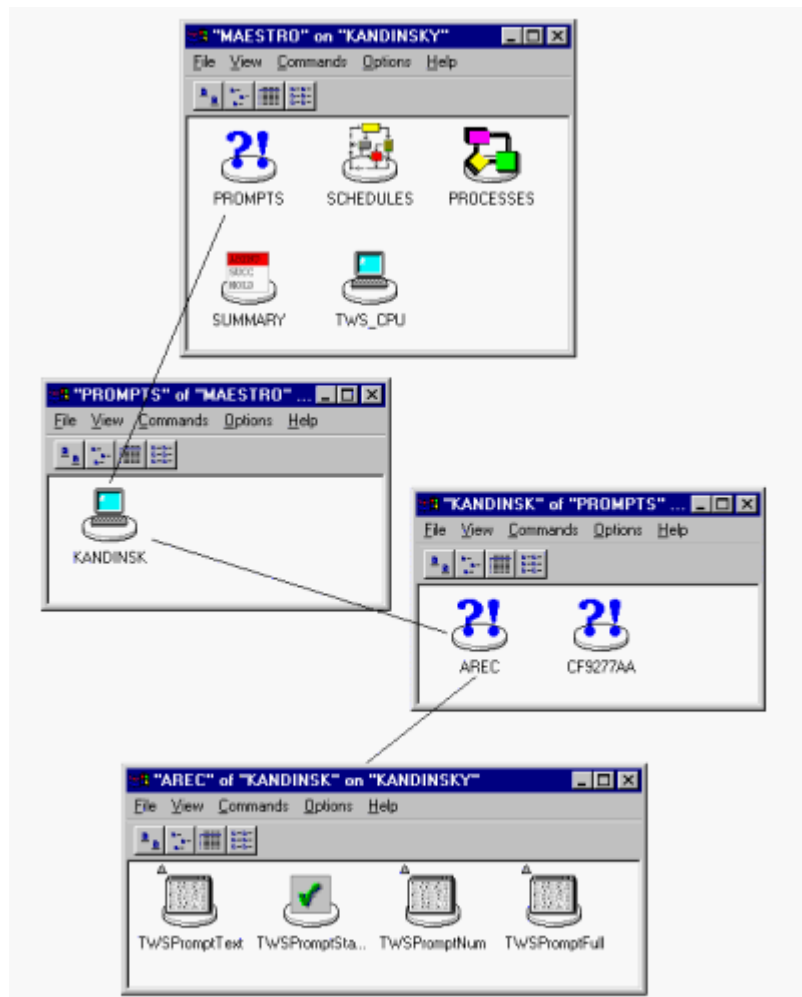


Figure 1-3 TWS KM TWS\_PROMPTS Application Class Hierarchy

Below is the TWS\_PROCESS application hierarchy. Each instance represents one of the main TWS processes.

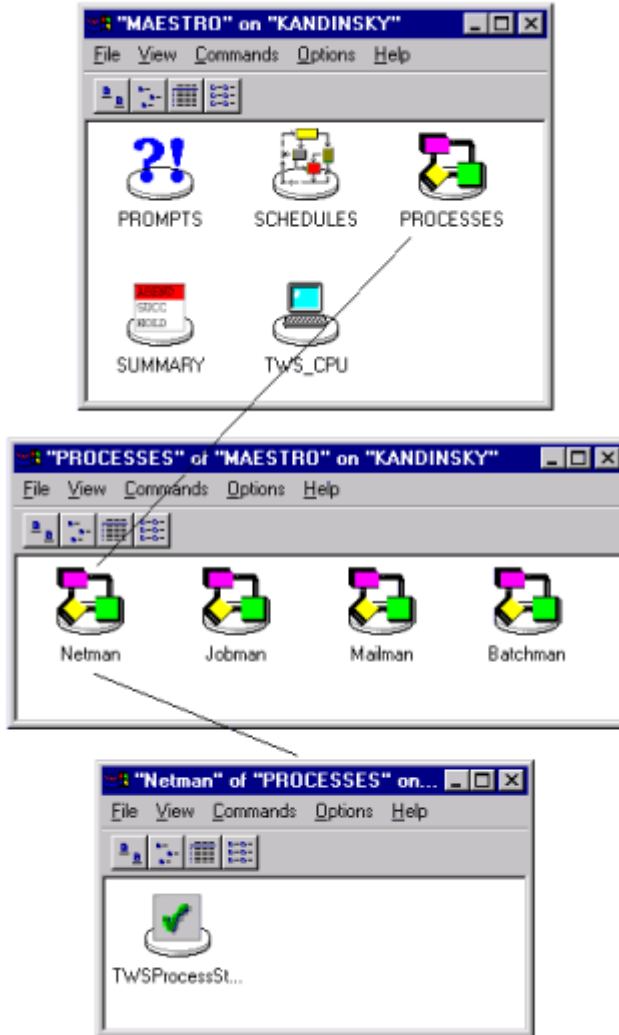


Figure 1-4 TWS KM TWS\_PROCESS Application Class Hierarchy

Below is the TWS\_CPU heirarchy.

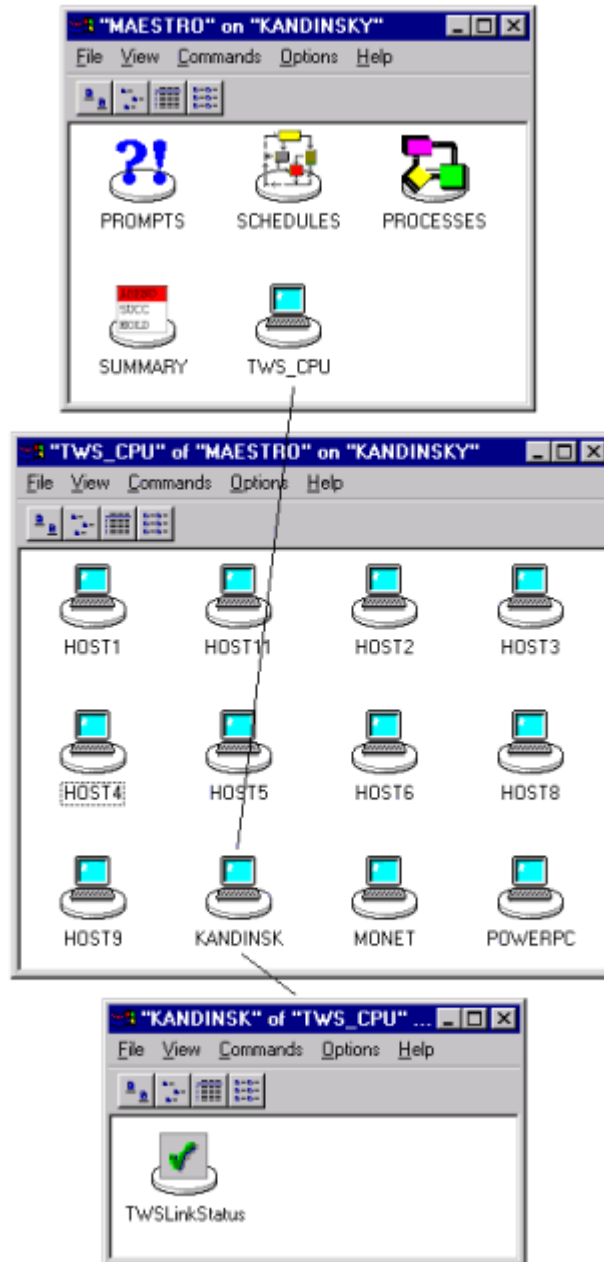
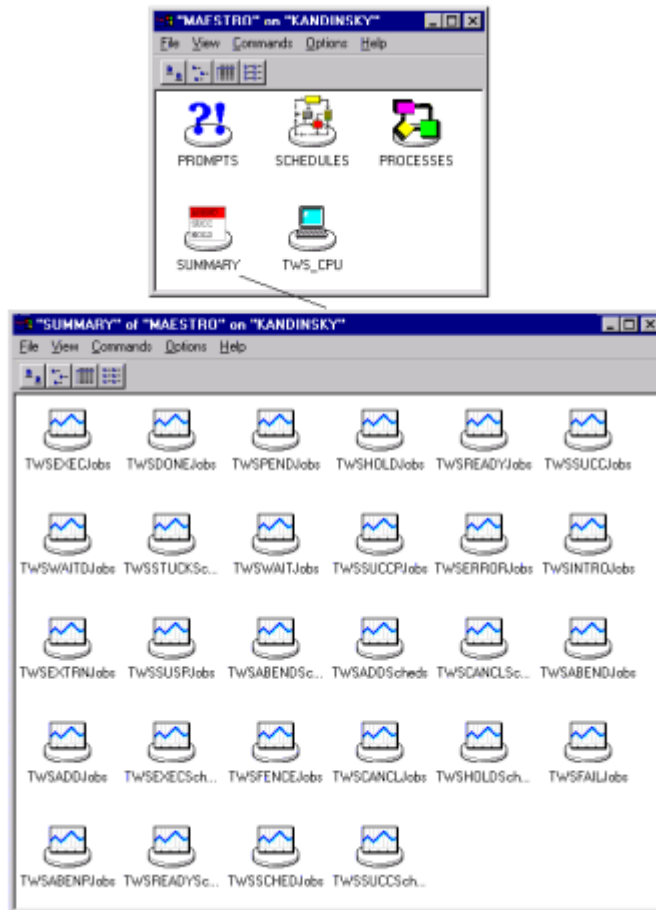


Figure 1-5 TWS KM TWS\_CPU Application Class Hierarchy



Below is the TWS\_SUMMARY providing summary information about all schedules and jobs in their respective states in TWS.



**Figure 1-6 TWS KM TWS\_SUMMARY Application Class Hierarchy**

Below is the main application class hierarchy for PATROL for Tivoli Workload Scheduler. The top level application class is TWS\_SCHEDULES\_CONT. Each CPU with schedules and jobs that match their corresponding filter is represented by an instance of TWS\_SCHEDULES\_CPU. For each CPU, an instance of TWS\_SUMMARY is created providing summary information for schedules and jobs. Each schedule is represented by an instance of TWS\_SCHEDULES. Each schedule has an instance of

TWS\_SCHEDULES\_STATUS that provides instant details on the status of the schedule and also a container for all jobs, TWS\_JOBS\_CONT. For each job matching the filter, an instance of TWS\_JOBS is created. Each job contains an instance of TWS\_JOBS\_STATUS that provides status information about the job.

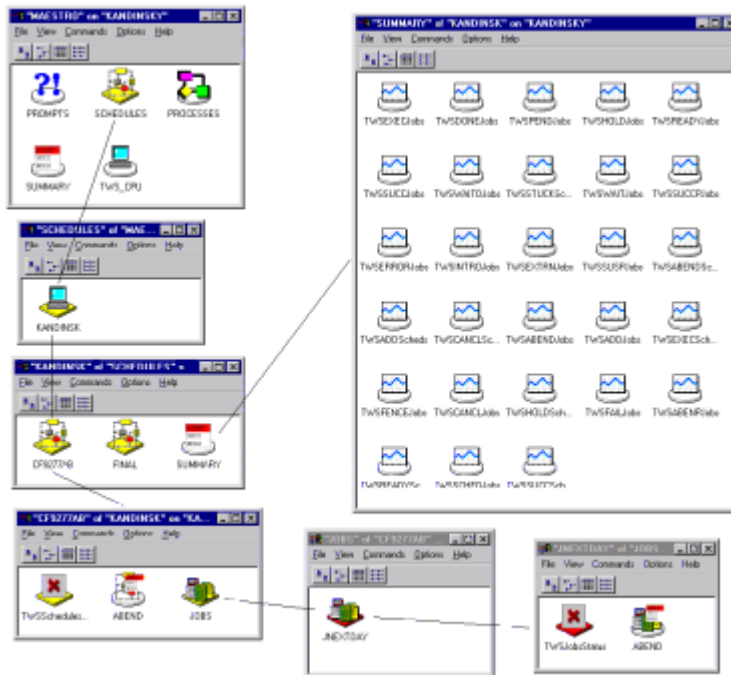


Figure 1-7 TWS KM TWS\_SCHEDULES Application Class Hierarchy

## Instance Naming

Each application in the KM uses a naming convention to differentiate the particular instance.

### **TWS\_SERVER Application Instance Naming**

There is only one instance of the TWS\_SERVER application and this is labeled **Tivoli Workload Scheduler**, after the product name or **TWS\_SETUP** if the product requires licensing.

### **TWS\_CPU\_CONT Application Instance Naming**

There is one instance of this application class labeled **CPUS**. This application class contains instances of the **TWS\_CPU** for each process being monitored.

### **TWS\_CPU Application Instance Naming**

Each CPU defined within TWS is represented by an application instance for **TWS\_CPU**. The name of each reflects the CPU defined within TWS.

### **TWS\_PROCESS\_CONT Application Instance Naming**

There is one instance of this application class labeled **PROCESSES**. This application class contains instances of the **TWS\_PROCESS** for each process being monitored.

### **TWS\_PROCESS Application Instance Naming**

Each of the key processes being monitored for TWS is represented by an application instance for **TWS\_PROCESS**.

### **TWS\_PROMPTS\_CONT Application Instance Naming**

The **TWS\_PROMPTS\_CONT** application class contains any outstanding prompts, or prompts that match the predefined filter rule within the TWS environment.

### **TWS\_PROMPTS\_CPU Application Instance Naming**

For each prompt that matches the predefined prompt filter, an instance is created for the CPU that it belongs to.

### **TWS\_PROMPTS Application Instance Naming**

Each prompt that matches the predefined prompt filter is represented by an instance of the **TWS\_PROMPTS** application class.

### **TWS\_SUMMARY Application Instance Naming**

An instance of **TWS\_SUMMARY** is created for each schedule being monitored within TWS. A main instance is also created at the top level containing summary for all schedules and jobs. This application class contains summary information relating to the number of schedules and jobs in each of the different states that are available.

### **TWS\_SCHEDULES\_CONT Application Instance Naming**

This application class is the container to hold any schedules and jobs that match the respectively defined filters. An application instance of **TWS\_SCHEDULES\_CONT** is created at the top level labeled **SCHEDULES**.

### **TWS\_SCHEDULES\_CPU Application Instance Naming**

For each schedule that matches the predefined schedule filter, an instance of **TWS\_SCHEDULES\_CPU** is created for each matching CPU.

### **TWS\_SCHEDULES Application Instance Naming**

An instance of **TWS\_SCHEDULES** is created for each schedule that matches the predefined filter. The naming of the application instances matches the name of the TWS schedule. Full extended names are supported for version 6.x of TWS.

## **TWS\_SCHEDULES\_STATUS Application Instance Naming**

For each **TWS\_SCHEDULES** instance that has been created, an instance of **TWS\_SCHEDULES\_STATUS** is also created to display the current state of the schedule. The name of the **TWS\_SCHEDULES\_STATUS** instance matches the current state of the schedule.

## **TWS\_JOBS\_CONT Application Instance Naming**

This application class is the container to hold any jobs that match the defined filter for the current schedule. An application instance of **TWS\_JOBS\_CONT** is created within each **TWS\_SCHEDULES** application instance to hold that matching jobs.

## **TWS\_JOBS Application Instance Naming**

An instance of **TWS\_JOBS** is created for each job that matches the predefined filter. The naming of the application instances matches the name of the TWS job. Full extended names are supported for version 6.x of TWS.

## **TWS\_JOBS\_STATUS Application Instance Naming**

For each **TWS\_JOBS** instance that has been created, an instance of **TWS\_JOBS\_STATUS** is also created to display the current state of the job. The name of the **TWS\_JOBS\_STATUS** instance matches the current state of the job.

# InfoBoxes

InfoBoxes display summary information about an instance or application.

## TWS\_SERVER Application InfoBox

Listed below are the InfoBox items currently available for the TWS\_SERVER application instance labelled Tivoli Workload Scheduler:

**Table 1-3 TWS\_SERVER Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
KM Version	Version of the PATROL for Tivoli Workload Scheduler Knowledge Module
KM Release Date	Date of release of the Knowledge Module
License Name	Name of the Organisation/Person this KM is licensed to
Serial Number	Serial Number of the Software License
License Key	License Activation Key
Licensed CPU's	Returns the number of CPU's the Knowledge Module is licensed to monitor. If none are licensed then Not Licensed is returned.
Installation Directory	Displays directory where the software is installed
User Name	User name that the PATROL for Tivoli Workload Scheduler software runs as.
Maestro Version	Version of Tivoli Workload Scheduler software
Maestro Platform	Platform that Tivoli Workload Scheduler software is running on
Maestro Operating System	Operating System that Tivoli Workload Scheduler software is running on

## TWS\_CPU Application InfoBox

Listed below are the InfoBox items currently available for a TWS\_CPU application instance:

**Table 1-4 TWS\_CPU Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
CPU ID	The name of the cpu to which this information applies.
Host	The name of the cpu on which <b>batchman</b> is running. For domain managers and fault-tolerant agent cpus, this is the same as CPUID. For standard agent cpus, this is the name of the domain manager. For extended agents, this is the name of the host cpu.
TCP Port Number	The TCP port number of CPUID from its cpu definition.
Node Type	Node types are: UNIX, WINT, MPEV, MPIX, and OTHER.
Cpu Type	Cpu types are: MASTER, MANAGER, FTA, S-AGENT, and X-AGENT.
Full Status	Full Status mode (on/off)
AUTO Link	Auto-link (on/off)
Mailman Server ID	The <b>mailman</b> server id (a-z, 0-9).
Link Definition	The link definition. This may be one of: only DS commands, only TCP statements, or both DS commands and TCP statements.
Jobman Version	The version of <b>jobman</b> .
Time Zone	The time zone (the value of the <b>TZ</b> environment variable). Foreextended agents, the time zone of its host.
Info	The Operating System version and cpu model. For extended agents, no information is returned.
Run	The run number of the Production Control file ( <b>Symphony</b> ).
Limit	The Maestro job limit.
Fence	The Maestro job fence.
Date	The date <b>batchman</b> started executing the <b>Symphony</b> file.
Time	The time <b>batchman</b> started executing the <b>Symphony</b> file.
Method	For extended agents only. The name of the access method specified in the cpu definition.

**Table 1-4 TWS\_CPU Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
Domain	The name of the domain in which the cpu is a member. (This applies only to PATROL for Tivoli Workload Scheduler v6.x and above).
Link State	The state of the cpu link from the Master. This is normally LINKED or UNLINKED. No entry is displayed for the Master machine.
Link Type	Type of link that has been established. This will be one of the following: DS, TCP/IP, HOST, X-AGENT, ?, or blank if the CPU is not linked.
Jobman Init	Indicated is <b>jobman</b> is initialised (YES/NO).
Jobman Running	Indicates if <b>jobman</b> is running (YES/NO).
Link to Master	The link type from the Master CPU to the selected CPU. Link types are TCP/IP, DS, HOST, and X-AGENT. If the field is blank, the CPU is not linked.

**TWS\_PROMPTS Application InfoBox**

Listed below are the InfoBox items currently available for a TWS\_PROMPTS application instance:

**Table 1-5 TWS\_PROMPTS Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
Prompt State	The states are: <b>YES</b> Replied to with "y". <b>NO</b> Replied to with "n". <b>ASKED</b> Issued, but no reply. <b>INACT</b> Not yet issued.
Prompt Message Number	The message number of a literal prompt.
CPU	The name of the cpu on which a literal prompt was issued.
Schedule	The Schedule (if applicable) that this prompt applies to.
Job	The Job (if applicable) that this prompt applies to.
Prompt	The full prompt as displayed.



## TWS\_SCHEDULES Application InfoBox

Listed below are the InfoBox items currently available for a TWS\_SCHEDULES application instance:

**Table 1-6 TWS\_SCHEDULES Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
CPU	The cpu on which the schedule runs.
Schedule	The name of the schedule.
Schedule State	The state of the schedule. Schedule states are: <b>abend</b> Schedule terminated unsuccessfully. <b>add</b> Schedule has just been submitted. <b>cancl</b> Schedule cancelled. <b>exec</b> Schedule is executing. <b>hold</b> Awaiting dependency resolution. <b>ready</b> Dependencies resolved, ready to launch. <b>stuck</b> Execution interrupted. No jobs will be launched without operator intervention. <b>succ</b> Schedule completed successfully.
Priority	The priority of the schedule.
Estimated Start	Estimated schedule start time in HH:MM format. If the time is more than 24 hours in the future, the date is listed instead of the time.
Estimated Elapse	Estimated schedule run time based on logged statistics in MM:SS format.
Estimated End	Estimated schedule end time in HH:MM:SS format.
Start	Schedule start time in HH:MM format. If the time is more than 24 hours in the past, the date is listed instead of the time.
Elapse	Schedule run time in MM:SS format.
End	Schedule end time in HH:MM:SS format.
Jobs #	The number of jobs in the schedule.
Jobs OK	The number of jobs that have completed successfully.
Sch Limit	The schedule's job limit. If blank, no limit is in effect.

## TWS\_JOBS Application InfoBox

Listed below are the InfoBox items currently available for a TWS\_JOBS application instance:

**Table 1-7 TWS\_JOBS Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
CPU	The cpu on which the job runs.
Schedule	The name of the schedule.
Job	The name of the job.
Job State	<p>The state of the schedule or job. Job states are:</p> <p><b>abend</b> Job terminated with a non-zero exit code.</p> <p><b>abexp</b> "abend" confirmation received, but job not completed.</p> <p><b>add</b> Job is being submitted.</p> <p><b>done</b> (MPE only) Job completed in an unknown state.</p> <p><b>error</b> For an internetwork dependency, an error occurred while checking for the remote status.</p> <p><b>exec</b> Job is executing.</p> <p><b>extrn</b> For an internetwork dependency, unknown status. An error occurred, a <b>rerun</b> action was just performed on the EXTERNAL job, or the remote job or schedule does not exist.</p> <p><b>fail</b> Unable to launch job.</p> <p><b>fence</b> Job 's priority is below the fence.</p> <p><b>hold</b> Job awaiting dependency resolution.</p> <p><b>intro</b> Job introduced for launching by the system.</p> <p><b>pend</b> Job completed, awaiting confirmation.</p> <p><b>ready</b> Job ready to launch, all dependencies resolved.</p> <p><b>sched</b> Job's <b>at</b> time has not arrived.</p> <p><b>succ</b> Job completed with zero exit code.</p> <p><b>succp</b> "succ" confirmation received, but job not completed.</p> <p><b>susp</b> (MPE only) Job suspended by <b>breakjob</b> command.</p> <p><b>wait</b> (X-agent jobs and MPE only) Job is being moved to <b>ready</b>.</p> <p><b>waitd</b> (MPE only) Job is in the <b>wait</b> state, and is deferred.</p>
Priority	The schedule or job priority. A plus sign (+) preceding the priority means the job has been launched.

**Table 1-7 TWS\_JOBS Application InfoBox Items**

<b>Info Item</b>	<b>Meaning</b>
Estimated Start	Estimated job start time in HH:MM format. If the time is more than 24 hours in the future, the date is listed instead of the time.
Estimated Elapse	Estimated job run time based on logged statistics in MM:SS format.
Estimated End	Estimated job end time in HH:MM:SS format.
Start	Job start time in HH:MM format. If the time is more than 24 hours in the past, the date is listed instead of the time.
Elapse	Job run time in MM:SS format.
End	Job end time in HH:MM:SS format.
Job Number	Job number of completed job.

## Where to Go from Here

The following table suggests topics that you should read next:

<b>If you want information on...</b>	<b>Refer to...</b>
How to use online help	<b>Help =&gt; Using Help</b> from the PATROL Console menu bar.
How to load and configure the PATROL KM	Chapter 2, "Getting Started," and the Help
What a certain menu command does	Chapter 3, "Menu Summary," and the Help
What a certain parameter does	Chapter 4, "Parameter Summary," and the Help
How to perform a task using this KM	Chapter 5, "Monitoring TWS," and the Help



---

# Getting Started

This chapter provides you with information that you will need to get started with PATROL for Tivoli Workload Scheduler by OTL Software (also referred to as TWS KM). The following topics are discussed:

Preparing to Use TWS KM .....	2-2
KM Requirements .....	2-2
License requirements for the KM .....	2-2
Software Requirements for the KM .....	2-2
Installing the KM .....	2-3
Installing the KM (Using BMC Installation Utility) .....	2-3
Preparing to Install or Upgrade (Using All in One Exe/Zip File) .....	2-4
Installing the KM on a Unix Platform .....	2-5
Installing the KM on a Microsoft Windows Platform .....	2-6
Installing or Upgrading the PAR file on BPPM Portal .....	2-7
Loading the KM .....	2-8
Preparing to Load the KM .....	2-8
Loading the KM on PATROL Console .....	2-8
Loading the KM on PATROL Central .....	2-9
Loading the KM on BPPM Portal .....	2-10
Configuring the KM .....	2-11
Licensing the KM .....	2-12
Agent and Console Re-start .....	2-12
Discovery Cycle .....	2-13
Help .....	2-13
Accessing Help .....	2-13
Where to Go from Here .....	2-14

## Preparing to Use TWS KM

After installing TWS KM, you must perform certain tasks before you can use the KM. If the KM has not been installed, refer to “Installing the KM” on page 2-3 for installation procedures.

Before proceeding, refer to the section “Supported Operating Systems” and “Supported Versions” on page 1-4.

## KM Requirements

This section describes the software and information requirements for setting up this KM.

### License requirements for the KM

A valid license is required before you can use the KM. The TWS KM is shipped with a built-in 30 day trial key, which is deployed automatically when the product is first installed and loaded onto a PATROL Agent. The trial key will only work on a fresh installation and it will not license any subsequent installations.

For more information on licensing, see the “Licensing the KM” on page 2-12 below.

### Software Requirements for the KM

You must meet the general operating system platform and software requirements described under “Supported Operating Systems” and “Supported Versions” on page 1-4.

# Installing the KM

This section describes steps required to install or upgrade the TWS KM.

## Installing the KM (Using BMC Installation Utility)

1. Check all prerequisites have been met.
2. Download the latest *Installation Utility* from BMC EPD site and extract the contents to create **bmc\_products** directory under a temporary directory.
3. Extract the contents of the TWS KM distribution file (**twskm\_v14xx.zip** on Microsoft Windows platforms or **twskm\_v14xx.tar** on Unix platforms) to the same temporary directory, created in Step 2 on page 2-3.
4. Start the *Installation Utility* (**setup.exe** on Microsoft Windows platforms or **setup.sh** on Unix platforms), follow the instructions and install the required components of the KM. Table 2-1 describes the contents of the TWS KM distribution file.

**Table 2-1 Contents of the Distribution File for Installation Utility**

<b>File</b>	<b>Description</b>
<b>twssagt14xxu</b>	Unix PATROL Agent installation utility files
<b>twssagt14xxw</b>	MS Windows PATROL Agent installation utility files
<b>twsscon14xxu</b>	PATROL Console for Unix installation utility files
<b>twsscon14xxw</b>	PATROL Console for MS Windows installation utility files
<b>twsscs14xxu</b>	PATROL Central Console Server for Unix installation utility files
<b>twsscs14xxw</b>	PATROL Central Console Server for MS Windows installation utility files
<b>twssws14xx</b>	PATROL Central Web Server for Unix and PATROL Central Web Server for MS Windows installation utility files
<b>otl-tws-solution-1.4.xx.par</b>	BMC ProactiveNet Performance Management Portal PAR file

## Preparing to Install or Upgrade (Using All in One Exe/Zip File)

1. Check all prerequisites have been met.
2. Extract the contents of the distribution file to a temporary folder. This distribution file can be obtained as a Microsoft Windows self-extracting file (**twsex.exe**), zip file (**twsex.zip**) or a compressed tar file (**twsex.tar.z**). Zip files can be extracted using WinZip application or PKUNZIP command (Microsoft Windows platforms) or uncompress and tar command (Unix platforms). Table 2-2 describes the contents of the TWS KM distribution file.

**Table 2-2 Contents of the Distribution File**

<b>File</b>	<b>Description</b>
<b>twsex_ug14.pdf</b>	User Guide
<b>twsex_rn&lt;release_date&gt;.pdf</b>	Release Notes
<b>twsexagt14.exe</b>	MS Windows PATROL Agent installation file
<b>twsex_r14_agent.tar</b>	Unix PATROL Agent installation file
<b>twsex_r14_console.tar</b>	PATROL Console for Unix installation file
<b>twsexcon14.exe</b>	PATROL Console for MS Windows installation file
<b>twsexconserver14.tar</b>	PATROL Central Console Server for Unix installation file
<b>twsexscs14.exe</b>	PATROL Central Console Server for MS Windows installation file
<b>twsexwebserver14.tar</b>	PATROL Central Web Server for Unix installation file
<b>twsexsws14.exe</b>	PATROL Central Web Server for MS Windows installation file
<b>otl-twsex-solution-1.4.xx.par</b>	BMC ProactiveNet Performance Management Portal PAR file

3. Read the Release Notes, and confirm all requirement for this release have been met.
4. If you are upgrading the TWS KM, please follow the steps described under “Uninstalling the KM” on page 5-20 to uninstall the old version of the KM before attempting to install the new version.



# Installing the KM on a Unix Platform

1. Copy or ftp appropriate installation files under the paths for the relevant systems as shown in Table 2-3.

**Table 2-3 Unix Platform Installation Files and Extraction Paths**

<b>File</b>	<b>Path</b>
<b>tw_s_r14_agent.tar</b>	<b>\$PATROL_HOME/</b> on monitored Unix servers  <i>Example:</i> <b>/opt/bmc/Patrol3/</b> or <b>/opt/bmc/Patrol3/Solaris29-sun4/</b>
<b>tw_s_r14_console.tar</b>	<b>\$PATROL_HOME/</b> on PATROL Console for Unix  <i>Example:</i> <b>/opt/bmc/Patrol3/</b> or <b>/opt/bmc/Patrol3/Solaris29-sun4/</b>
<b>tw_sconserver14.tar</b>	<b>\$PATROL_ROOT/</b> on PATROL Central Console Server for Unix  <i>Example:</i> <b>/opt/bmc/Patrol7/</b>
<b>tw_swebserver14.tar</b>	<b>\$BMC_ROOT/webcentral/</b> on PATROL Central Web Server for Unix  <i>Example:</i> <b>/opt/bmc/webcentral/</b>

2. Extract the contents of the installation **.tar** file as PATROL user, using:  

```
tar xvf <file name>
```
3. Remove the installation **.tar** file copied in Step 1.

## Installing the KM on a Microsoft Windows Platform

1. Copy or ftp appropriate installation files to a temporary folder (such as **C:\temp**) on the relevant system.
2. Double-click the file and extract the contents of the self-extracting files to relevant paths as shown in Table 2-4

**Table 2-4 MS Windows Platform Installation Files and Extraction Paths**

<b>File</b>	<b>Path</b>
<b>twragt14.exe</b>	%PATROL_HOME%\ on monitored servers for Tivoli Workload Scheduler for Microsoft Windows  <b>Example:</b> <b>C:\Program Files\BMC Software\Patrol3\</b>
<b>twsgon14.exe</b>	%PATROL_HOME%\ on PATROL Console for MS Windows  <b>Example:</b> <b>C:\Program Files\BMC Software\Patrol3\</b>
<b>twsgcs14.exe</b>	%PATROL_ROOT%\ on PATROL Central Console Server for MS Windows  <b>Example:</b> <b>C:\Program Files\BMC Software\Patrol7\</b>
<b>twsgws14.exe</b>	%BMC_ROOT%\WebCentral\ on PATROL Central Web Server for MS Windows  <b>Example:</b> <b>C:\Program Files\BMC Software\WebCentral\</b>

3. Remove the self-extracting installation file copied in Step 1.

## Installing or Upgrading the PAR file on BPPM Portal

The PAR file enables BMC ProactiveNet Performance Management Portal to retrieve the KM data from the PATROL Agent.

1. Log on to the BMC ProactiveNet Performance Management Portal with portal credentials, and select the **Portal** tab.
2. Under **Tasks** in the navigation pane, select **Performance Managers**.
3. Click **Upload**
4. Click **Browse** and then select the PAR file extracted under the temporary folder on the local system. For complete integration select **otl-tws-solution-1.4.xx.par** and for static components only integration (without Job or Mount Request integration) select **otl-tws-express-solution-1.4.xx.par**.
5. Click **Upload**.

---

### Note

If you are upgrading the PAR file on BMC ProactiveNet Performance Management Portal, the initial Status of the newly uploaded PAR solution on Portal is “**Unpublished**”.

To push-out the new version to effect the upgrade process, select the checkbox next to the new version PAR solution on Portal, and click **Publish**.

---

# Loading the KM

This section provides instructions to load TWS KM on Microsoft Windows and Unix platforms.

Before you load the TWS KM, you must install the KM on PATROL Console and PATROL Agent systems, following the instructions in “Installing the KM” on page 2-3.

## Preparing to Load the KM

1. Start the PATROL Console and update the connection to all PATROL Agent systems where the TWS KM is installed.
2. Check the value of the PATROL Agent tuning variable, “/AgentSetup/AgentTuning/psInstructionMax”, and if necessary, increase it.

## Loading the KM on PATROL Console

1. From the PATROL Console menu bar, choose **File => Load KM...**
2. Select the **TWS\_LOAD.kml** file, and click **Open** or **OK**. The TWS KM will be loaded to the PATROL Console, and all connected PATROL Agents will start discovering the Tivoli Workload Scheduler environment. If the automatic discovery successfully finds the Tivoli Workload Scheduler installation, **TWS\_SETUP** instance will be instantiated, as shown in Figure 2-1.

---

**Note**

---

This automatic discovery may take up to 5 minutes to instantiate the **TWS\_SETUP** instance. Look for any error messages on the PATROL Console System Output Window (SOW) during the initial discovery..

---



**Figure 2-1** TWS\_SETUP Icon

3. Select **File => Save Configuration** to save the new list of loaded KMs as the PATROL Console user preference.
4. Repeat the above steps on each PATROL Console.

## Loading the KM on PATROL Central

1. Right click on the **PATROL Main Map**, and choose **Load Knowledge Modules...** A wizard box will display a list of all managed systems.
2. Select the managed systems where the TWS KM is to be loaded, and click **Next>**. The wizard will display a list of all available **.kml** files for the managed systems you selected.
3. Select the **TWS\_LOAD.kml** file for each managed system, click **Next>** and **Finish**.

---

**Note**

---

If you cannot find the **TWS\_LOAD.kml** file in the wizard list for any system you selected, then the KM has not been installed on that system. Check that you have followed the instructions in “Installing the KM” on page 2-3.

---

The TWS KM will be loaded to the PATROL Central Console Server, and all connected selected managed systems will start discovering the Tivoli Workload Scheduler environment. If the automatic discovery successfully finds the Tivoli Workload Scheduler installation, the **TWS\_SETUP** instance will be instantiated, as shown in Figure 2-1.

---

**Note**

---

This automatic discovery may take up to 5 minutes to instantiate **TWS\_SETUP** instance.

---

4. Repeat the above steps for each PATROL Central Management Profile where TWS KM is to be unloaded.

## Loading the KM on BPPM Portal

1. Logon to BMC ProactiveNet Performance Management Portal with appropriate credentials to add infrastructure elements.
2. Click on **Configure** tab.
3. Under **Tasks**, select **Elements** to open the Elements page.
4. Click in **Add**.
5. Select **Infrastructure Element**, and click **Next**.
6. Select appropriate RSM, and click **Next**.
7. Enter the label for the element, select **PATROL Integration**, and click **Next**.
8. Select/create a group name, and click **Next**.
9. Select appropriate method to discover the PATROL Agent hosts, and click **Next**.
10. Fill the required details for discovering the PATROL Agents, and click **Commit**.

# Configuring the KM

---

**Note**

---

This version of the TWS KM cannot be configured from BMC ProactiveNet Performance Management Portal. Use PATROL Console or PATROL Central Console to access the KM configuration menus (KM commands) described below.

---

If the automatic KM discovery detects that Tivoli Workload Scheduler is installed but the Knowledge Module licensing information has not been entered, then the TWS\_SETUP application class icon will appear. This allows the user to enter the PATROL for Tivoli Workload Scheduler license information.



**Figure 2-2 TWS\_SETUP Icon**

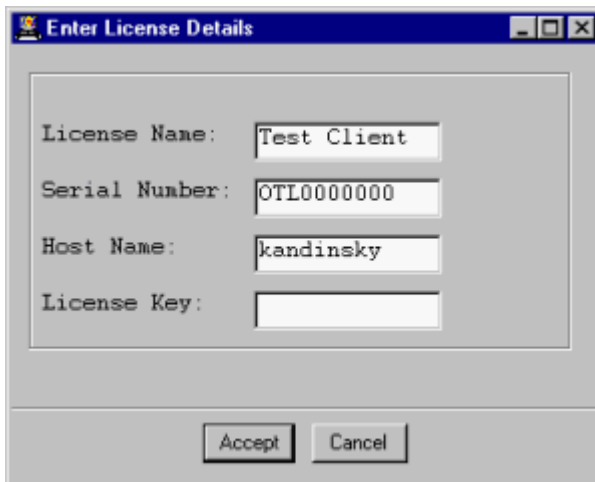
At this point the KM needs to be licensed before the KM is able to start operation.

PATROL for Tivoli Workload Scheduler is shipped with a free 30 day trial license. As soon as KM is installed and loaded to the PATROL Agent on a new managed node, the trial license will be automatically activated. The trial license gives full monitoring capability for the KM, and works only once on a managed node.

If you want to purchase a permanent license key, contact your local reseller or OTL Software Limited. To generate the license key, you will need to provide your system architectre details.

## Licensing the KM

Licensing will be the main item to be performed at this stage. This is done by selecting a menu item available from TWS\_SETUP, **KM Setup** => **License KM**.



**Figure 2-3 License TWS KM Menu**

This window allows the user to license the TWS KM. The KM is licensed on a per-server basis.

Once the licensing has been successful, the main **TWS\_SETUP** icon will be replaced with **Tivoli Workload Scheduler**.

## Agent and Console Re-start

At this point it is optional whether or not you re-start the PATROL Agent and Console. Re-starting the Agent and Console will force the loading of the new KM files into memory, and will start the discovery process straight away. Leaving the PATROL Agent running will mean that the Discovery process will take longer.



## Discovery Cycle

Discovery may take up to 15 minutes to automatically find all of the CPU's, prompts, schedules and jobs present in the system. Discovery can be forced by choosing **Utilities => Patrol => Force Discovery** from the host application class menu. Individual schedule and job details will be discovered on the next cycle the TWSSchedulesColl collector is run after discovery is finished.

## Help

Help describes the function of the currently displayed window or dialog box and the use of its elements. The tasks in this section describe how to access help.

## Accessing Help

*Summary:* You can access help from the PATROL Console through the List of Applications Classes window, the parameter window, and the parameter pop-up menu.

---

### To Access Help from the List of Applications Classes Window with the PATROL Console for Unix

- » Select the application class and choose **Help => This Application** from the List of Applications Classes window.

### To Access Help from Context-Sensitive Parameter Help with the PATROL Console for Unix

- » Select the parameter from the parameter pop-up window and choose **Help**


Or

» Right-click any parameter pop-up window and choose **Help On**.

Or

» Choose **Help => This Window** from any parameter window.

### To Access Help from Context-Sensitive Parameter Help with the PATROL Console for Windows NT

» Click on  from any parameter window.

Or

» Right-click any parameter pop-up window and choose **Help On**.

## Where to Go from Here

The following table suggests topics that you should read next:

<b>If you want information on...</b>	<b>See...</b>
How to use help	<b>Help =&gt; Using Help</b> from the PATROL Console for Unix menu bar.
What a certain menu command does	Chapter 3, "Menu Summary," and TWS KM help.
What a certain parameter does	Chapter 4, "Parameter Summary," and TWS KM help.
How to perform a task using this KM	Chapter 5, "Monitoring TWS." and TWS KM help

---

## Menu Summary

This chapter summarizes the application menus and menu commands for PATROL for Tivoli Workload Scheduler by OTL Software. The application menu architecture is provided in a table for each of the application classes.

When a Knowledge Module (KM) is loaded, its associated menu commands are added to the KM area of a menu. The KM area of a menu is below the menu's horizontal line. Menu commands above the line belong to the Console.

This chapter describes KM menu commands only. The KM help system provides further details about these menu commands. For descriptions of Console menu commands, refer to the appropriate PATROL user guide for your Console. The following topics are discussed:

Menu Summary . . . . .	3-3
TWS_MAESTRO Application Menu . . . . .	3-4
TWS_CPU_CONT Application Menu . . . . .	3-9
TWS_CPU Application Menu . . . . .	3-10
TWS_PROMPTS_CONT Application Menu . . . . .	3-12
TWS_PROMPTS Application Menu . . . . .	3-13
TWS_SCHEDULES_CONT Application Menu . . . . .	3-14
TWS_SCHEDULES_CPU Application Menu . . . . .	3-15
TWS_SCHEDULES Application Menu . . . . .	3-16
TWS_JOBS_CONT Application Menu . . . . .	3-18
TWS_JOBS Application Menu . . . . .	3-19
Where to Go from Here . . . . .	3-21

# Accessing Application Menus

To access application menu items, perform the following steps:

---

**Note**

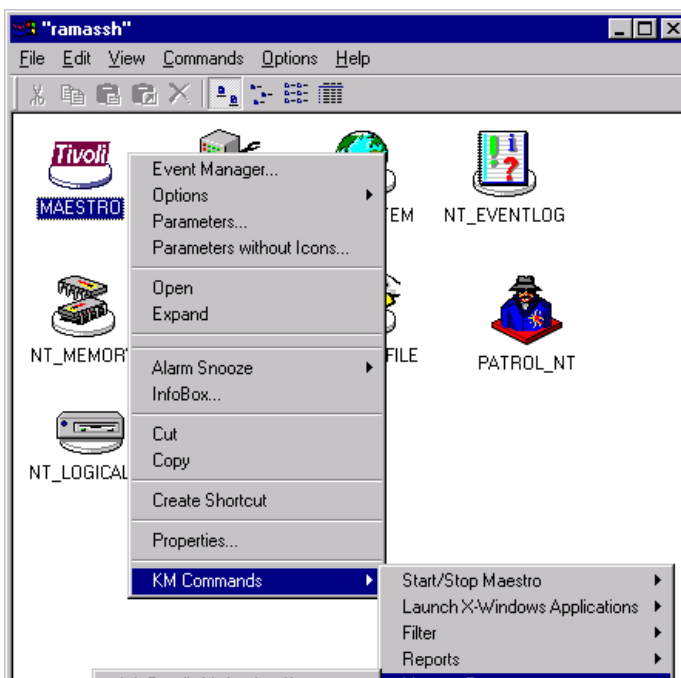
A summary of each menu item is provided later in this section.

---

**Step 1** To access the application menu, perform one of the following actions:

- **With the PATROL Console for Unix**, click and hold MB3 on a PATROL Console icon.
- **With the PATROL Console for Windows**, right-click a PATROL Console icon.

The main menu is displayed as shown in the figure below.



**Figure 3-1 Application Menus**

**Step 2** Select the appropriate menu item to perform the required task.

# Menu Summary

This section describes PATROL for Tivoli Workload Scheduler by OTL Software application menus for the following application classes.

- TWS\_MAESTRO
- TWS\_SETUP
- TWS\_CPU\_CONT
- TWS\_CPU
- TWS\_PROMPTS\_CONT
- TWS\_PROMPTS
- TWS\_SCHEDULES\_CONT
- TWS\_SCHEDULES\_CPU
- TWS\_SCHEDULES
- TWS\_JOBS\_CONT
- TWS\_JOBS

---

**Note**

---

The TWS\_PROCESS\_CONT, TWS\_PROCESS, TWS\_SUMMARY, TWS\_PROMPTS\_CPU, TWS\_SCHEDULES\_STATUS, and TWS\_JOBS\_STATUS application classes do not have menu items for performing specific TWS KM functions.

---

## TWS\_MAESTRO Application Menu

The TWS\_MAESTRO application menu is available from the Tivoli Workload Scheduler or TWS\_SETUP icon as shown in Figure 3-1.

The TWS\_MAESTRO application menu has the following menu items:

**Table 3-1 Menu Items for TWS\_MAESTRO Application**

<b>Menu</b>	<b>Action</b>
<b>Start/Stop Maestro</b>	The sub-menu which contains the administration functions to start and stop the Tivoli Workload Scheduler processes.
Start NETMAN (StartUp)	This option asks for confirmation to proceed with starting the Tivoli Workload Scheduler process (netman). The user is prompted with a login and password screen in order to run the command to start the process. The account entered must have the necessary permissions to start the process. This option does not start the other processes in the Maestro Process Tree.
Stop Maestro (conman shutdown)	This option asks for confirmation to proceed with stopping all the Tivoli Workload Scheduler processes. The user is prompted with a login and password screen in order to run the command to stop the processes. The account entered must have the necessary permissions to stop the processes. This option will stop all the processes in the Maestro Process Tree.
Start Processes (conman start)	This option asks for confirmation to proceed with starting the Tivoli Workload Scheduler processes. The user is prompted with a login and password screen in order to run the command to start the processes. The account entered must have the necessary permissions to start the processes. This option will start all the processes in the Maestro Process Tree.
Stop Processes (conman stop)	This option asks for confirmation to proceed with stopping the Tivoli Workload Scheduler processes. The user is prompted with a login and password screen in order to run the command to stop the processes. The account entered must have the necessary permissions to stop the processes. This option will stop the processes in the Maestro Process Tree except for netman.

<b>Menu</b>	<b>Action</b>
<b>Launch X-Windows Applications</b>	The sub-menu which launches Tivoli Workload Scheduler text or X-Windows programs for UNIX machines only.
Launch GMAESTRO	Launch main Maestro GUI Application where gcomposer and gmaestro can be launched. Menu item only available on UNIX. Must have an X-Windows application to display the output. Menu item requires a login and password to run the command.
Launch GCOMPOSER	Launch gcomposer GUI Application to create or modify scheduling objects. Menu item only available on UNIX. Must have an X-Windows application to display the output. Menu item requires a login and password to run the command.
Launch COMPOSER	Launch composer Text-based Application to create or modify scheduling objects. Output will be in an Xterm window. Menu item only available on UNIX. Must have an X-Windows application to display the output. Menu item requires a login and password to run the command.
Launch GCONMAN	Launch gconman GUI Application for managing the production day environment. Menu item only available on UNIX. Must have an X-Windows application to display the output. Menu item requires a login and password to run the command.
Launch CONMAN	Launch conman Text-based for managing the production day environment. Output will be in an Xterm window. Menu item only available on UNIX. Must have an X-Windows application to display the output. Menu item requires a login and password to run the command.
<b>Filter</b>	The sub-menu provides windows to define filters for schedules, jobs, and prompts.
Schedules	Filter window to specify the schedules that will be monitored. Schedules can be filtered on a state basis (e.g. ABEND, STUCK, SUCC, etc.) or on a CPU#SCHEDULE name basis. The states are all listed and those of interest can be selected from the list. By default schedules in the STUCK and ABEND states are monitored. In addition, specific schedules can be monitored. By default the @#FINAL schedule is always monitored. Standard Maestro wildcards can be used (e.g. @, ?, and %) to assist in looking for schedules that match a specific pattern (e.g. list all schedules belonging to the Human Resources department could be shown using a filter of @#HR@ if all the schedules for the HR department began with HR.

<b>Menu</b>	<b>Action</b>
Jobs	Filter window to specify the jobs that will be monitored. Jobs can be filtered on a state basis (e.g. ABEND, FAIL, SUCC, etc.) or on a CPU#SCHEDULE.JOB name basis. The states are all listed and those of interest can be selected from the list. Alternately, all can be selected by clicking on the All Filters button. By default jobs in the ABEND, ABENP, DONE, ERROR, EXTRN, and FAIL states are monitored. In addition, specific jobs can be monitored. By default the @#@.JNEXTDAY job is always monitored. Standard Maestro wildcards can be used (e.g. @, ?, and %) to assist in looking for jobs that match a specific pattern (e.g. list all jobs belonging to the Human Resources department could be shown using a filter of @#@.HR@ if all the jobs for the HR department began with HR.
Prompts	Filter window to specify the prompts that will be monitored. Prompts can be filtered on a state basis (e.g. INACT, ASKED, YES, and NO). By default prompts in the ASKED state are monitored.
<b>Reports</b>	The sub-menu which launches Tivoli Workload Scheduler reports to provide important information.
List Symphony Log Files	Reports on available Symphony Log Files that are currently available on the server.
Show All Schedules	Reports on all schedules that are running in the current production day.
Show All Jobs	Reports on all jobs that are running in the current production day.
Show All Resources	Reports on all resource dependencies that are defined against the current production day.
Show All Files	Reports on all file dependencies that are defined against the current production day.
Show All Prompts	Reports on all prompts that have, or will be issued in the current production day.
<b>Maestro Reports</b>	The sub-menu which launches Tivoli Workload Scheduler reports to provide important information.
Job Details Listing (rep1)	Runs the standard Tivoli Workload Scheduler report rep1.
Prompt Messages Listing (rep2)	Runs the standard Tivoli Workload Scheduler report rep2.
User Calendar Listing (rep3)	Runs the standard Tivoli Workload Scheduler report rep3.
User Parameters Listing (rep4a)	Runs the standard Tivoli Workload Scheduler report rep4a.



<b>Menu</b>	<b>Action</b>
Resource Listing (rep4b)	Runs the standard Tivoli Workload Scheduler report rep4b.
Job History Listing (rep7)	Runs the standard Tivoli Workload Scheduler report rep7.
Job Histogram (rep8)	Runs the standard Tivoli Workload Scheduler report rep8.
Planned Prod Schedule (rep11)	Runs the standard Tivoli Workload Scheduler report rep11.
Planned Prod Summary (reptr)	Runs the standard Tivoli Workload Scheduler report reptr -pre -summary.
Planned Prod Detail (reptr)	Runs the standard Tivoli Workload Scheduler report reptr -pre -detail.
Actual Prod Summary (reptr)	Runs the standard Tivoli Workload Scheduler report reptr -post -summary.
Actual Prod Detail (reptr)	Runs the standard Tivoli Workload Scheduler report reptr -post -detail.
<b>Validate Configurations</b>	The sub-menu which dumps the scheduling objects from the database (using the composer "create" command) and runs the validate command against them.
Calendars	Validates Calendar definitions.
CPU's	Validates CPU definitions.
Jobs	Validates Job definitions.
Parameters	Validates Parameter definitions.
Prompts	Validates Prompt definitions.
Resources	Validates Resource definitions.
Schedules	Validates Schedule definitions.
Users	Validates User definitions.
<b>KM Setup</b>	The sub-menu which contains the initial setup menus to license the knowledge module.
License Tivoli Maestro KM	Set the License Information for the Knowledge Module.

<b>Menu</b>	<b>Action</b>
psetcode Information	Provides licensing and validation code information from Tivoli Workload Scheduler using the psetcode -d and psetcode -i commands.
Set Number of Instances	This option allows some control over the number of application class instances that can be discovered. By default a maximum of 200 CPU definitions and 1000 Jobs will be discovered. This allows for some control over the amount of resources that PATROL can consume in large installations. Any instances exceeding these limits are recorded into the Event Manager and a message is displayed onto the console output window.
Reset Global Channel	This option will reset the global channel used for collecting information from Tivoli Workload Scheduler. The global channel will close the session to conman and restart a new channel.
Debug	This option turns on debugging information for several areas of reporting. It is intended to provide limited additional information output onto the system console window.
Refresh Parameters	This option will refresh PATROL parameters for the TWS_MAESTRO application class.
About	This option gives details about the OTL Software and the KM version.

# TWS\_CPU\_CONT Application Menu

The TWS\_CPU\_CONT menu is shown in Figure 3-2

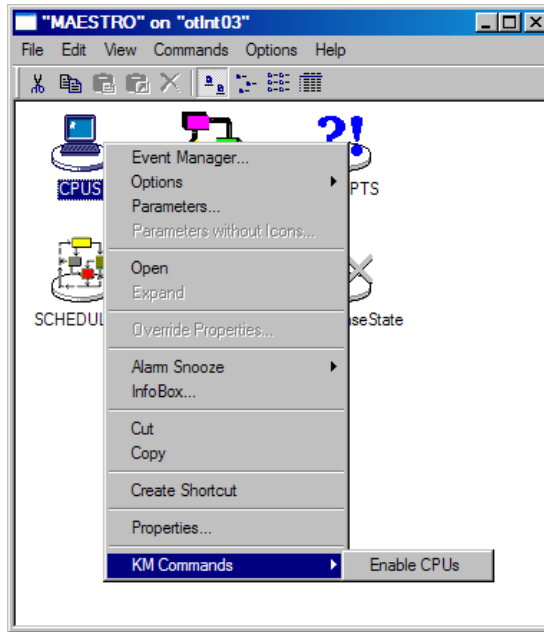


Figure 3-2 TWS\_CPU\_CONT Application Menu

The TWS\_CPU\_CONT application menu has the following menu items:

Table 3-2 Menu Items for TWS\_PROMPTS\_CONT Application

Menu	Action
Enable CPUs	Allows selected CPUs to be enabled for monitoring. By default, all CPUs are monitored.

## TWS\_CPU Application Menu

The TWS\_CPU menu is shown in Figure 3-3

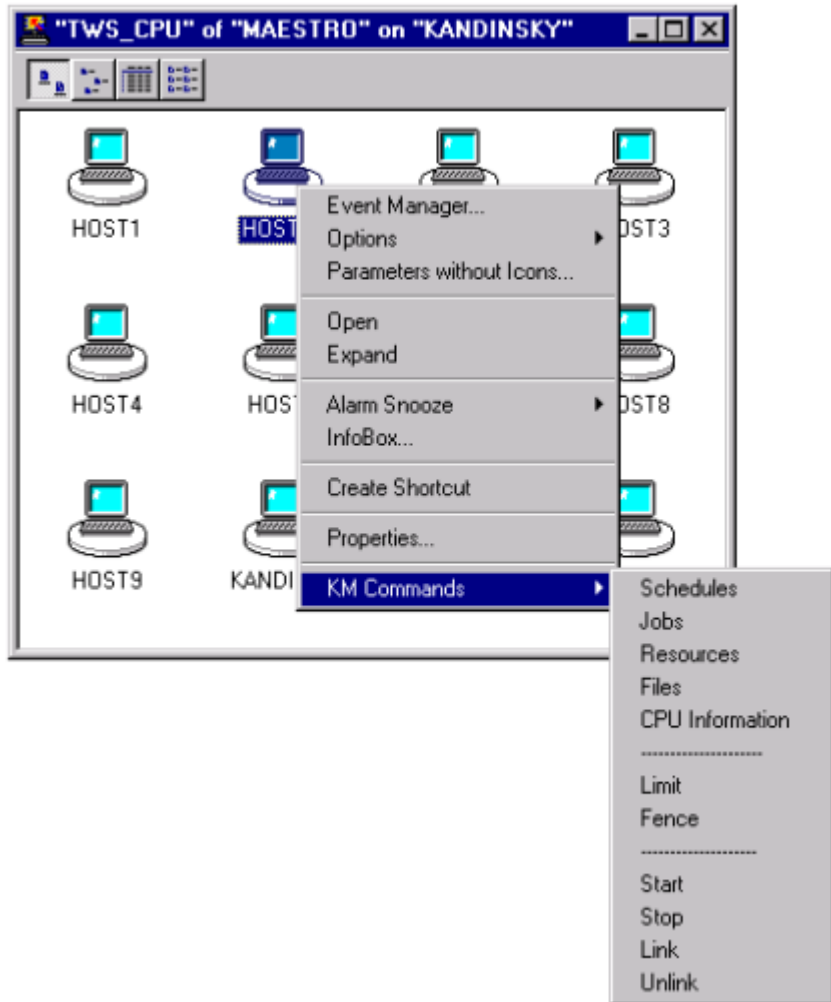


Figure 3-3 TWS\_CPU Application Menu

The TWS\_CPU application menu has the following menu items:

**Table 3-3 Menu Items for TWS\_CPU Application**

<b>Menu</b>	<b>Action</b>
Schedules	Reports on current schedules that are running against this CPU.
Jobs	Reports all jobs that are run against the current CPU.
Resources	Reports on all resource dependencies that are defined in the production day for the current CPU.
Files	Reports on all file dependencies that are defined in the production day for the current CPU.
CPU Information	Reports on CPU information by running the cpuserinfo command.
Limit	Changes the CPU Limit for the number of job that can be simultaneously run against this CPU.
Fence	Changes the Fence limit for jobs that run against this CPU. A job must have a priority one higher than the Fence limit to actually run.
Start	Issues a conman start command to the CPU.
Stop	Issues a conman stop command to the CPU.
Link	Issues a link command to the CPU.
Unlink	Issues an unlink command to the CPU.

# TWS\_PROMPTS\_CONT Application Menu

The TWS\_PROMPTS\_CONT menu is shown in Figure 3-4



Figure 3-4 TWS\_PROMPTS\_CONT Application Menu

The TWS\_PROMPTS\_CONT application menu has the following menu items:

Table 3-4 Menu Items for TWS\_PROMPTS\_CONT Application

Menu	Action
Filter Prompts	Filter window to specify the prompts that will be monitored. Prompts can be filtered on a state basis (e.g. INACT, ASKED, YES, and NO). By default prompts in the ASKED state are monitored.
Show All Prompts	Report to show all prompts in all states.

# TWS\_PROMPTS Application Menu

The TWS\_PROMPTS menu is shown in Figure 3-5



**Figure 3-5 TWS\_PROMPTS Application Menu**

The TWS\_PROMPTS application menu has the following menu items:

**Table 3-5 Menu Items for TWS\_PROMPTS Application**

<b>Menu</b>	<b>Action</b>
<b>Reply</b>	The sub-menu which contains the menus to respond to a prompt.
Yes	Respond Yes to the prompt.
No	Respond No to the prompt.
Dependent Objects	Report on the dependencies on the prompt.

## TWS\_SCHEDULES\_CONT Application Menu

The TWS\_SCHEDULES\_CONT menu is shown in Figure 3-6

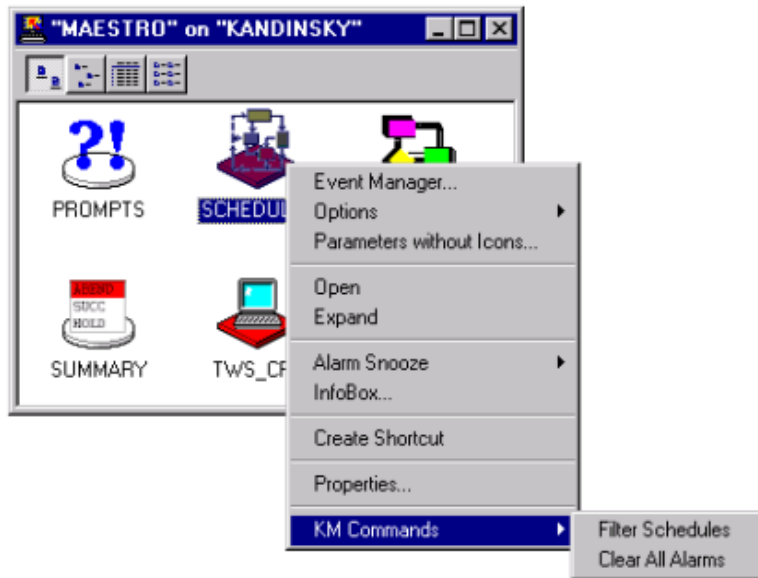


Figure 3-6 TWS\_SCHEDULES\_CONT Application Menu

The TWS\_SCHEDULES\_CONT application menu has the following menu items:

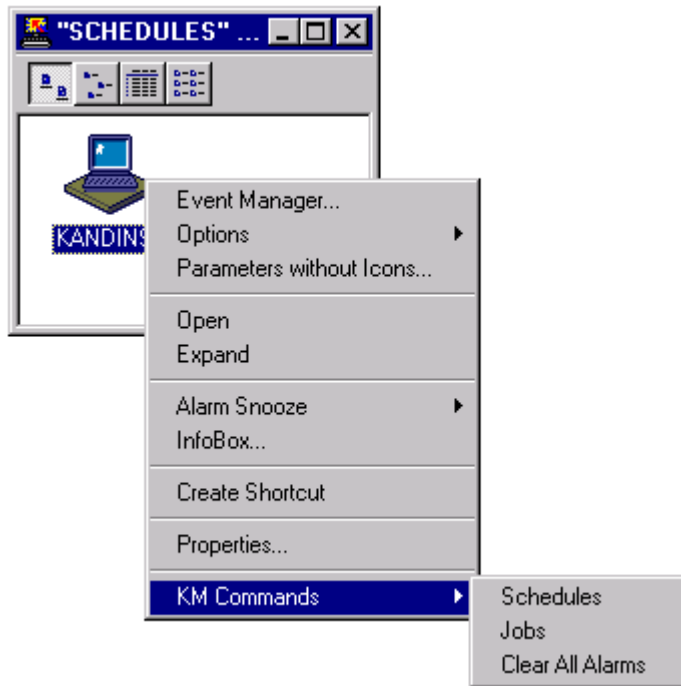
Table 3-6 Menu Items for TWS\_SCHEDULES\_CONT Application

Menu	Action
Filter Schedules	Filter window to specify the schedules that will be monitored. Schedules can be filtered on a state basis (e.g. ABEND, STUCK, SUCC, etc.) or on a CPU#SCHEDULE name basis. The states are all listed and those of interest can be selected from the list. By default schedules in the STUCK and ABEND states are monitored. In addition, specific schedules can be monitored. By default the @#FINAL schedule is always monitored. Standard Maestro wildcards can be used (e.g. @, ?, and %) to assist in looking for schedules that match a specific pattern (e.g. list all schedules belonging to the Human Resources department could be shown using a filter of @#HR@ if all the schedules for the HR department began with HR.
Clear All Alarms	Clears all Schedule and Job alarms. This applies directly to the parameters TWSSchedulesStatus and TWSJobsStatus.



## TWS\_SCHEDULES\_CPU Application Menu

The TWS\_SCHEDULES\_CPU menu is shown in Figure 3-7



**Figure 3-7 TWS\_SCHEDULES\_CPU Application Menu**

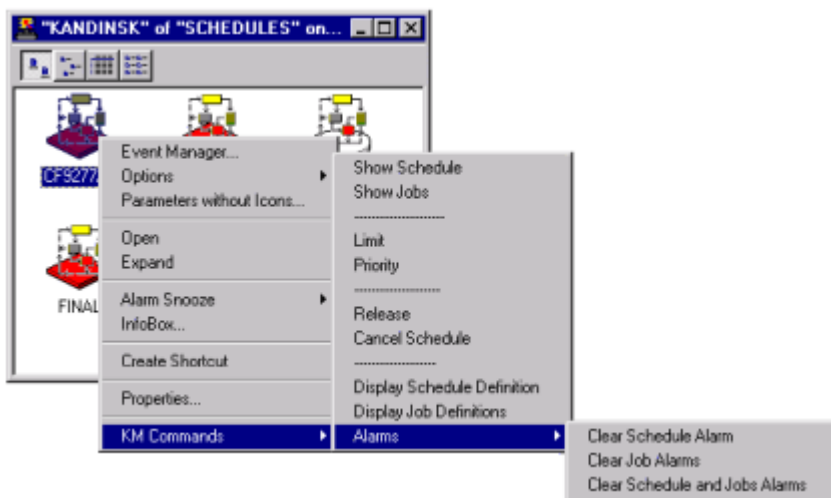
The TWS\_SCHEDULES\_CPU application menu has the following menu items:

**Table 3-7 Menu Items for TWS\_SCHEDULES\_CPU Application**

<b>Menu</b>	<b>Action</b>
Schedules	Reports on current schedules that are running against this CPU.
Jobs	Reports all jobs that are run against the current CPU.
Clear All Alarms	Clears all Schedule and Job alarms for the selected CPU. This applies directly to the parameters TWSSchedulesStatus and TWSJobsStatus.

## TWS\_SCHEDULES Application Menu

The TWS\_SCHEDULES menu is shown in Figure 3-8



**Figure 3-8 TWS\_SCHEDULES Application Menu**

The TWS\_SCHEDULES application menu has the following menu items:

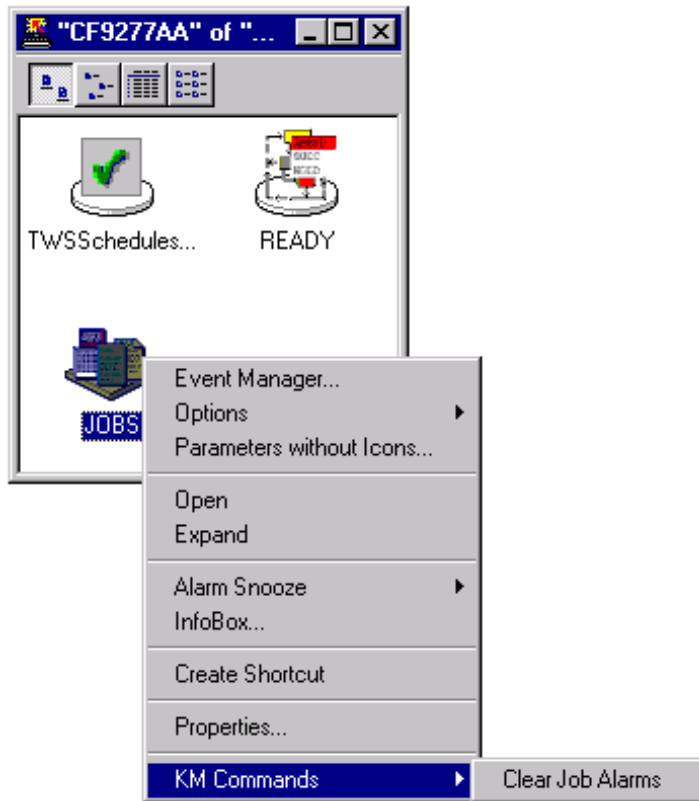
**Table 3-8 Menu Items for TWS\_SCHEDULES Application**

Menu	Action
Show Schedule	Reports on the selects schedule providing status information.
Show Jobs	Reports on the jobs currently running against the selected schedule.
Limit	Changes the job limit for the selected schedule. This limits the number of jobs that can run simultaneously in the schedule. This limit is over-ruled by the CPU limit defined.
Priority	Changes the priority for the schedule. Only schedules and jobs with a priority higher than the CPU's Fence limit will be able to run.
Release	Release the selected schedule's dependencies.

<b>Menu</b>	<b>Action</b>
Cancel Schedule	Cancels the selected schedule. If a schedule is cancelled before it is launched, it will not launch. If cancelled after it is launched, its executing jobs are allowed to complete, but no other jobs are launched. Jobs and schedules that are dependent on the cancelled schedule are released immediately.
Display Schedule Definition	Display schedule definitions
Display Job Definition	Displays the contents of the job's script files for all the jobs defined for the current schedule.
<b>Alarms</b>	The sub-menu which contains the menus to respond to an alarm.
Clear Schedule Alarm	Clears the TWSSchedulesStatus alarm for the current schedule.
Clear Job Alarms	Clears all the job alarms for the parameters TWSJobsStatus for the current schedule.
Clear Schedule and Jobs Alarms	Clears the schedule and job alarms for the selected schedule.

# TWS\_JOBS\_CONT Application Menu

The TWS\_JOBS\_CONT menu is shown in Figure 3-9



**Figure 3-9 TWS\_JOBS\_CONT Application Menu**

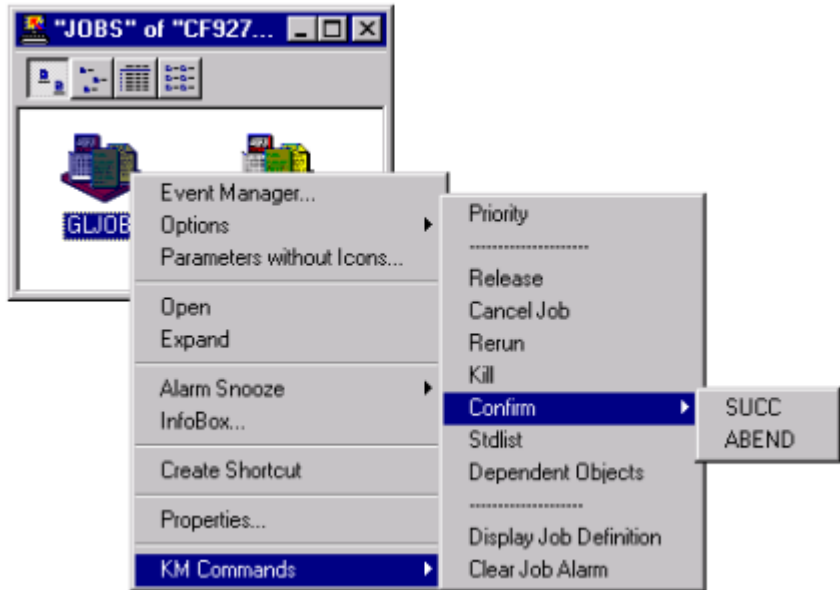
The TWS\_JOBS\_CONT application menu has the following menu items:

**Table 3-9 Menu Items for TWS\_JOBS\_CONT Application**

Menu	Action
Clear Job Alarms	Clears all the job alarms for the parameters TWSJobsStatus for the current schedule.

# TWS\_JOBS Application Menu

The TWS\_JOBS menu is shown in Figure 3-10



**Figure 3-10 TWS\_JOBS Application Menu**

The TWS\_JOBS application menu has the following menu items:

**Table 3-10 Menu Items for TWS\_JOBS Application**

Menu	Action
Priority	Changes the priority for the job. Only schedules and jobs with a priority higher than the CPU's Fence limit will be able to run. The priority of a job will not run at a higher priority than the schedule it is currently running in.
Release	Releases the current job from all of it's dependencies.
Cancel Job	Cancels the selected job. If a job is cancelled before it is launched, it will not launch. If cancelled after it is launched, the job continues to execute. If an executing job is cancelled and it completes in the ABEND state, no automatic job recovery steps are attempted. Jobs and schedules that are dependent on the cancelled job are released immediately.

<b>Menu</b>	<b>Action</b>
Rerun	Reruns the selected job. To rerun a job, it must be in the <b>succ</b> , <b>fail</b> or <b>abend</b> state. A rerun job is placed in the same schedule as the original job, and inherits the original's dependencies. If you rerun a repetitive ( <b>every</b> ) job, the rerun job is scheduled to run at the same rate as the original job.
Kill	Stops an executing job. On UNIX, this is accomplished with a UNIX <b>kill</b> command. The kill operation is not performed by Conman, but is executed by Jobman, so there may be a short delay. Killed jobs terminate in the <b>abend</b> state. Any jobs or schedules that are dependent on a killed job are not released. Killed jobs can be rerun.
<b>Confirm</b>	The sub-menu which contains the menus to acknowledge the status of a job.
SUCC	Confirm the completion of a job that was scheduled with the SUCC state.
ABEND	Confirm the completion of a job that was scheduled with the ABEND state.
Stdlist	The standard list files for the job selection are displayed in their standard format.
Dependent Objects	The jobs used in <b>follows</b> dependencies are listed followed by the dependent jobs and schedules. Jobs are listed in the basic <b>showjobs</b> format.
Display Job Definition	Display the contents of the job's script file.
Clear Job Alarm	Clears the alarm on the TWSJobsStatus parameter.

## Where to Go from Here

The following table suggests topics that you should read next:

<b>If you want information on...</b>	<b>See...</b>
How to load the TWS KM	Chapter 2, "Getting Started." and TWS KM help
What a certain parameter does	Chapter 4, "Parameter Summary," and TWS KM help.
How to perform a task using this KM	Chapter 5, "Monitoring TWS." and TWS KM help





---

# Parameter Summary

This chapter provides a summary of parameters for PATROL for Tivoli Workload Scheduler by OTL Software (also known as TWS KM). Refer to the *PATROL for Unix User Guide* and the *PATROL for Windows User Guide (Volume 3)* for additional information about the different types of parameters and their functions. Refer to the KM help system for details about KM-specific parameters. The following topics are discussed:

Functional Parameter Summary . . . . .	4-2
Parameter Default Values . . . . .	4-8
Where to Go from Here . . . . .	4-12

# Functional Parameter Summary

TWS KM has various parameters that provide statistical information about resources, operating status, and performance. Table 4-1 provides information that you can use when selecting or reviewing the appropriate parameters that are used in monitoring the KM.

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

Parameter	Description	See Also Page
<b>TWS_MAESTRO Application Class</b>		
TWScpuColl	This collector sets consumer parameters (TWSLinkStatus) in the TWS_CPU application class. The TWScpuColl collector collects information regarding the status of each defined CPU within TWS.	4-9
TWSPProcessColl	This collector sets consumer parameter (TWSPProcessStatus) in the TWS_PROCESS application class. The TWSPProcessColl collector collects information regarding the status of the four key TWS processes (netman, mailman, jobman, and batchman).	4-10
TWSPromptsColl	This collector sets consumer parameters in the TWS_PROMPTS application class. The TWSPromptsColl collector collects information regarding the status of prompts that match the prompts filter.	4-10
TWSSchedulesColl	This collector sets consumer parameters in the TWS_SCHEDULES and TWS_JOBS application classes. TWSSchedulesColl collects information regarding the status of all jobs and schedules in TWS. It is one of the primary collectors of the Knowledge Module, providing detailed information about jobs and schedules that match the filter requirements set by the PATROL Administrator. The status of each job and schedule is displayed with an application icon (TWS_JOBS_STATUS and TWS_SCHEDULES_STATUS) representing the current state visually. Important information such as the status of each job and schedule is set. Summary information for all jobs and schedules and for selected schedules is also displayed using the parameters in the TWS_SUMMARY application class.	4-10

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

Parameter	Description	See Also Page
<b>TWS_CPU Application Class</b>		
TWSLinkStatus	This consumer parameter, set by the TWSCpuColl collector, displays the link status of the CPU with the Master machine.	4-10
<b>TWS_PROCESS Application Class</b>		
TWSProcessStatus	This consumer parameter, set by the TWSPProcessColl collector, monitors the status of key TWS processes (netman, mailman, jobman, and batchman).	4-10
<b>TWS_PROMPTS Application Class</b>		
TWSPromptFull	This consumer parameter, set by the TWSPromptsColl collector, monitors any prompts that match the defined prompts filter and sets the value to be the full text details of the prompt.	4-10
TWSPromptNum	This consumer parameter, set by the TWSPromptsColl collector, monitors any prompts that match the defined prompts filter and sets the value to be the number of the prompt.	4-10
TWSPromptStatus	This consumer parameter, set by the TWSPromptsColl collector, monitors any prompts that match the defined prompts filter and sets the value to be the current status of the prompt (INACT, ASKED, YES, and NO).	4-10
TWSPromptText	This consumer parameter, set by the TWSPromptsColl collector, monitors any prompts that match the defined prompts filter and sets the value to be the text details of the prompt.	4-10
<b>TWS_SUMMARY Application Class</b>		
TWSABENDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the ABEND state. The job terminated with a non-zero exit code.	4-9
TWSABENDScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the ABEND state. The schedule terminated unsuccessfully.	4-9

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

<b>Parameter</b>	<b>Description</b>	<b>See Also Page</b>
TWSABENPJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the ABENP state. ABEND confirmation received, but the job has not completed.	4-9
TWSADDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the ADD state. The job is being submitted.	4-9
TWSADDScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the ADD state. The schedule has just been submitted.	4-9
TWSCANCLJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the CANCL state. The job has been cancelled.	4-9
TWSCANCLScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the CANCL state. The schedule has been cancelled.	4-9
TWSDONEJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the DONE state. The job has completed in an unknown state (MPE only).	4-9
TWSERRORJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the ERROR state. For an internetwork dependency , an error occured while checking for the remote status.	4-9
TWSEXECJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the EXEC state. The job is currently executing.	4-9
TWSEXECScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the EXEC state. The schedule is currently executing.	4-9
TWSEXTRNJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the EXTRN state. For an internetwork dependency, unknown status. An error occured, a <b>rerun</b> action was just performed on the EXTERNAL job, or the remote job or schedule does not exist.	4-9

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

<b>Parameter</b>	<b>Description</b>	<b>See Also Page</b>
TWSFAILJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the FAIL state. Unable to launch the job. This is typically if the user account does not exist, the script, or command, does not exist, the jobmanrc file cannot be run, the stdlist file could not be opened, or could not login as the user and change to the user's home directory.	4-9
TWSFENCEJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the FENCE state. The job's priority is below the fence of the CPU.	4-9
TWSHOLDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the HOLD state. The job is awaiting dependency resolution.	4-9
TWSHOLDScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the HOLD state. The schedule is awaiting dependency resolution.	4-9
TWSINTROJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the INTRO state. The job has been introduced for launching by the system.	4-9
TWSPENDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the PEND state. The job has completed and is awaiting confirmation of the exit state by the operator (state to be set is either SUCC or ABEND).	4-10
TWSREADYJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the READY state. The job is ready to launch, all of the dependencies have been resolved.	4-10
TWSREADYScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the READY state. All dependencies for the schedule have been resolved and is ready to launch.	4-10

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

<b>Parameter</b>	<b>Description</b>	<b>See Also Page</b>
TWSSCHEDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the SCHED state. The job's <b>at</b> time has not arrived.	4-10
TWSSSTUCKScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the STUCK state. The schedule's execution has been interrupted. No jobs will be launched without operator intervention.	4-11
TWSSUCCJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the SUCC state. The job has completed with a zero exit code.	4-11
TWSSUCCPJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the SUCCP state. SUCCP confirmation received, but the job has not completed.	4-11
TWSSUCCScheds	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of schedules that are in the SUCC state. The schedule has completed successfully.	4-11
TWSSUSPJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the SUSP state. The job has been suspended by a <b>breakjob</b> command (MPE only).	4-11
TWSWAITDJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the WAITD state. The job is in the WAIT state and is deferred (MPE only).	4-11
TWSWAITJobs	This consumer parameter, set by the TWSSchedulesColl collector, displays the total number of jobs that are in the WAIT state. The job is being moved to READY (X-agent jobs and MPE only).	4-11

**Table 4-1 PATROL for Tivoli Workload Scheduler Parameter Summary**

Parameter	Description	See Also Page
<b>TWS_SCHEDULES Application Class</b>		
TWSSchedulesDelayTime	This consumer parameter, set by the TWSSchedulesColl collector, displays the delay time of the schedule start. This is derived from the estimated start time and the actual start time of the schedule.	4-7
TWSSchedulesElapsedTime	This consumer parameter, set by the TWSSchedulesColl collector, displays the elapsed time since the actual start time.	4-7
TWSSchedulesStatus	This consumer parameter, set by the TWSSchedulesColl collector, displays the status of the current schedule. The valid states are as follows: 0=ALARM CLEARED, 1=ADD, 2=HOLD, 3=READY, 4=EXEC, 5=SUCC, 6=CANCL, 7=JOB FAIL, 8=STUCK, 9=ABEND. The parameter will go into a warning, if the schedule is in the JOB FAIL state. JOB FAIL is not an internal Maestro state, but a special state within the KM, for when a job within a schedule fails, but the recovery actions for the job are such that the schedule has a SUCC state in Maestro. The parameter will alarm if the schedule is in the STUCK or ABEND state.	4-11
<b>TWS_JOBS Application Class</b>		
TWSJobsDelayTime	This consumer parameter, set by the TWSSchedulesColl collector, displays the delay time of the job start. This is derived from the estimated start time and the actual start time of the job.	4-7
TWSJobsElapsedTime	This consumer parameter, set by the TWSSchedulesColl collector, displays the elapsed time since the actual start time.	4-7
TWSJobsStatus	This consumer parameter, set by the TWSSchedulesColl collector, displays the status of the current job. The valid states are as follows: 0=ALARM CLEARED, 1=ADD, 2=INTRO, 3=READY, 4=EXEC, 5=SUCC, 6=SUCCP, 7=SUSP, 8=WAIT, 9=WAITD, 10=FENCE, 11=HOLD, 12=PEND, 13=SCHED, 14=CANCL, 15=ABEND, 16=ABENP, 17=DONE, 18=ERROR, 19=EXTRN, 20=FAIL. The parameter will alarm if the schedule is in the ABEND through to FAIL states.	4-10

# Parameter Default Values

Table 4-2 lists default values for parameters. Interpret the column headings as follows. Depending on the type of parameter, some information is not applicable, denoted by N/A in the table.

<b>Parameter</b>	Specifies the parameter name.
<b>Active?</b>	Specifies whether the parameter is active or inactive when discovered.
<b>Type</b>	Specifies whether the parameter is a Standard (Std.), Consumer (Con.), or Collector (Coll.) parameter.
<b>Alarm 1</b>	Specifies the thresholds for the first alarm. This information is not applicable to Collectors.
<b>Alarm 2</b>	Specifies the thresholds for the second alarm. This information is not applicable to Collectors.
<b>Scheduling</b>	Specifies the time interval in the poll cycle. This information is not applicable to Consumers.
<b>Icon</b>	Specifies whether the icon is a graph, gauge, or text box.
<b>Units</b>	Specifies the type of unit in which the parameter output is expressed, such as a percentage, a number, or bytes.
<b>History Level</b>	Specifies the history retentions period. This information is not applicable to Collectors.
<b>See Also Page</b>	Specifies other pages in this chapter where you can find more functional information about the parameter.



**Table 4-2 TWS KM Parameter Default Values**

<b>Parameter/Description</b>	<b>Active?</b>	<b>Type</b>	<b>Alarm 1</b>	<b>Alarm 2</b>	<b>Scheduling</b>	<b>Icon</b>	<b>Units</b>	<b>History Level</b>	<b>See Also Page</b>
TWSABENDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSABENDScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSABENPJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSADDDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSADDScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSCANCLJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSCANCLScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSCpuColl	Y	Coll	N/A	N/A	10 min.	N/A	N/A	Inherited	4-9
TWSDONEJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSERRORJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSEXECJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSEXECScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSEXTRNJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSFAILJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSFENCEJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSHOLDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSHOLDScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSINTROJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-9
TWSJobsDelayTime	Y	Con	N/A	N/A	10 min.	Graph	Minutes (-1 = Not Executed; -2 = Unknown)	Inherited	4-7

**Table 4-2 TWS KM Parameter Default Values**

<b>Parameter/Description</b>	<b>Active?</b>	<b>Type</b>	<b>Alarm 1</b>	<b>Alarm 2</b>	<b>Scheduling</b>	<b>Icon</b>	<b>Units</b>	<b>History Level</b>	<b>See Also Page</b>
TWSJobsElapsedTime	Y	Con	N/A	N/A	10 min.	Graph	Minutes (-1 = Not Executed; -2 = Unknown)	Inherited	4-7
TWSJobsStatus		Con	15 - 20	N/A	10 min.	Boolean	0-14 = Acceptable 15-20 = Alert Event	Inherited	4-10
TWSLinkStatus	Y	Con	0 - 0	1 - 1	10 min.	Boolean	2=LINKED, 1=JNEXTDAY RUNNING, 0=UNLINKED	Inherited	4-10
TWSPENDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-10
TWSPProcessColl	Y	Coll	N/A	N/A	10 min.	N/A	N/A	Inherited	4-10
TWSPProcessStatus	Y	Con	0 - 0	1 - 1	10 min.	Boolean	2 = yes, 1=Jnextday running, 0 = no	Inherited	4-10
TWSPromptsColl	Y	Coll	N/A	N/A	10 min.	N/A	N/A	Inherited	4-10
TWSPromptFull	Y	Con	N/A	N/A	10 min.	Text	N/A	Inherited	4-10
TWSPromptNum	Y	Con	N/A	N/A	10 min.	Text	N/A	Inherited	4-10
TWSPromptStatus	Y	Con	3 - 3	N/A	10 min.	Boolean	0=INACT, 1=NO, 2=YES, 3=ASKED	Inherited	4-10
TWSPromptText	Y	Con	N/A	N/A	10 min.	Text	N/A	Inherited	4-10
TWSREADYJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-10
TWSREADYScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-10
TWSSCHEDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-10
TWSSchedulesColl	Y	Coll	N/A	N/A	10 min.	N/A	N/A	Inherited	4-10

**Table 4-2 TWS KM Parameter Default Values**

<b>Parameter/Description</b>	<b>Active?</b>	<b>Type</b>	<b>Alarm 1</b>	<b>Alarm 2</b>	<b>Scheduling</b>	<b>Icon</b>	<b>Units</b>	<b>History Level</b>	<b>See Also Page</b>
TWSSchedulesDelayTime	Y	Con	N/A	N/A	10 min.	Graph	Minutes (-1 = Not Executed; -2 = Unknown)	Inherited	4-7
TWSSchedulesElapsedTime	Y	Con	N/A	N/A	10 min.	Graph	Minutes (-1 = Not Executed; -2 = Unknown)	Inherited	4-7
TWSSchedulesStatus		Con	6 - 7	8 - 9	10 min.	Boolean	0=ALARM CLEARED, 1=ADD,2=HOLD,3=READY,4=EXEC,5=SUCCESS,6=CANCEL,7=JOB FAIL,8=STUCK,9=ABEND		4-11
TWSSSTUCKScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSUCCJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSUCCPJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSUCCScheds	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSUSPJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSWAITDJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11
TWSSWAITJobs	Y	Con	N/A	N/A	10 min.	Graph	#	Inherited	4-11

## Where to Go from Here

The following table suggests topics that you should read next.

<b>If you want information on...</b>	<b>See...</b>
How to load TWS KM	Chapter 2, "Getting Started." and TWS KM help
What a certain menu command does	Chapter 3, "Menu Summary" and TWS KM help.
How to perform a task using this KM	Chapter 5, "Monitoring TWS." and TWS KM help

---

# Monitoring TWS

This chapter introduces you to basic PATROL for Tivoli Workload Scheduler by OTL Software (also referred to as TWS KM) tasks. The following topics are discussed:

Overview .....	5-2
Objectives of the KM .....	5-2
Monitoring Server Availability .....	5-2
Monitoring CPU Availability .....	5-3
Schedule, Job, and Prompt Filters .....	5-3
Schedule Filter .....	5-4
Job Filter .....	5-5
Prompt Filter .....	5-6
Administering TWS KM .....	5-7
Starting the NETMAN Process .....	5-7
Stopping the Process Tree .....	5-8
Starting the Process Tree .....	5-9
Stopping the Production Processes .....	5-10
Launching Administration Utilities .....	5-11
Reporting from TWS KM .....	5-12
Tivoli Workload Scheduler Database Validation .....	5-13
Debugging TWS KM .....	5-14
Refreshing Parameters .....	5-15
Displaying a Parameter Graph, Gauge, or Text Output Window . . .	5-16
Customizing Parameters .....	5-16
Unloading the KM .....	5-17
Uninstalling the KM .....	5-20
Where to Go from Here .....	5-25

## Overview

After loading TWS KM, you can use default parameter settings to monitor the KM, or you can customize parameter settings to meet the demands of your environment. If the KM has not been loaded, refer to the Chapter 2, “Getting Started,” for procedures on loading the KM.

Objectives of TWS KM are discussed in “Objectives of the KM” on page 5-2. This information will assist you in achieving maximum availability, performance, and integrity of your scheduling environment.

## Objectives of the KM

The primary objective of TWS KM is to ensure the availability, maximum schedule/job throughput performance, and integrity of your scheduling environment. This section describes how to use TWS KM to achieve these goals. A PATROL Developer or Operator Console can be used, as required, to change any of the default behaviors listed below.

The KM also provides specific monitoring of the status of prompts, jobs and schedules to ensure that any problems are detected and reported.

## Monitoring Server Availability

TWS KM monitors the TWS Master server processes to ensure availability (i.e. The processes netman, batchman, jobman, and mailman are running). The parameter TWSServerStatus monitors availability of all of the above processes. In the event of any process not running the parameter will go into an alarm state. As a result of this several recovery actions are initiated; these being a window appearing on the PATROL Console indicating the status of all five key processes, and the same information recorded as a parameter annotation point.

## Monitoring CPU Availability

TWS KM monitors the link status of all the CPU's as defined within TWS. If the CPU becomes unlinked for more than two collection cycles (allowing for Jnextday to run each day) then an alarm is raised. Each CPU can be managed within the Knowledge Module, with the correct login details provided by the PATROL operator, allowing the following actions; link, unlink, start, stop.

## Schedule, Job, and Prompt Filters

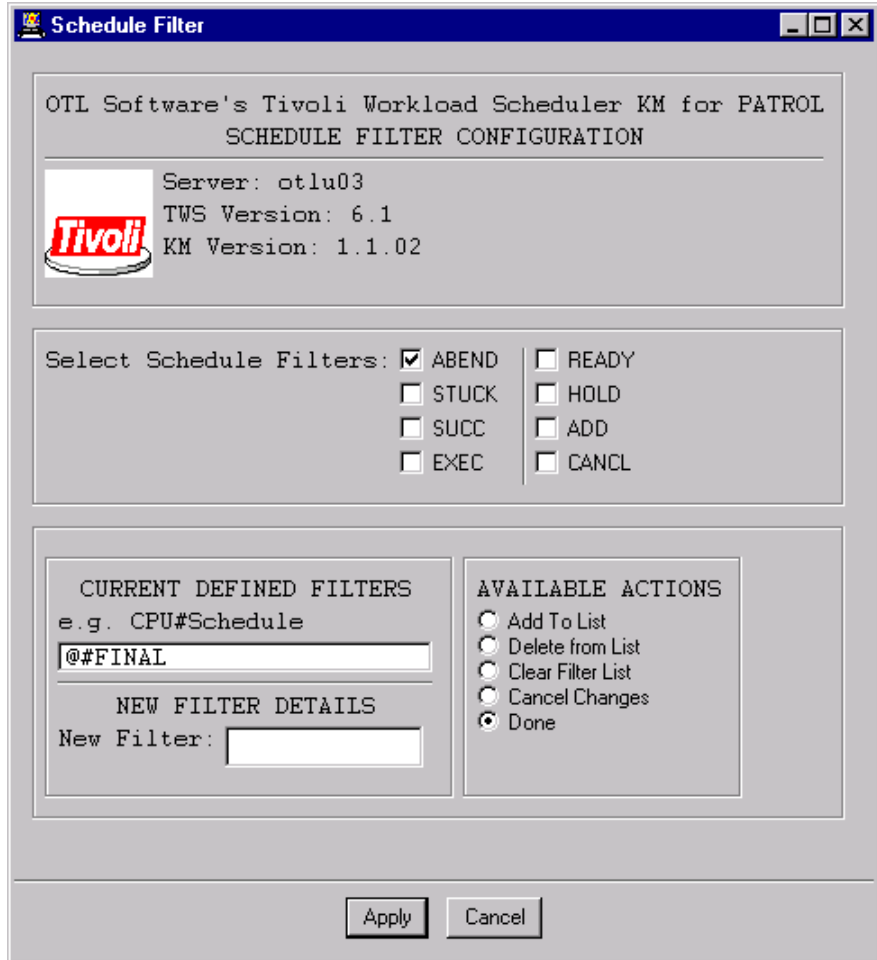
Filters are used in three key areas of the TWS KM; schedules, jobs, and prompts. Filters allow for customisation of what is monitored by the Knowledge Module. By default all schedules, jobs, and prompts that are important are monitored. The schedule and job filters will accept the standard wildcards used in TWS - @, %, and ?.

A reference to one or more Maestro entities must be in the following syntax:

```
CPU query:      <Domain>!<CPU>
Schedule query: <CPU>#<Schedule>
                <Domain>!<CPU>#<Schedule>
Job query:      <Schedule>.<Job>
                <CPU>#<Schedule>.<Job>
                <Domain>!<CPU>#<Schedule>.<Job>
```

## Schedule Filter

The schedule filter allows for monitoring of schedules in specific states and also specific schedules that are specified (either explicitly or using wildcards).



**Figure 5-1 Schedule Filter Window**

By default, only schedules that are in the STUCK and ABEND states, and the FINAL schedule itself are monitored as shown above.



# Job Filter

The job filter allows for monitoring of jobs in specific states and also specific jobs that are specified (either explicitly or using wildcards). The jobs depend on the schedule existing.

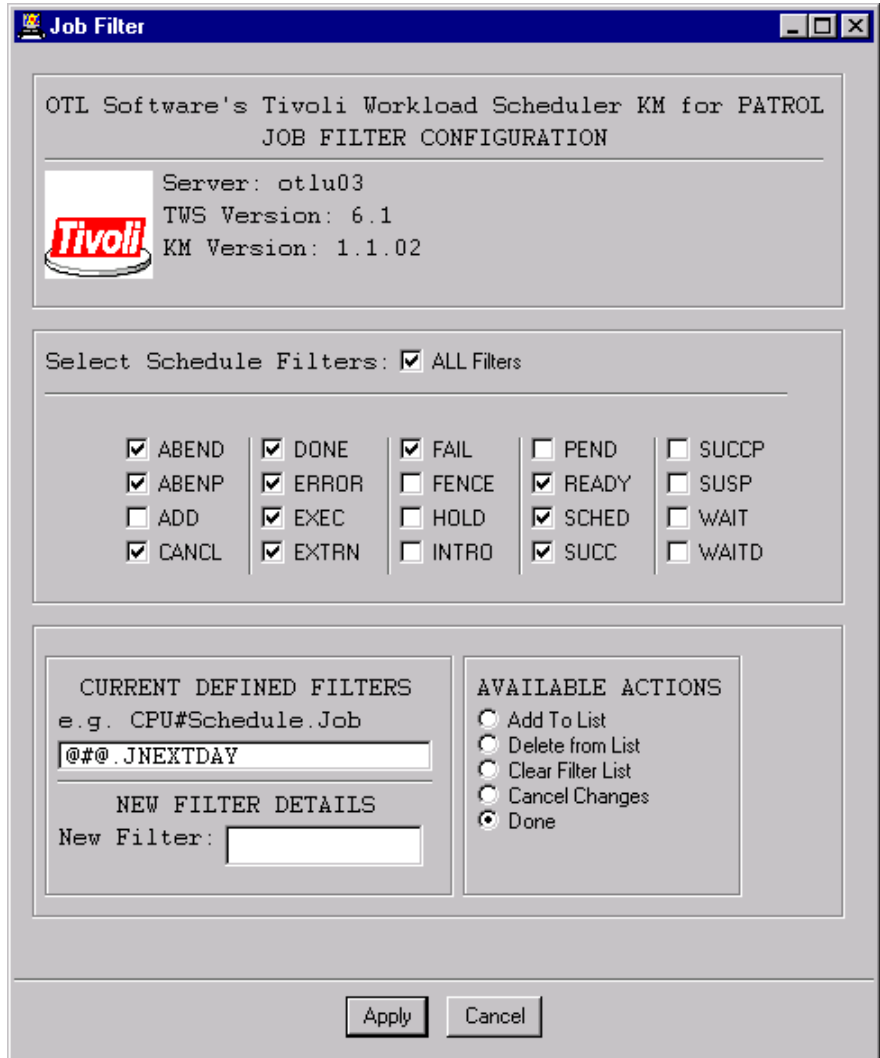
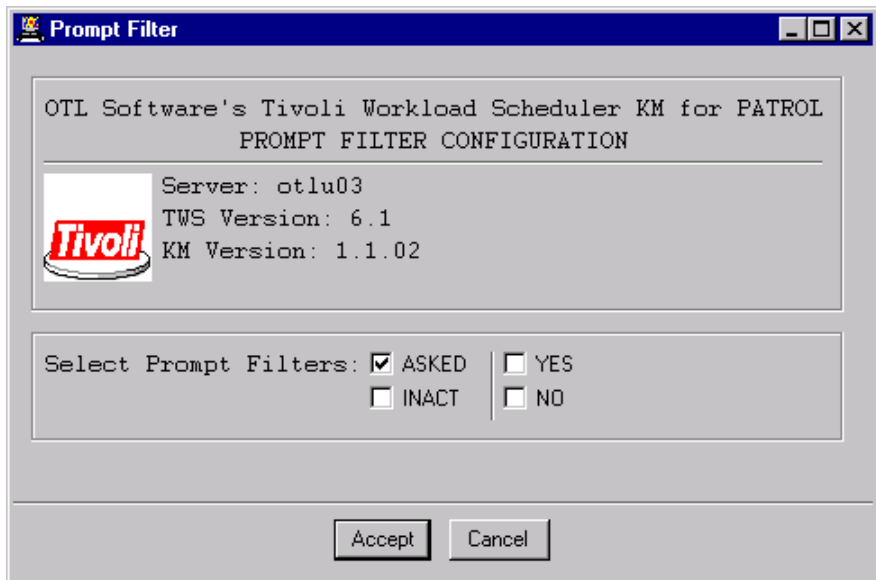


Figure 5-2 Job Filter Window

By default, only jobs that are in the ABEND, ABENP, DONE, ERROR, EXTRN, and FAIL states, and the JNEXTDAY job itself are monitored as shown above.

## Prompt Filter

The prompt filter allows for monitoring of prompts in specific states.



**Figure 5-3 Prompts Filtler Window**

By default, prompts in the ASKED state are monitored as shown above.

# Administering TWS KM

This section describes how to start and stop the TWS production processes. This is broken up into the following sections:

- Starting the TWS NETMAN process
- Stopping the TWS process tree
- Starting the TWS process tree
- Stopping the TWS production processes

## Starting the NETMAN Process

**Summary:** This task explains how to start the TWS netman process automatically on the server. This is equivalent of issuing a StartUp command.

---

### Before You Begin

When the discovery cycle is complete, verify that the application icon appears for the Tivoli Workload Scheduler instance. If the icon does not appear there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help. Alternately the discovery cycle may not have been able to find your installation of TWS.

### To Start the NETMAN process

**Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

**Step 2** Choose **Start/Stop Maestro => Start NETMAN (StartUp)**.

The confirmation dialog box appears.

**Step 3** Click **yes** to proceed.

The login and password dialog box appears.

- Step 4** Enter the login details of an account that has enough privileges to start the TWS netman process.

**If a Problem Occurs**

1. Ensure that you have properly installed the KM. Refer to the *PATROL Installation Guide* for help.
2. Verify the configuration settings according to Chapter 2, “Getting Started”.
3. Verify the login and password used.

## Stopping the Process Tree

*Summary:* This task explains how to stop all of the TWS processes automatically on the server. This is equivalent of issuing a conman shutdown command.

---

### Before You Begin

When the discovery cycle is complete, verify that the application icon appears for the Tivoli Workload Scheduler instance. If the icon does not appear there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help. Alternately the discovery cycle may not have been able to find your installation of TWS.

### To Stop all of the Processes

- Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

- Step 2** Choose **Start/Stop Maestro => Stop Processes (conman shutdown)**.

The confirmation dialog box appears.

- Step 3** Click **yes** to proceed.

The login and password dialog box appears.

- Step 4** Enter the login details of an account that has enough privileges to stop the TWS processes

A dialog box will appear with the results of stopping the processes.

#### **If a Problem Occurs**

1. Ensure that you have properly installed the KM. Refer to the *PATROL Installation Guide* for help.
2. Verify the configuration settings according to Chapter 2, “Getting Started”.
3. Verify the login and password used.

## **Starting the Process Tree**

**Summary:** This task explains how to start all of the TWS processes automatically on the server. This is equivalent of issuing a `conman start` command.

---

### **Before You Begin**

When the discovery cycle is complete, verify that the application icon appears for the Tivoli Workload Scheduler instance. If the icon does not appear there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help. Alternately the discovery cycle may not have been able to find your installation of TWS.

### **To Start all of the Processes**

- Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

- Step 2** Choose **Start/Stop Maestro => Start Processes (conman start)**.

The confirmation dialog box appears.

**Step 3** Click **yes** to proceed.

The login and password dialog box appears.

**Step 4** Enter the login details of an account that has enough privileges to start the TWS processes

A dialog box will appear with the results of starting the processes.

#### **If a Problem Occurs**

1. Ensure that you have properly installed the KM. Refer to the *PATROL Installation Guide* for help.
2. Verify the configuration settings according to Chapter 2, “Getting Started”.
3. Verify the login and password used.

## **Stopping the Production Processes**

**Summary:** This task explains how to stop the TWS processes automatically on the server, leaving the netman process running. This is equivalent of issuing a conman stop command.

---

### **Before You Begin**

When the discovery cycle is complete, verify that the application icon appears for the Tivoli Workload Scheduler instance. If the icon does not appear there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help. Alternately the discovery cycle may not have been able to find your installation of TWS.

## To Stop the Production Processes

**Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

**Step 2** Choose **Start/Stop Maestro => Stop Processes (conman stop)**.

The confirmation dialog box appears.

**Step 3** Click **yes** to proceed.

The login and password dialog box appears.

**Step 4** Enter the login details of an account that has enough privileges to stop the TWS processes

A dialog box will appear with the results of stopping the processes.

### If a Problem Occurs

1. Ensure that you have properly installed the KM. Refer to the *PATROL Installation Guide* for help.
2. Verify the configuration settings according to Chapter 2, “Getting Started”.
3. Verify the login and password used.

## Launching Administration Utilities

This section describes the X-Window administration functions that are accessible through PATROL for Tivoli Workload Scheduler (UNIX only). These utilities require the TWS master server is running UNIX and the PATROL Console machine is running X-Windows. These are as follows:

- GMAESTRO - GUI window to launch gcomposer and gconman.

- GCOMPOSER - create scheduling objects using the GUI.
- GCONMAN - manage production environment using the GUI.
- COMPOSER - text utility for creating scheduling objects.
- CONMAN - text utility for managing production environment.

## Reporting from TWS KM

This section describes the main reporting functions provided by PATROL for Tivoli Workload Scheduler. The reports available are:

- **List Symphony Log Files** - Reports on available Symphony Log Files that are currently available on the server.
- **Show All Schedules** - Reports on all schedules that are running in the current production day.
- **Show All Jobs** - Reports on all jobs that are running in the current production day.
- **Show All Resources** - Reports on all resource dependencies that are defined against the current production day.
- **Show All Files** - Reports on all file dependencies that are defined against the current production day.
- **Show All Prompts** - Reports on all prompts that have, or will be issued in the current production day.
- **Job Details Listing (rep1)** - Runs the standard TWS report rep1.
- **Prompt Messages Listing (rep2)** - Runs the standard TWS report rep2.
- **User Calendar Listing (rep3)** - Runs the standard TWS report rep3.
- **User Parameters Listing (rep4a)** - Runs the standard TWS report rep4a.
- **Resource Listing (rep4b)** - Runs the standard TWS report rep4b.
- **Job History Listing (rep7)** - Runs the standard TWS report rep7.
- **Job Histogram (rep8)** - Runs the standard TWS report rep8.
- **Planned Prod Schedule (rep11)** - Runs the standard TWS report rep11.



# Tivoli Workload Scheduler Database Validation

This section describes the method for performing a database validation against each database item. Essentially, each command will run a composer “create <file> from <database item>” and then runs a composer “validate <file>” from the data that has been dumped. The validation checks the integrity of the database objects and will report on any problems.:

## To Validate TWS Database

- Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

- Step 2** Choose **Validate Configurations => Calendars (or another database item)**.

The login and password dialog box appears.

- Step 3** Enter the login details of an account that has enough privileges to run the validate command.

A dialog box will appear with the results of the validation.

# Debugging TWS KM

**Summary:** This task explains how to set debugging information for TWS KM.

---

## Before You Begin

Verify that the application icon for the loaded application instance is displayed. If the icon does not appear, there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help.

## To Enable or Disable Debugging

**Step 1** Right-click and hold the mouse pointer on the Tivoli Workload Scheduler application instance icon.

The application menu appears.

**Step 2** Choose **Debug**.

The Debug Admin dialog box appears.

**Step 3** Click on the areas that debugging information is required:

**Step 4** Click **OK**.

Debugging information will appear on the system output window for the computer.

# Refreshing Parameters

**Summary:** This task explains how to refresh all parameters for an application instance.

---

## Before You Begin

Verify that the application icon for the loaded application instance is displayed. If the icon does not appear, there may be a problem with your PATROL installation. Refer to the *PATROL Installation Guide* for help.

## To Refresh All Parameters

**Step 1** Right-click and hold the mouse pointer on the MAESTRO instance icon.

The application menu appears.

**Step 2** Choose **Refresh Parameters**.

All the parameters are updated for all application classes.

## Displaying a Parameter Graph, Gauge, or Text Output Window

Each computer icon in the PATROL main window represents an instance of a host system that PATROL is monitoring. When you add a computer to the main window, PATROL establishes a default series of application and parameter icons for monitoring. For information on displaying a parameter graph, gauge, or text output window, see the *PATROL for Unix User Guide* or the *PATROL for Windows NT User Guide (Volume 2)*.

## Customizing Parameters

Most parameters defined in a KM are activated by default. They continuously monitor key resources and warn you of potential problems. All parameters in KMs are global parameters; that is, they automatically run on all KM instances discovered. They are the common parameters used for all applications and computers. You can customize these parameters at the local level for a specific application or computer. You can customize parameters at the local level by performing some of the tasks listed below. For information on these tasks, see the *PATROL for Unix User Guide* or the *PATROL for Windows User Guide (Volume 3)*.

- Activating Help
- Adding Parameters
- Clearing Parameter History
- Deleting Parameters
- Entering the Parameter Command
- Entering the Parameter Environment
- Scheduling When the Parameter Runs
- Selecting the Parameter Computer Class
- Selecting the Parameter Command Type
- Selecting the Parameter Type
- Setting Alarm Ranges
- Setting Parameter Security
- Setting the Parameter History Retention Level
- Setting the Parameter Output
- Setting the State
- Suspending Parameters

# Unloading the KM

This section describes the procedures for unloading (not uninstalling) the TWS KM from the PATROL Agent, PATROL Console or PATROL Central Console. The intention is that this section is only used if you no longer want to use *some parts* of the KM, though there may be circumstances where you may wish to fully unload the KM without uninstalling the files.

---

## Note

---

If you want to completely uninstall the KM, please follow the instructions under “Uninstalling the KM” on page 5-20.

---

When the TWS KM is unloaded from a PATROL Agent, PATROL stops monitoring the TWS application on that system when there is no connection to a PATROL Console with TWS KM loaded.

When the TWS KM is unloaded from PATROL Console, the PATROL Console stops displaying and monitoring the Tivoli Workload Scheduler application on *any* PATROL Console connected system.

## Unloading the KM from PATROL Agent

1. Remove the TWS KM from the list of preloaded KMs, using the utility `wconfig` (on MS Windows) or `xpconfig` (on Unix), remove **TWS\_LOAD.kml** from the PATROL Agent configuration variable “/AgentSetup/preloadedKMs”.
2. Restart the PATROL Agent.
3. Repeat these steps on every PATROL Agent system (managed node) where the TWS KM is to be unloaded.

---

## Note

---

The TWS KM cannot be *partially* unloaded from the PATROL Agent using the above steps. If you need to unload some parts of the KM from the PATROL Agent, refer to the *PATROL Agent Reference Manual*.

---

## Unloading the KM from PATROL Console

Application classes are unloaded individually. This allows for a partial unloading of the KM to enable it to run as a reduced installation. Follow the steps below to unload the TWS KM completely or partially.

1. Remove the unwanted TWS KM application classes from the list of loaded application classes, as required. All TWS KM application classes start with “**TWS\_**”.

### With the PATROL Console for Unix:

- A. From the PATROL Console main window choose **Attributes => Application Classes...**
- B. Select a TWS KM application class to be removed and choose **Edit => Delete**.
- C. Repeat for all classes to be removed.

### With the PATROL Console for MS Windows:

- A. From the PATROL Console tree view choose the **KM** tab and expand the folder **Application Classes**.
  - B. Right-click on a TWS KM application class to be removed and choose **Delete**.
  - C. Repeat for all classes to be removed.
2. Select **File => Save Configuration** to save the modified list of loaded application classes as the PATROL Console user preference.
  3. Repeat these steps on every PATROL Console system where the TWS KM is to be unloaded.

## Unloading the KM from PATROL Central Console

A KM can be unloaded for particular managed systems or for all systems across the monitored environment. Application classes may be unloaded individually, or by selecting all classes at once. This allows for a partial unloading of the KM to enable it to run as a reduced installation on some or all of the managed systems. Follow the steps below to completely or partially unload the TWS KM from some or all of the managed systems.

1. Right click on the **PATROL Main Map**, and choose **Unload Knowledge Modules...**
2. Select the managed systems where the TWS KM is to be unloaded, and click **Next>**.

A list is displayed showing all the loaded application classes on each of the selected managed systems. All TWS KM application classes start with “**TWS\_**”.

3. Select the TWS KM application classes to be removed for the appropriate managed systems, click **Next>** and **Finish**.
4. Repeat the above steps for each PATROL Central Management Profile where the TWS KM is to be unloaded.

## Uninstalling the KM

This section describes steps for uninstalling the PATROL for Tivoli Workload Scheduler. The KM is uninstalled when upgrading the TWS KM from an older version, or if the TWS KM is no longer required for monitoring the Tivoli Workload Scheduler application. The steps required for uninstalling depend on the type of installation:

- For PATROL Agent and PATROL Console, “uninstallation” involves unloading the KM and then removing the files.
- For PATROL Central Console, the KM is uninstalled by unloading.
- For PATROL Central Console Server or PATROL Central Web Server, the KM is uninstalled by removing the files.

---

### Note

---

If the TWS KM was installed using the *Installation Utility*, use the same to remove the KM files after unloading.

---

## Uninstalling the KM from PATROL Agent

1. Unload the TWS KM by removing it from the list of preloaded KMs. Use the utility `wconfig` (on MS Windows) or `xpconfig` (on Unix) to remove **TWS\_LOAD.kml** from the PATROL Agent configuration variable “/AgentSetup/preloadedKMs”.
2. Restart the PATROL Agent.
3. Remove all TWS KM files listed in Table 5-1 under the paths for PATROL Home (**PATROL\_HOME**) and PATROL Cache (**PATROL\_CACHE**).

---

### Note

---

There may be more than one PATROL Cache directory depending on how PATROL has been set up in your installation. Users can set up a local Cache directory to override the global setting.

---



- Repeat the above steps on every PATROL Agent system where the TWS KM is installed.

**Table 5-1 Uninstallation from the PATROL Agent**

File Types to Delete	Path relative to PATROL_HOME	
	Unix	Microsoft Windows
PSL Library Files	lib/psl/TWS_*.*	lib\psl\TWS_*.*
KM & Catalog Files	lib/knowledge/TWS_*.*	lib\knowledge\TWS_*.*
Archive Files	lib/archive/TWS_*.*	lib\archive\TWS_*.*
Other Files & Folders	TWS/* TWS TWS_*.*	TWS\*.*
	lib/TWS/*	lib\TWS\*.*
	lib/TWS	lib\TWS
	lib/TWS_*.*	lib\TWS_*.*

## Uninstalling the KM from PATROL Console

- Unload the TWS KM by removing all the application classes from the list of loaded application classes. All TWS KM application classes start with “TWS\_”.

### On PATROL Console for Unix:

- From the PATROL Console main window choose **Attributes => Application Classes...**
- Select a TWS KM application class and choose **Edit => Delete**.
- Repeat for all classes.
- Select **File => Save Configuration** to save the modified list of loaded KMs as the PATROL Console user preference.

### On PATROL Console for Microsoft Windows:

- From the PATROL Console tree view choose the **KM** tab and expand the folder Application Classes.
- Right-click on a TWS KM application class and choose **Delete**.
- Repeat for all classes.

- D. Select **File => Save Configuration** to save the modified list of loaded KMs as the PATROL Console user preference.
2. Remove all TWS KM files listed in Table 5-2 under the paths for PATROL Home (**PATROL\_HOME**) and PATROL Cache (**PATROL\_CACHE**).

---

**Note**

---

There may be more than one PATROL Cache directory depending on how PATROL has been setup in your installation. Users can set up a local Cache directory to override the global setting.

---

**Table 5-2 Uninstallation from PATROL Console**

File Types to Delete	Path relative to PATROL_HOME	
	Unix	Microsoft Windows
PSL Library Files	<b>lib/psl/TWS_*.*</b>	<b>lib\psl\TWS_*.*</b>
KM & Catalog Files	<b>lib/knowledge/TWS_*.*</b>	<b>lib\knowledge\TWS_*.*</b>
Archive Files	<b>lib/archive/TWS_*.*</b>	<b>lib\archive\TWS_*.*</b>
Icon & Image Files	<b>lib/images/TWS_*.*</b> <b>lib/images/tws_*.*</b>	<b>lib\images\tws_*.*</b>
Online Help Files & Folders	<b>lib/help/tws_*.*</b> <b>lib/help/tws_km/*.*</b> <b>lib/help/tws_km</b> <b>lib/help/km_help_tws_km</b>	<b>lib\help\tws_*.*</b> <b>lib\help\tws_km/*.*</b> <b>lib\help\tws_km</b> <b>lib\help\km_help_tws_km</b>

3. Repeat above steps on every PATROL Console system where the TWS KM is installed.

## Uninstalling the KM from PATROL Central Console

1. Unload the TWS KM by removing all the application classes from the list of loaded application classes. All TWS KM application classes start with “**TWS\_**”.
  - A. Right-click on the **PATROL Main Map** and choose **Unload Knowledge Modules...**
  - B. Select the managed systems where the TWS KM is to be unloaded, and click **Next>**.
  - C. Select all TWS KM application classes, click **Next>** and **Finish**.
2. Repeat the above steps on every PATROL Central Console where the TWS KM is installed.

## Uninstalling the KM from PATROL Central Console Server

1. Remove all TWS KM files listed in Table 5-3 under the PATROL Central Console Server installation path (**PATROL\_ROOT**).

**Table 5-3 Uninstallation from the PATROL Central Console Server**

File Types to Delete	Path for PATROL Central Console Server	
	Unix	Microsoft Windows
Online Help Files	lib/knowledge/tws_*/lib/help /EN_USA/tws_km.chm	lib\knowledge\tws_*\lib\help \EN_USA\tws_km.chm
Icon & Image Files & Folders	lib/knowledge/tws_*/*. * lib/knowledge/tws_*	lib\knowledge\tws_*\*. * lib\knowledge\tws_*

2. Repeat above steps on every PATROL Central Console Server system where the TWS KM is installed.

## Uninstalling the KM from PATROL Central Web Server

1. Remove all TWS KM files listed in Table 5-4 under the PATROL Central Web Server installation path (**\$BMC\_ROOT/webcentral** on Unix and **%BMC\_ROOT%\WebCentral** on Microsoft Windows).

**Table 5-4 Uninstallation from the PATROL Central Web Server**

File Types to Delete	Path for PATROL Central Web Server	
	Unix	Microsoft Windows
Online Help Files	help_services/tws_*.jar km_services/html/default/lib/help/EN_USA/tws_*.jar	help_services\tws_*.jar km_services\html\default\lib\help\EN_USA\tws_*.jar

2. Repeat above steps on every PATROL Central Web Server system where the TWS KM is installed.

## Uninstalling the PAR File from the BPPM Portal

---

### Note

If you are upgrading the PAR file on BMC Portal, skip the uninstallation procedures outlined below and follow the upgrade steps in “Installing or Upgrading the PAR file on BPPM Portal” on page 2-7.

---

1. Remove all elements using the TWS KM PAR file from the BMC ProactiveNet Performance Management Portal Infrastructure.
2. Log on to the BMC ProactiveNet Performance Management Portal with portal credentials, and select the **Portal** tab.
3. Under **Tasks** in the navigation pane, select **Performance Managers**.
4. Check the check-box next to the TWS KM solution from the list.
5. Click **Delete**.

# Deleting PATROL Agent Configuration Variables

1. Remove all PATROL Agent configuration variables created by the TWS KM. These variables are stored under configuration paths **TWS** and **TWS\_License**. They can be removed using `wconfig` (on MS Windows) or `xpconfig` (on Unix). Alternatively, you can use the following single line PSL command through the PATROL Console OS> prompt to remove TWS KM configuration variables. The TWS KM should be uninstalled on the PATROL Console before attempting the following PSL command:

```
%PSL foreach var(grep("^/TWS[/_]",pconfig("LIST")))
    { pconfig("DELETE", var); }
```

2. Repeat the above step on every PATROL Agent system where the TWS KM has been loaded.

## Where to Go from Here

The following table summarizes where to look for more information on using PATROL and PATROL for Tivoli Workload Scheduler by OTL Software. The shaded rows indicate tasks that you can accomplish only from a PATROL Developer Console.

<b>If you want information on...</b>	<b>See...</b>
TWS KM parameters	Chapter 4, "Parameter Summary," and TWS KM help.
TWS KM applications	TWS KM help.
TWS KM menu commands	Chapter 3, "Menu Summary," and TWS KM help.
TWS KM InfoBoxes	Chapter 1, "Introduction," and TWS KM help.
KMs in general	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1)
KM versioning and customizations	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 3).

<b>If you want information on...</b>	<b>See...</b>
the PATROL interface	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1).
managing events	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 2) and the PATROL Event Manager Console for Unix User Guide.
the PATROL Script Language (PSL)	the PATROL Script Language Reference Manual.
defining your monitoring environment	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 1).
adding computers to PATROL	the PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 1).
working with parameters	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 2).
working with menu commands	the PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 2).
working with tasks	the PATROL for Unix Getting Started or the PATROL for Windows User Guide (Volume 2).
managing monitored objects	the PATROL User Guide or the PATROL for Windows User Guide (Volume 2).
unloading the KM	the PATROL for Unix User Guide or the PATROL for Windows User Guide (Volume 2).

---

## Index

### A

Accessing Application Menus 3-2  
adding computers to PATROL 5-26  
Administering TWS KM 5-7  
Administration Utilities 5-11  
Agent Re-start 2-12  
Applications and Icons 1-8

### B

BPPM versions 1-4

### C

Components 1-6  
Console Re-start 2-12  
Customizing Parameters 5-16

### D

Debugging a Server Application Instance  
5-14

defining your monitoring environment 5-26  
Discovery Cycle 2-13  
Disk Usage 1-5  
Displaying a Parameter Graph, Gauge, or  
Text Output Window 5-16  
distribution file for installation utility,  
contents 2-3  
distribution file, contents 2-4  
distribution server files 2-3

### F

Features 1-2  
Functional Parameter Summary 4-2

### G

Getting Started 2-1

### H

Help 2-13  
Hierarchical Structure 1-11

## I

- InfoBoxes 1-20
- installation
  - BPPM Portal 2-3
  - files 2-3, 2-4
  - MS Windows platform 2-6
  - PAR File 2-7
  - Unix platform 2-5
- Installation Utility 2-3
- Installing the KM 2-3
- Instance Naming 1-17
- Introduction 1-1

## J

- Job Filter 5-5

## K

- KM Requirements 2-2
- KM versioning and customizations 5-25
- KMs in general 5-25

## L

- Launching Tivoli Workload Scheduler Administration Utilities 5-11
- License requirements for the KM 2-2
- Loading
  - BPPM Portal 2-10
  - PATROL Central 2-9
  - PATROL Console 2-8
- Loading the KM 2-8

## M

- managing events 5-26
- managing monitored objects 5-26
- Memory Usage 1-5
- Menu Summary 3-1
- Monitoring CPU Availability 5-3
- Monitoring Tivoli Workload Scheduler 5-1
- Monitoring Tivoli Workload Scheduler Availability 5-2

## N

- NETMAN Process 5-7

## O

- Objectives of the KM 5-2
- Overview 5-2

## P

- Parameter Summary 4-1
- PATROL Agent configuration variables, deleting 5-25
- PATROL interface 5-26
- PATROL Script Language 5-26
- PATROL versions 1-4
- Preparing to Install or Upgrade 2-4
- Preparing to Use TWS KM 2-2
- Process Tree 5-8, 5-9
- Production Processes 5-10
- Prompt Filter 5-6
- pslInstructionMax 2-8



## R

Refreshing Parameters 5-15  
release notes 2-4  
Reporting from TWS KM 5-12

## S

Schedule Filter 5-4  
Schedule, Job, and Prompt Filters 5-3  
Security Requirements 1-4  
Software Requirements for the KM 2-2  
Starting the Tivoli Workload Scheduler  
NETMAN Process 5-7  
Starting the Tivoli Workload Scheduler  
Process Tree 5-9  
Stopping the Tivoli Workload Scheduler  
Process Tree 5-8  
Stopping the Tivoli Workload Scheduler  
Production Processes 5-10  
Supported Operating Systems 1-3  
supported platforms 1-3  
supported software 1-4  
Supported Versions 1-4

## T

Tivoli Workload Scheduler versions 1-4  
Tivoli™ Workload Scheduler Database  
Validation 5-13  
trial license 2-11  
TWS\_CPU 1-6  
TWS\_CPU Application Class 4-3  
TWS\_CPU Application Menu 3-10  
TWS\_CPU\_CONT 1-6  
TWS\_JOBS 1-6  
TWS\_JOBS Application Class 4-7  
TWS\_JOBS Application Menu 3-19

TWS\_JOBS\_CONT 1-6  
TWS\_JOBS\_CONT Application Menu 3-18  
TWS\_JOBS\_STATUS 1-6  
TWS\_MAESTRO 1-6  
TWS\_MAESTRO Application Class 4-2  
TWS\_MAESTRO Application Menu 3-4  
TWS\_PROCESS 1-6  
TWS\_PROCESS Application Class 4-3  
TWS\_PROCESS\_CONT 1-6  
TWS\_PROMPTS 1-6  
TWS\_PROMPTS Application Class 4-3  
TWS\_PROMPTS Application Menu 3-13  
TWS\_PROMPTS\_CONT 1-6  
TWS\_PROMPTS\_CONT Application Menu  
3-12  
TWS\_PROMPTS\_CPU 1-6  
TWS\_SCHEDULES 1-6  
TWS\_SCHEDULES Application Class 4-7  
TWS\_SCHEDULES Application Menu  
3-16  
TWS\_SCHEDULES\_CONT 1-6  
TWS\_SCHEDULES\_CONT Application  
Menu 3-14  
TWS\_SCHEDULES\_CPU 1-6  
TWS\_SCHEDULES\_CPU Application  
Menu 3-15  
TWS\_SCHEDULES\_STATUS 1-6  
TWS\_SETUP 1-6  
TWS\_SUMMARY 1-6  
TWS\_SUMMARY Application Class 4-3  
TWSABENDJobs 4-3, 4-9  
TWSABENDScheds 4-3, 4-9  
TWSABENPJobs 4-4, 4-9  
TWSADDJobs 4-4, 4-9  
TWSADDScheds 4-4, 4-9  
TWSANCLJobs 4-4, 4-9  
TWSANCLScheds 4-4, 4-9  
TWScpuColl 4-2, 4-9  
TWSDONEJobs 4-4, 4-9  
TWSERRORJobs 4-4, 4-9

- TWSEXECJobs 4-4, 4-9
- TWSEXECScheds 4-4, 4-9
- TWSEXTRNJobs 4-4, 4-9
- TWSFAILJobs 4-5, 4-9
- TWSFENCEJobs 4-5, 4-9
- TWSHOLDJobs 4-5, 4-9
- TWSHOLDScheds 4-5, 4-9
- TWSINTROJobs 4-5, 4-9
- TWSJobsDelayTime 4-7
- TWSJobsElapsedTime 4-7
- TWSJobsStatus 4-7, 4-10
- TWSLinkStatus 4-3, 4-10
- TWSPENDJobs 4-5, 4-10
- TWSProcessColl 4-2, 4-10
- TWSProcessStatus 4-3, 4-10
- TWSPromptFull 4-3, 4-10
- TWSPromptNum 4-3, 4-10
- TWSPromptsColl 4-2, 4-10
- TWSPromptStatus 4-3, 4-10
- TWSPromptText 4-3, 4-10
- TWSREADYJobs 4-5, 4-10
- TWSREADYScheds 4-5, 4-10
- TWSSCHEDJobs 4-6, 4-10
- TWSSchedulesColl 4-2, 4-10
- TWSSchedulesDelayTime 4-7
- TWSSchedulesElapsedTime 4-7
- TWSSchedulesStatus 4-7, 4-11
- TWSSTUCKScheds 4-6, 4-11
- TWSSUCCJobs 4-6, 4-11
- TWSSUCCPJobs 4-6, 4-11
- TWSSUCCScheds 4-6, 4-11
- TWSSUSPJobs 4-6, 4-11
- TWSWAITDJobs 4-6, 4-11
- TWSWAITJobs 4-6, 4-11

- PAR File from the BMC Portal 5-24
- PATROL Agent 5-20
- PATROL Central Console 5-23
- PATROL Central Console Server 5-23
- PATROL Central Web Server 5-24
- PATROL Console 5-21
- Unloading
  - PATROL Agent 5-17
  - PATROL Central 5-19
  - PATROL Console 5-18
- unloading the KM 5-26
- User Guide, pdf format 2-4

## W

- working with menu commands 5-26
- working with parameters 5-26
- working with tasks 5-26

## U

- uninstalling 5-20
  - Deleting PATROL Agent Configuration Variables 5-25