

1. Record Nr.	UNINA9910815710903321
Autore	Gucer Vasfi
Titolo	End-to-end scheduling with IBM Tivoli Workload Scheduler V 8.2 // [Vasfi Gucer, Michael A. Lowry, Finn Bastrup Knudsen]
Pubbl/distr/stampa	[S.I.], : IBM, International Technical Support Organization, 2004
Edizione	[1st ed.]
Descrizione fisica	1 online resource (384 p.)
Collana	IBM redbooks
Altri autori (Persone)	LowryMichael A KnudsenFinn Bastrup
Disciplina	004/.36
Soggetti	Electronic data processing - Distributed processing Production scheduling
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	"September 2004." "SG24-6624-00."
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Front cover -- Contents -- Notices -- Trademarks -- Preface -- The team that wrote this redbook -- Notice -- Become a published author -- Comments welcome -- Chapter 1. Introduction -- 1.1 Job scheduling -- 1.2 Introduction to end-to-end scheduling -- 1.3 Introduction to Tivoli Workload Scheduler for z/OS -- 1.3.1 Overview of Tivoli Workload Scheduler for z/OS -- 1.3.2 Tivoli Workload Scheduler for z/OS architecture -- 1.4 Introduction to Tivoli Workload Scheduler -- 1.4.1 Overview of IBM Tivoli Workload Scheduler -- 1.4.2 IBM Tivoli Workload Scheduler architecture -- 1.5 Benefits of integrating Tivoli Workload Scheduler for z/OS and Tivoli Workload Scheduler -- 1.6 Summary of enhancements in V8.2 related to end-to-end scheduling -- 1.6.1 New functions related with performance and scalability -- 1.6.2 General enhancements -- 1.6.3 Security enhancements -- 1.7 The terminology used in this book -- Chapter 2. End-to-end scheduling architecture -- 2.1 IBM Tivoli Workload Scheduler for z/OS architecture -- 2.1.1 Tivoli Workload Scheduler for z/OS configuration -- 2.1.2 Tivoli Workload Scheduler for z/OS database objects -- 2.1.3 Tivoli Workload Scheduler for z/OS plans -- 2.1.4 Other Tivoli Workload Scheduler for z/OS features -- 2.2 Tivoli Workload Scheduler architecture -- 2.2.1 The IBM Tivoli Workload Scheduler network -- 2.2.2 Tivoli Workload Scheduler workstation types -- 2.2.3 Tivoli

Workload Scheduler topology -- 2.2.4 IBM Tivoli Workload Scheduler components -- 2.2.5 IBM Tivoli Workload Scheduler plan -- 2.3 End-to-end scheduling architecture -- 2.3.1 How end-to-end scheduling works -- 2.3.2 Tivoli Workload Scheduler for z/OS end-to-end components -- 2.3.3 Tivoli Workload Scheduler for z/OS end-to-end configuration -- 2.3.4 Tivoli Workload Scheduler for z/OS end-to-end plans -- 2.3.5 Making the end-to-end scheduling system fault tolerant. 2.3.6 Benefits of end-to-end scheduling -- 2.4 Job Scheduling Console and related components -- 2.4.1 A brief introduction to the Tivoli Management Framework -- 2.4.2 Job Scheduling Services (JSS) -- 2.4.3 Connectors -- 2.5 Job log retrieval in an end-to-end environment -- 2.5.1 Job log retrieval via the Tivoli Workload Scheduler connector -- 2.5.2 Job log retrieval via the OPC connector -- 2.5.3 Job log retrieval when firewalls are involved -- 2.6 Tivoli Workload Scheduler, important files, and directory structure -- 2.7 conman commands in the end-to-end environment -- Chapter 3. Planning end-to-end scheduling with Tivoli Workload Scheduler 8.2 -- 3.1 Different ways to do end-to-end scheduling -- 3.2 The rationale behind end-to-end scheduling -- 3.3 Before you start the installation -- 3.3.1 How to order the Tivoli Workload Scheduler software -- 3.3.2 Where to find more information for planning -- 3.4 Planning end-to-end scheduling with Tivoli Workload Scheduler for z/OS -- 3.4.1 Tivoli Workload Scheduler for z/OS documentation -- 3.4.2 Service updates (PSP bucket, APARs, and PTFs) -- 3.4.3 Tivoli Workload Scheduler for z/OS started tasks for end-to-end scheduling -- 3.4.4 Hierarchical File System (HFS) cluster -- 3.4.5 Data sets related to end-to-end scheduling -- 3.4.6 TCP/IP considerations for end-to-end server in sysplex -- 3.4.7 Upgrading from Tivoli Workload Scheduler for z/OS 8.1 end-to-end scheduling -- 3.5 Planning for end-to-end scheduling with Tivoli Workload Scheduler -- 3.5.1 Tivoli Workload Scheduler publications and documentation -- 3.5.2 Tivoli Workload Scheduler service updates (fix packs) -- 3.5.3 System and software requirements -- 3.5.4 Network planning and considerations -- 3.5.5 Backup domain manager -- 3.5.6 Performance considerations -- 3.5.7 Fault-tolerant agent (FTA) naming conventions. 3.6 Planning for the Job Scheduling Console -- 3.6.1 Job Scheduling Console documentation -- 3.6.2 Job Scheduling Console service (fix packs) -- 3.6.3 Compatibility and migration considerations for the JSC -- 3.6.4 Planning for Job Scheduling Console availability -- 3.6.5 Planning for server started task for JSC communication -- 3.7 Planning for migration or upgrade from previous versions -- 3.8 Planning for maintenance or upgrades -- Chapter 4. Installing IBM Tivoli Workload Scheduler 8.2 end-to-end scheduling -- 4.1 Before the installation is started -- 4.2 Installing Tivoli Workload Scheduler for z/OS end-to-end scheduling -- 4.2.1 Executing EQQJOBS installation aid -- 4.2.2 Defining Tivoli Workload Scheduler for z/OS subsystems -- 4.2.3 Allocate end-to-end data sets -- 4.2.4 Create and customize the work directory -- 4.2.5 Create started task procedures for Tivoli Workload Scheduler for z/OS -- 4.2.6 Initialization statements for Tivoli Workload Scheduler for z/OS end-to-end scheduling -- 4.2.7 Initialization statements used to describe the topology -- 4.2.8 Example of DOMREC and CPUREC definitions -- 4.2.9 The JTOPTS TWSJOBNAME() parameter -- 4.2.10 Verify end-to-end installation in Tivoli Workload Scheduler for z/OS -- 4.3 Installing Tivoli Workload Scheduler in an end-to-end environment -- 4.3.1 Installing multiple instances of Tivoli Workload Scheduler on one machine -- 4.3.2 Verify the Tivoli Workload Scheduler installation -- 4.4 Define, activate, verify fault-tolerant workstations -- 4.4.1 Define fault-tolerant workstation in Tivoli Workload Scheduler controller workstation database -- 4.4.2

Activate the fault-tolerant workstation definition -- 4.4.3 Verify that the fault-tolerant workstations are active and linked -- 4.5 Creating fault-tolerant workstation job definitions and job streams -- 4.5.1 Centralized and non-centralized scripts. 4.5.2 Definition of centralized scripts -- 4.5.3 Definition of non-centralized scripts -- 4.5.4 Combination of centralized script and VARSUB, JOBREC parameters -- 4.5.5 Definition of FTW jobs and job streams in the controller -- 4.6 Verification test of end-to-end scheduling -- 4.6.1 Verification of job with centralized script definitions -- 4.6.2 Verification of job with non-centralized scripts -- 4.6.3 Verification of centralized script with JOBREC parameters -- 4.7 Activate support for the Tivoli Workload Scheduler Job Scheduling Console -- 4.7.1 Install and start Tivoli Workload Scheduler for z/OS JSC server -- 4.7.2 Installing and configuring Tivoli Management Framework 4.1 -- 4.7.3 Alternate method using Tivoli Management Framework 3.7.1 -- 4.7.4 Creating connector instances -- 4.7.5 Creating WTMF administrators for Tivoli Workload Scheduler -- 4.7.6 Installing the Job Scheduling Console -- Chapter 5. End-to-end implementation scenarios and examples -- 5.1 Description of our environment and systems -- 5.2 Creation of the Symphony file in detail -- 5.3 Migrating Tivoli OPC tracker agents to end-to-end scheduling -- 5.3.1 Migration benefits -- 5.3.2 Migration planning -- 5.3.3 Migration checklist -- 5.3.4 Migration actions -- 5.3.5 Migrating backward -- 5.4 Conversion from Tivoli Workload Scheduler network to Tivoli Workload Scheduler for z/OS managed network -- 5.4.1 Illustration of the conversion process -- 5.4.2 Considerations before doing the conversion -- 5.4.3 Conversion process from Tivoli Workload Scheduler to Tivoli Workload Scheduler for z/OS -- 5.4.4 Some guidelines to automate the conversion process -- 5.5 Tivoli Workload Scheduler for z/OS end-to-end fail-over scenarios -- 5.5.1 Configure Tivoli Workload Scheduler for z/OS backup engines -- 5.5.2 Configure DVIPA for Tivoli Workload Scheduler for z/OS end-to-end server. 5.5.3 Configure backup domain manager for first-level domain manager -- 5.5.4 Switch to Tivoli Workload Scheduler backup domain manager -- 5.5.5 Implementing Tivoli Workload Scheduler high availability on high availability environments -- 5.6 Backup and maintenance guidelines for FTAs -- 5.6.1 Backup of the Tivoli Workload Scheduler FTAs -- 5.6.2 Stdlist files on Tivoli Workload Scheduler FTAs -- 5.6.3 Auditing log files on Tivoli Workload Scheduler FTAs -- 5.6.4 Monitoring file systems on Tivoli Workload Scheduler FTAs -- 5.6.5 Central repositories for important Tivoli Workload Scheduler files -- 5.7 Security on fault-tolerant agents -- 5.7.1 The security file -- 5.7.2 Sample security file -- 5.8 End-to-end scheduling tips and tricks -- 5.8.1 File dependencies in the end-to-end environment -- 5.8.2 Handling offline or unlinked workstations -- 5.8.3 Using dummy jobs -- 5.8.4 Placing job scripts in the same directories on FTAs -- 5.8.5 Common errors for jobs on fault-tolerant workstations -- 5.8.6 Problems with port numbers -- 5.8.7 Cannot switch to new Symphony file (EQQPT52E) messages -- Appendix A. Connector reference -- Setting the Tivoli environment -- Authorization roles required -- Working with Tivoli Workload Scheduler for z/OS connector instances -- The wopconn command -- Working with Tivoli Workload Scheduler connector instances -- The wtwsconn.sh command -- Useful Tivoli Framework commands -- Related publications -- IBM Redbooks -- Other publications -- Online resources -- How to get IBM Redbooks -- Help from IBM -- Abbreviations and acronyms -- Index -- Back cover.

Today's challenge is to manage disparate systems with minimal effort and maximum reliability. People experienced in scheduling traditional host-based batch work must now manage distributed systems, and those working in the distributed environment must take responsibility for work running on the corporate OS/390 system. This IBM Redbooks publication considers how best to provide end-to-end scheduling using IBM Tivoli Workload Scheduler Version 8.2, both distributed (previously known as Maestro) and mainframe (previously known as OPC) components. In this book, we provide the information for installing the necessary Tivoli Workload Scheduler 8.2 software components and configuring them to communicate with each other. In addition to technical information, we consider various scenarios that may be encountered in the enterprise and suggest practical solutions. We describe how to manage work and dependencies across both environments using a single point of control. We believe that this book will be a valuable reference for IT specialists who implement end-to-end scheduling with Tivoli Workload Scheduler 8.2.
