

Certification Guide Series: IBM Tivoli Netcool/Impact V4.0 Implementation

Detailed architecture and components discussion

Installation and configuration processing

Policy-based automation engine

Budi Darmawan Thomas Boiocchi Andre Ricardo Cavalcanti de Araujo Mario Schuerewegen Phillip Stanton

Redbooks

ibm.com/redbooks



International Technical Support Organization

Certification Guide Series: IBM Tivoli Netcool/Impact V4.0 Implementation

September 2009

Note: Before using this information and the product it supports, read the information in "Notices" on page vii.

First Edition (September 2009)

This edition applies to Version 4, Release 0 of IBM Tivoli Netcool/Impact (product number 5724-S43).

© Copyright International Business Machines Corporation 2009. All rights reserved. Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Trademarks viii Preface ix The team who wrote this book ix Become a published author ix Comments welcome ixi Comments welcome ixi Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 4 1.1.2 IBM Tivoli Software Professional Certification 4 1.1.3 Growth through skills 6 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact deployment types 35	Notices	. vii
Preface ix The team who wrote this book ix Become a published author ix Comments welcome ixi Comments welcome ixi Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.3 Growth through skills 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact components 30 2.1.1 IBM Tivoli Netcool/Impact deployment types 3	Trademarks	. viii
Preface ix The team who wrote this book ix Become a published author ix Comments welcome ix Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.1.3 Growth through skills 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 16 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration <td></td> <td></td>		
The team who wrote this book.	Preface	ix
Become a published author xi Comments welcome xii Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.3 Growth through skills 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35	The team who wrote this book	ix
Comments welcome xii Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.1.3 Growth through skills 6 1.2.1 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Impact deployment types <	Become a published author	Xİ
Chapter 1. Certification overview 1 1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.3 Growth through skills. 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2.1 IBM Tivoli Netcool/Impact deployment types 35 2.2.2 Important concepts 36 2.2.3 Data Service Adapter (DSAs)	Comments welcome	. xii
1.1 IBM Professional Certification Program 2 1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.3 Growth through skills 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39	Chapter 1 Cortification overview	-
1.1.1 Benefits of certification 3 1.1.2 IBM Tivoli Software Professional Certification 4 1.1.3 Growth through skills. 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2.2 Important concepts 36 2.2.3 Data Service Adapter (DSAs) 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 <td>1.1 IBM Professional Cartification Program</td> <td>ເ</td>	1.1 IBM Professional Cartification Program	ເ
1.1.1 Berlenis of certification 4 1.1.2 IBM Tivoli Software Professional Certification 4 1.1.3 Growth through skills 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcoo	1.1 Denofite of portification	ב. ג ס
1.1.2 IBM Twoli Software Professional Certification 4 1.1.3 Growth through skills. 6 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning. 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning	1.1.2 IPM Tiveli Software Professional Cartification	د ۸
1.2 IBM Tivoli Netcool/Impact V4.0 test objectives 8 1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.1.2 Crowth through akilla	4 6
1.2.1 Planning 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.0. IDM Tiveli Netecol/Import V/4.0 test objectives	0 0
1.2.1 Flaming 9 1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2 IBM INVOLVELCOOMINPACT V4.0 lest objectives	0
1.2.2 Installation 11 1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 IBM Tivoli Netcool/Webtop services 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data Models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3	1.2.1 Pidilining	9
1.2.3 Configuration 13 1.2.4 Development 16 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Impact deployment types 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.2 Installation	. 11
1.2.4 Development 10 1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.3 Configuration	. 13
1.2.5 Administration 19 1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.4 Development	10
1.2.6 Troubleshooting and performance tuning 21 1.3 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.5 Administration	. 19
1.3.1 Certification achieved 23 1.3.1 Tivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.6 Troubleshooling and performance luning	. 21
1.3.1 Trivoli Netcool/Impact V4.0 23 1.3.2 Tivoli Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.2.1 Tiveli Neteool/Import V/4.0	. ∠ა იი
1.3.2 Twoir Fault Management Solutions 2008 25 1.3.3 IBM Service Management Network and Service Assurance 2009 26 1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	1.9.0 Tivoli Nelcool/Impact V4.0	. 23 05
1.3.3 IBM Service Management Network and Service Assurance 2009 20 1.4 Recommended study resources	1.3.2 The Fault Management Network and Service Acquirence 2000	. 20
1.4 Recommended study resources 27 1.4.1 Courses 28 Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme. 42	1.4. Decommended study recourses	. 20
Chapter 2. Planning. 29 2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components. 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme. 42		. 21
Chapter 2. Planning.292.1 IBM Tivoli Netcool/Impact architecture302.1.1 IBM Tivoli Netcool/Impact components.302.1.2 Communication configuration342.1.3 IBM Tivoli Netcool/Impact deployment types352.2 Important concepts362.2.1 IBM Tivoli Netcool/Webtop services362.2.2 Data models382.2.3 Data Service Adapter (DSAs)392.3 IBM Tivoli Netcool/Impact policy planning412.4 User and authorization scheme.42	1.4.1 Courses	. 20
2.1 IBM Tivoli Netcool/Impact architecture 30 2.1.1 IBM Tivoli Netcool/Impact components 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	Chapter 2. Planning	. 29
2.1.1 IBM Tivoli Netcool/Impact components. 30 2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.1 IBM Tivoli Netcool/Impact architecture	. 30
2.1.2 Communication configuration 34 2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.1.1 IBM Tivoli Netcool/Impact components.	. 30
2.1.3 IBM Tivoli Netcool/Impact deployment types 35 2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.1.2 Communication configuration	. 34
2.2 Important concepts 36 2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.1.3 IBM Tivoli Netcool/Impact deployment types	. 35
2.2.1 IBM Tivoli Netcool/Webtop services 36 2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.2 Important concepts	. 36
2.2.2 Data models 38 2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.2.1 IBM Tivoli Netcool/Webtop services	. 36
2.2.3 Data Service Adapter (DSAs) 39 2.3 IBM Tivoli Netcool/Impact policy planning 41 2.4 User and authorization scheme 42	2.2.2 Data models	. 38
2.3 IBM Tivoli Netcool/Impact policy planning	2.2.3 Data Service Adapter (DSAs)	. 39
2.4 User and authorization scheme	2.3 IBM Tivoli Netcool/Impact policy planning	. 41
	2.4 User and authorization scheme.	. 42
2.5 Application dependency	2.5 Application dependency	. 43

2.6 Failover configuration	44
Chapter 3. Installation	47
3.1 Preparation	48
3.1.1 Prerequisites	48
3.1.2 Parameters	49
3.1.3 Installation image	49
3.2 Basic installation	50
3.2.1 Graphical installation	50
3.2.2 Other installation types	63
3.2.3 Installation log file	64
3.2.4 JRExec service on Windows	64
3.3 Cluster installation	64
3.3.1 Simple IBM Tivoli Netcool/Webtop cluster installation	65
3.3.2 Complete failover IBM Tivoli Netcool/Impact cluster installation	69
3.4 Self-monitoring	73
3.4.1 Configuring the self-monitoring tool using the GUI	74
3.4.2 Configuring self-monitoring using the CLI	75
3.5 Startup configuration	79
	~~~
Chapter 4. Configuration	83
4.1 Data sources	04 01
4.1.1 Data source deminion	04
4.1.2 Configuring a new data source	
4.2 Data types	
4.2.1 Data type definition	
	92
4.3 1 Policy definition	
4.3.2 Policy configuration	06
4.5.2 Folicy conliguration.	08
4.4 1 Event Reader definition	08
4.4.1 Event Reader configuration	100
4.5 Event Processor	10/
4.5.1 Event Processor definition	104
4.5.2 Event Processor configuration	104
4.6 Logging configuration	104
4.7 Operator View configuration	100
	. 105
Chapter 5. Development	. 113
5.1 IBM Tivoli/Netcool Impact policy language (IPL)	. 114
5.1.1 Common elements	. 114
5.1.2 Function	. 114
5.1.3 Operator	. 115

5.1.4 IBM Tivoli Netcool/Webtop policy development types	116
5.2 Development requirements	117
5.2.1 Scenario	117
5.2.2 Development plan	117
5.2.3 Data information	119
5.3 Sample development.	119
5.3.1 Create database fields in the ObjectServer	120
5.3.2 Define the data source for a customer table	125
5.3.3 Create the policy	131
5.3.4 Configure a service to use the policy	132
5.4 Verifying and testing the policy	136
5.5 Built-in functions	138
5.5.1 Action functions	138
5.5.2 Parser functions	140
5.5.3 Web services functions	142
5.5.4 SNMP functions	143
5.6 Exceptions in policy	143
5.6.1 Raising exceptions	143
5.6.2 Handling exceptions	144
5.6.3 Using exceptions	145
5.7 Auto execute of policy	145
	4 4 7
Chapter 6. Administration	147
Chapter 6. Administration	147 148
Chapter 6. Administration 6.1 IBM Tivoli Netcool/Impact project creation 6.2 Configure users and roles C.2 IBM Tivoli Netcool/Impact chiest deployment	147 148 149
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment	147 148 149 149
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import	147 148 149 149 150
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import         6.5 Enterprise application redeployment	147 148 149 149 150 151
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment	147 148 149 149 150 151 151
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment	147 148 149 149 150 151 151 152
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration	147 148 149 149 150 151 151 152
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface	147 148 149 150 151 151 152 152
Chapter 6. Administration 6.1 IBM Tivoli Netcool/Impact project creation 6.2 Configure users and roles 6.3 IBM Tivoli Netcool/Impact object deployment 6.4 Data export/import. 6.5 Enterprise application redeployment 6.5.1 Hot redeployment 6.5.2 Cold redeployment 6.5.3 WebSphere Application Server Community Edition administration interface Chapter 7. Troubleshooting and performance tuning	147 148 149 149 150 151 151 152 152 152
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface         7.1 System verification	147 148 149 149 150 151 151 152 152 152
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface         7.1 System verification         7.2 Performance reporting.	147 148 149 149 150 151 151 152 152 155 156 163
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import.         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface         7.1 System verification         7.2 Performance reporting.         7.3 Logging and testing.	147 148 149 149 150 151 151 152 152 155 156 163 166
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface         7.1 System verification         7.2 Performance reporting.         7.3 Logging and testing         7.3.1 Log files.	147 148 149 150 151 151 152 152 155 156 163 166
Chapter 6. Administration         6.1 IBM Tivoli Netcool/Impact project creation         6.2 Configure users and roles         6.3 IBM Tivoli Netcool/Impact object deployment         6.4 Data export/import         6.5 Enterprise application redeployment         6.5.1 Hot redeployment         6.5.2 Cold redeployment         6.5.3 WebSphere Application Server Community Edition administration interface         7.1 System verification         7.2 Performance reporting.         7.3 Logging and testing.         7.3.1 Log files.         7.3 2 Event processing	147 148 149 150 151 151 152 155 156 163 166 166
<ul> <li>Chapter 6. Administration</li> <li>6.1 IBM Tivoli Netcool/Impact project creation</li> <li>6.2 Configure users and roles</li> <li>6.3 IBM Tivoli Netcool/Impact object deployment</li> <li>6.4 Data export/import</li> <li>6.5 Enterprise application redeployment</li> <li>6.5.1 Hot redeployment</li> <li>6.5.2 Cold redeployment</li> <li>6.5.3 WebSphere Application Server Community Edition administration interface</li> <li>Chapter 7. Troubleshooting and performance tuning</li> <li>7.1 System verification</li> <li>7.2 Performance reporting.</li> <li>7.3 Logging and testing</li> <li>7.3.1 Log files</li> <li>7.3.2 Event processing.</li> <li>7.3.3 Other troubleshooting tips</li> </ul>	147 148 149 150 151 151 152 152 155 156 163 166 166 167 167
<ul> <li>Chapter 6. Administration</li> <li>6.1 IBM Tivoli Netcool/Impact project creation</li> <li>6.2 Configure users and roles</li> <li>6.3 IBM Tivoli Netcool/Impact object deployment</li> <li>6.4 Data export/import</li> <li>6.5 Enterprise application redeployment</li> <li>6.5.1 Hot redeployment</li> <li>6.5.2 Cold redeployment</li> <li>6.5.3 WebSphere Application Server Community Edition administration interface</li> </ul> Chapter 7. Troubleshooting and performance tuning <ul> <li>7.1 System verification</li> <li>7.2 Performance reporting.</li> <li>7.3 Logging and testing</li> <li>7.3.1 Log files</li> <li>7.3.3 Other troubleshooting tips</li> <li>7.4 Impact CVS</li> </ul>	147 148 149 150 151 151 152 155 156 163 166 166 167 167
<ul> <li>Chapter 6. Administration</li> <li>6.1 IBM Tivoli Netcool/Impact project creation</li> <li>6.2 Configure users and roles</li> <li>6.3 IBM Tivoli Netcool/Impact object deployment</li> <li>6.4 Data export/import.</li> <li>6.5 Enterprise application redeployment.</li> <li>6.5.1 Hot redeployment</li> <li>6.5.2 Cold redeployment</li> <li>6.5.3 WebSphere Application Server Community Edition administration interface</li> </ul> Chapter 7. Troubleshooting and performance tuning <ul> <li>7.1 System verification</li> <li>7.2 Performance reporting.</li> <li>7.3 Logging and testing.</li> <li>7.3.1 Log files.</li> <li>7.3.2 Event processing.</li> <li>7.3.3 Other troubleshooting tips.</li> <li>7.4 Impact CVS.</li> </ul>	147 148 149 150 151 151 152 155 156 163 166 166 167 167 168
<ul> <li>Chapter 6. Administration</li> <li>6.1 IBM Tivoli Netcool/Impact project creation</li> <li>6.2 Configure users and roles</li> <li>6.3 IBM Tivoli Netcool/Impact object deployment</li> <li>6.4 Data export/import</li> <li>6.5 Enterprise application redeployment</li> <li>6.5.1 Hot redeployment</li> <li>6.5.2 Cold redeployment</li> <li>6.5.3 WebSphere Application Server Community Edition administration interface</li> <li>Chapter 7. Troubleshooting and performance tuning</li> <li>7.1 System verification</li> <li>7.2 Performance reporting.</li> <li>7.3 Logging and testing.</li> <li>7.3.1 Log files.</li> <li>7.3.2 Event processing.</li> <li>7.3.3 Other troubleshooting tips.</li> <li>7.4 Impact CVS.</li> </ul>	147 148 149 150 151 151 152 152 155 156 163 166 167 167 167 168 171

Sample answers	5
Related publications       17         BM Redbooks       17         Dther publications       17         17       17         17       17         17       17         17       17         17       17         17       17         17       17         17       17         17       17	777
Johine resources       176         How to get Redbooks       176         Help from IBM       176	8 8 8
ndex	9

## **Notices**

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

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

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

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

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

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

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

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

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

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

#### COPYRIGHT LICENSE:

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

#### **Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX 5L™	Netcool®
AIX®	PartnerWorld®
ClearCase®	Passport Advantage®
DB2®	POWER®
Foundations™	Rational®
IBM®	Redbooks®
Lotus®	Redbooks (logo) 🧬 🛽

System z® Tivoli Enterprise Console® Tivoli® ValueNet® WebSphere®

The following terms are trademarks of other companies:

ValueNet, and the FileNet logo are registered trademarks of FileNet Corporation in the United States, other countries or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

SUSE, the Novell logo, and the N logo are registered trademarks of Novell, Inc. in the United States and other countries.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

Red Hat, and the Shadowman logo are trademarks or registered trademarks of Red Hat, Inc. in the U.S. and other countries.

EJB, Java, JDBC, JVM, MySQL, Solaris, Sun, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Active Directory, Internet Explorer, Microsoft, SQL Server, Windows Server, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

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

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

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

### Preface

This IBM® Redbooks® publication is a study guide for the IBM Tivoli® Netcool®/Impact V4.0 Implementation Certification test. It is aimed at the IT professional who wants to be an IBM Certified Professional for this product.

The IBM Tivoli Netcool/Impact V4.0 Implementation Certification test is offered through the IBM Professional Certification program. It is designed to validate the skills required of technical professionals who work in the implementation and deployment of IBM Tivoli Netcool/Impact V4.0.

This book provides the necessary information for understanding the subject matter. It includes sample questions. The sample questions help evaluate your personal progress. It familiarizes you with the types of questions that may be encountered in the exam.

This guide does not replace practical experience. It is not designed to be a stand-alone guide on the subject. Instead, this guide should be combined with educational activities and experiences and used as a very useful preparation guide for the exam.

For your convenience, the chapters are based on the certification objectives of the IBM Tivoli Netcool/Impact V4.0 Implementation Certification test. Those requirements are planning, prerequisites, installation, configuration, administration, and problem determination. Studying each chapter helps you prepare for the objectives of the exam.

#### The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.



Figure 1 Mario, Thomas, Andre, and Phillip

**Budi Darmawan** is a Project Leader at the International Technical Support Organization, Austin Center. He writes extensively and teaches IBM classes worldwide on all areas of Tivoli and systems management. Before joining the ITSO 10 years ago, Budi worked in Integrated Solution Services, IBM Indonesia as lead implementer and solution architect.

**Thomas Boiocchi** an IT Specialist based in Italy, working for Tivoli Services since 2007. He joined IBM after working for several years as a Netcool Specialist for Eirteic Consulting travelling around the world. He has 10 years of IT experience and previously worked as a system and network administrator in Telkom and banks in Italy. His area of expertise include IBM Tivoli Netcool/OMNIbus, IBM Tivoli Business Services Manager, Network Manager, and IBM Tivoli Netcool/Impact.

Andre Ricardo Cavalcanti de Araujo is a System Management Information Technology Specialist working with Tivoli Management Product in Brazil. He has 12 years of experience in servers and systems support. He hold a degree in Telecommunication Engineering and has MBA in Network Computer Management. His areas of expertise include UNIX/Linux® support, Cisco networking, networking security, and infrastructure and networking management.

**Mario Schuerewegen** is a Technical Presales specialist based in Belgium that specializes in Netcool products. He has 10 years of experience in the network and event management field. His areas of expertise include Cisco networking, SNMP, and network management.

Philip Stanton is an L2 software support specialist based in the United States. He has 12 years of Information Technology experience and holds a Bachelors of Science degree in Business Administration with an emphasis on Information Systems. Most of his experience is in supporting and administrating mixed UNIX® and Microsoft® Windows® platforms for various Business to Business services and e-commerce Web sites. He joined IBM 2 years ago in L2 support for the IBM Tivoli Netcool/OMNIbus, IBM Tivoli Netcool/Webtop, System Service monitors, and Internet Service Monitors software solutions. He currently holds the following certifications: Netcool Core V2, Netcool Core V3, Tivoli Level 2 Support Tools and Processes, and MCSE NT4/2000.

Thanks to the following people for their contributions to this project:

Tamikia Barrow, Margaret Ticknor, Wade Wallace International Technical Support Organization, Austin Center

Jill Kanatzar IBM Software Group, Worldwide Sales Channel Growth Executive

#### Become a published author

Join us for a two- to six-week residency program! Help write a book dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

#### **Comments welcome**

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

Use the online Contact us review Redbooks form found at:

ibm.com/redbooks

► Send your comments in an e-mail to:

redbooks@us.ibm.com

Mail your comments to:

IBM Corporation, International Technical Support Organization Dept. HYTD Mail Station P099 2455 South Road Poughkeepsie, NY 12601-5400

# 1

## **Certification overview**

This chapter provides an overview of the skills requirements needed to obtain an IBM Tivoli Netcool/Impact V4.0 Implementation Certification. This chapter provides a comprehensive review of the topics that are essential for obtaining the certification in the following sections:

- ▶ 1.1, "IBM Professional Certification Program" on page 2
- ► 1.2, "IBM Tivoli Netcool/Impact V4.0 test objectives" on page 8
- ► 1.3, "Certification achieved" on page 23
- ▶ 1.4, "Recommended study resources" on page 27

#### **1.1 IBM Professional Certification Program**

Having the right skills for the job is critical in the growing global marketplace. IBM Professional Certification is designed to validate your skills and proficiency with the latest IBM solution and product technology. It can help provide that competitive edge.

The Professional Certification Program from IBM offers a business solution for skilled technical professionals seeking to demonstrate their expertise to the world. The program is designed to validate your skills and demonstrate your proficiency in the latest IBM technology and solutions. In addition, professional certification may help you excel at your job by giving you and your employer the confidence that your skills have been tested. You may be able to deliver higher levels of service and technical expertise than non-certified employees and move to a faster career track.

The certification requirements are difficult, but they are not overwhelming. IBM Professional Certification is a rigorous process that differentiates you from everyone else. The mission of IBM Professional Certification is to:

- Provide a reliable, valid, and fair method of assessing skills and knowledge.
- Provide IBM with a method of building and validating the skills of individuals and organizations.
- Develop a loyal community of highly skilled certified professionals who recommend, sell, service, support, and use IBM products and solutions.

The Professional Certification Program from IBM has developed certification role names to guide you in your professional development. The certification role names include IBM Certified Specialist, IBM Certified Solutions/Systems Expert, and IBM Certified Advanced Technical Expert. These role names are for technical professionals who sell, service, and support IBM solutions. For technical professionals in application development, the certification roles include IBM Certified Developer Associate and IBM Certified Developer. An IBM Certified Instructor certifies the professional instructor.

The Professional Certification Program from IBM provides you with a structured program leading to an internationally recognized qualification. The program is designed for flexibility by allowing you to select your role, prepare for and take tests at your own pace, and, in some cases, select from a choice of elective tests best suited to your abilities and needs. Some roles also offer a shortcut by giving credit for a certification obtained in other industry certification programs.

You can be a network administrator, systems integrator, network integrator, solution architect, solution developer, value-added reseller, technical coordinator, sales representative, or educational trainer. Regardless of your role, you can start charting your course through the Professional Certification Program from IBM today.

The IBM Professional Certification Program Web site is available at the following address:

http://www.ibm.com/certify/index.shtml

#### 1.1.1 Benefits of certification

Certification is a tool to help objectively measure the performance of a professional on a given job at a defined skill level. Therefore, it is beneficial for individuals who want to validate their own skills and performance levels, their employees, or both. For optimum benefit, the certification tests must reflect the critical tasks required for a job, the skill levels of each task, and the frequency by which a task needs to be performed. IBM prides itself in designing comprehensive, documented processes that ensure that IBM certification tests remain relevant to the work environment of potential certification candidates.

In addition to assessing job skills and performance levels, professional certification can also provide such benefits as:

- For employees:
  - Promotes recognition as an IBM certified professional
  - Helps to create advantages in interviews
  - Assists in salary increases, corporate advancement, or both
  - Increases self-esteem
  - Provides continuing professional benefits
- For employers:
  - Measures the effectiveness of training
  - Reduces course redundancy and unnecessary expenses
  - Provides objective benchmarks for validating skills
  - Makes long-range planning easier
  - Helps to manage professional development
  - Aids as a hiring tool
  - Contributes to competitive advantage
  - Increases productivity
  - Increases morale and loyalty

Specific benefits can vary by country (region) and role. In general, after you become certified, you should receive the following benefits:

Industry recognition

Certification may accelerate your career potential by validating your professional competency and increasing your ability to provide solid, capable technical support.

Program credentials

As a certified professional, you receive through e-mail your certificate of completion and the certification mark associated with your role for use in advertisements and business literature. You can also request a hardcopy certificate, which includes a wallet-size certificate.

The Professional Certification Program from IBM acknowledges the individual as a technical professional. The certification mark is for the exclusive use of the certified individual.

Ongoing technical vitality

IBM Certified professionals are included in mailings from the Professional Certification Program from IBM.

#### 1.1.2 IBM Tivoli Software Professional Certification

The IBM Tivoli Professional Certification program offers certification testing that sets the standard for qualified product consultants, administrators, architects, and Business Partners.

The program also offers an internationally recognized qualification for technical professionals seeking to apply their expertise in today's complex business environment. The program is designed for those who implement, buy, sell, service, and support IBM Tivoli solutions and want to deliver higher levels of service and technical expertise.

#### **Benefits of being Tivoli certified**

Tivoli certification provides the following benefits:

- For the individual:
  - IBM Certified certificate and use of logos on business cards
  - Recognition of your technical skills by your peers and management
  - Enhanced career opportunities
  - Focus for your professional development

- ► For the IBM Business Partner:
  - Confidence in the skills of your employees
  - Enhanced partnership benefits from the IBM Business Partner program
  - Billing your employees out at higher rates
  - Strengthens your proposals to customers
  - Demonstrates the depth of technical skills available to prospective customers
- ► For the customer:
  - Confidence in the service professionals handling your implementation
  - Ease of hiring competent employees to manage your Tivoli environment
  - Enhanced return on investment (ROI) through more thorough integration with Tivoli and third-party products
  - Ease of selecting a Tivoli Business Partner that meets your specific needs

#### **Certification checklist**

The certification process is as follows:

- 1. Select the certification that you want to pursue.
- 2. Determine which test or tests are required by reading the certification role description.
- 3. Prepare for the test, using the following resources:
  - Test objectives (1.2, "IBM Tivoli Netcool/Impact V4.0 test objectives" on page 8)
  - Recommended educational resources (1.4, "Recommended study resources" on page 27)
  - Sample/assessment test (Appendix A, "Sample test" on page 171)
  - Other reference materials
  - Opportunities for experience
- 4. Register to take a test by contacting one of our worldwide testing vendors:
  - Thomson Prometric
  - Pearson Virtual University Enterprises (VUE)
- 5. Take the test. Be sure to keep the Examination Score Report provided upon test completion as your record of taking the test.

- 6. Repeat steps three through five until all required tests are successfully completed for the desired certification role. If additional requirements are needed (such as another vendor certification or exam), follow the instructions on the certification description page to submit these requirements to IBM.
- 7. After you complete your certification requirements, you will be sent an e-mail asking you to accept the terms of the IBM Certification Agreement before receiving the certificate.
- 8. Upon acceptance of the terms of the IBM Certification Agreement, an e-mail will be sent containing the following electronic deliverables:
  - A Certification Certificate in PDF format, which can be printed in either color or black and white
  - A set of graphic files of the IBM Professional Certification mark associated with the certification achieved
  - Guidelines for the use of the IBM Professional Certification mark
- 9. To avoid unnecessary delay in receiving your certificate, ensure that we have your current e-mail on file by keeping your profile up to date. If you do not have an e-mail address on file, your certificate will be sent through postal mail.

After you receive a certificate by e-mail, you can also contact IBM at mailto:certify@us.ibm.com to request that a hardcopy certificate be sent by postal mail.

#### 1.1.3 Growth through skills

Customers want to work with experts who understand their business and can help them achieve their objectives. IBM Business Partners who have expertise across the IBM software portfolio are well positioned to deliver high client value.

IBM Software is announcing the next step in our Business Partner channel strategy by focusing on Growth Through Skills. In October 2009, IBM will roll out a new controlled distribution model to maximize value to our Business Partners and customers.

A subset of the IBM software portfolio will continue to be offered through the open distribution model or by using Software ValueNet®. The benefits of the growth through skills program are:

- Protects and maximizes your return on investment (ROI) in the technical, sales and marketing skills you have developed.
- Places a premium on your skills and solutions that differentiate your ability to offer your customers guidance in a tough economy.

- Rewards the value you bring throughout the sales cycle through the lucrative IBM Software Value Incentive (SVI).
- Provides financial rewards for integrating IBM software with your business solutions through the Value Advantage Plus (VAP) incentive.
- Accelerates your growth with experienced software Value Added Distributors (VADs).
- Improves access to IBM resources, including industry-leading sales, technical, and marketing.

Authorization to resell IBM software products within controlled distribution is achieved at the product group level. There are 14 products groups across the five brands.

- ► WebSphere®
  - SOA Foundation
  - Connectivity
  - Business Process Management
  - Commerce
  - SOA Appliances
  - Enterprise Solutions (IBM System z®)
- Tivoli
  - Storage Management
  - Security & Compliance Management
  - Automation
  - Enterprise Asset Management
- Information Management
  - Heritage CM
  - Data Management
- ► Lotus®
- Portal
- Rational®

The criteria for authorization to resell IBM Software products within controlled distribution include:

► Membership in the IBM PartnerWorld® program

- Approved participation in Software Value Incentive (SVI) or Value Advantage Plus (VAP)
  - For SVI, technical and sales skills in the product group(s) you want to sell
  - For VAP, an approved solution containing the product group(s) you want to sell
- An approved PartnerPlan
- ► Minimum revenue participation levels within SVI and VAP after the first year

IBM provides comprehensive enablement options to support the education, training, and certifications necessary to qualify for authorization to resell:

- Leverage the readiness assessment tools and work with your Distributor or IBM Business Partner Sales Representative to explore enablement opportunities and support.
- Visit the Subject Matter Expert (SME) Zone on the Virtual Innovation Center as a single point of entry to review software education and customized roadmaps.
- ► Learn about the You Pass, We Pay education reimbursement.
- Participate in readiness events throughout the year and revisit the Web for updates and the latest support materials.

#### 1.2 IBM Tivoli Netcool/Impact V4.0 test objectives

The test has the following objectives:

- ▶ 1.2.1, "Planning" on page 9
- ► 1.2.2, "Installation" on page 11
- ► 1.2.3, "Configuration" on page 13
- ▶ 1.2.4, "Development" on page 16
- 1.2.5, "Administration" on page 19
- ► 1.2.6, "Troubleshooting and performance tuning" on page 21

For the most updated objectives of the IBM Tivoli Netcool/Impact V4.0 Implementation test (test #938), refer to the following address:

http://www-03.ibm.com/certify/tests/obj938.shtml

This test has the following aspects:

- Number of questions: 65
- ► Time allowed in minutes: 75
- ► Required passing score: 68%

#### 1.2.1 Planning

Given the stakeholder's requirements, document the functional steps so that an IBM Tivoli Netcool/Impact solution is designed, with emphasis on the following steps:

- 1. Analyze stakeholder high-level requirements.
- 2. Determine methods of achieving customer requirements.
- 3. Identify local and remote data to process.
- 4. Determine the components necessary to achieve the requirements.
- 5. Create a solution with the necessary components design policies to fulfill the functional requirements.

Given the functional requirements determined by the stakeholders and the current state system architecture (if one exists), gather the technical deployment requirements from the stakeholders so that a full technical architecture is documented and approved, with emphasis on the following steps:

- 1. Determine the hardware and operating system restrictions (what is available or desired).
- 2. Gather the current Netcool component deployment architecture (ObjectServers, Security Manager, and unique deployment features).
- 3. Gather and determine event source requirements (Event Readers and Event Listeners).
- 4. Determine the sources of data.
- 5. Determine the availability requirements (clustering and failover requirements).
- 6. Determine the Release Management infrastructure/requirements (dev, test, production, or other).
- 7. Determine other external integration points (APIs, notification, and external systems).
- 8. Determine the connectivity requirements (firewall paths, network access, security restrictions, and requirements).
- 9. Create technical architecture documentation.
- 10. Present the architecture and obtain signoff from the relevant stakeholders.

Given the IBM Tivoli Netcool/Impact functional and architectural requirements, determine the IBM Tivoli Netcool/Impact policy flow so that an IBM Tivoli Netcool/Impact Policy Design standard is created, with emphasis on the following steps:

- 1. Determine policy naming convention.
- Determine data models: data sources, data types, data items, links, and event sources.
- 3. Determine policy types: event enrichment, x events in y time, event notification, and event gateway.
- 4. Determine services: Event Readers/Listeners, E-mail Sender/Readers, and Policy Activators.
- 5. Determine external requirements: data sources, ObjectServer fields, rules files, and scripts.
- 6. Determine pre-conditions: suppression states, triggering events, and timing.
- 7. Determine post-conditions: suppression states, synthetic/meta events creation, event fields, and notifications.
- 8. Determine logging standards: Required, Info, and Debug.

Given that there are stakeholders using IBM Tivoli Netcool/Impact, communicate and coordinate their involvement so that IBM Tivoli Netcool/Impact functionality can be exploited, with emphasis on the following steps:

- Identify, using data users, which IBM Tivoli Netcool/OMNIbus events are incomplete.
- 2. Identify which external data owners have information that can be used to enhance IBM Tivoli Netcool/OMNIbus events.
- 3. Request that external data owners advise you about their data structures (for example, database names, tables, and fields).
- Request that external data owners create user accounts to permit IBM Tivoli Netcool/Impact to access the data.
- 5. Determine the external data access details for creating data sources.
- 6. Determine which events need enriching and develop and implement the appropriate policies.
- 7. Confirm, with data users, that the events now contain additional information.

Given that functional requirements for IBM Tivoli Netcool/Impact have been identified and external resources have been established, determine the specifics about the users, administrators, and authentication so that users can be configured within IBM Tivoli Netcool/Impact, with emphasis on the following steps:

- 1. Determine the administrators of the Netcool users.
- 2. Determine the users of the IBM Tivoli Netcool/Impact system.
- 3. Determine the native or external authentication.
- 4. Determine the SSL requirements.

Given detailed functional requirements, assess the existing application infrastructure so that application dependencies are documented, with emphasis on the following steps:

- 1. Review the detailed functional requirements.
- 2. Assess the application infrastructure.
- 3. Determine the Netcool application dependencies: Precision, Tivoli Business Service Manager, ObjectServers, fields in ObjectServer, database accesses, data sources, and data types.
- 4. Determine other application dependencies, for example, Oracle®, Web Services, DB2®, Jabber, XML, e-mail, and data source adapters.
- 5. Document the application dependencies.

#### 1.2.2 Installation

Given certain prerequisite components, verify that they are set up so that IBM Tivoli Netcool/Impact Server is ready for installation, with emphasis on the following steps:

- 1. Verify the installation of Netcool Security Manager.
- 2. Verify the installation of an IBM Tivoli Netcool/OMNIbus ObjectServer instance.
- 3. Verify the IBM Tivoli Netcool/Impact server system requirements.
  - a. Verify file permissions for source and destination directories where IBM Tivoli Netcool/Impact will be installed.
  - b. Verify that all the necessary operating system patches and packages are installed on IBM Tivoli Netcool/Impact Server.
- 4. Verify the hard disk space requirements on the IBM Tivoli Netcool/Impact server.
- 5. Verify the availability of console emulation availability when remotely installing IBM Tivoli Netcool/Impact on a Windows Server®.

Given available ports, additional components, resiliency requirements, and nameserver location, identify all of the installation parameters so that the installation parameters are identified in preparation for an installation of IBM Tivoli Netcool/Impact, with emphasis on the following steps:

- Identify whether the default installation directory (\$NCHOME for UNIX, \$NCHOME\$ for Windows) has been defined. If so, assess whether there may be any conflicts with existing applications. Identify the configuration settings for additional components (such as License Manager (if applicable), Security Manager, GUI Server (if applicable), and default ObjectServer).
- 2. Determine the naming conventions for the IBM Tivoli Netcool/Impact servers and clusters.
- Determine if the IBM Tivoli Netcool/Impact database server will be installed. If so (for UNIX only), ensure that the minimum specified shared memory is available.
- 4. Determine if there are any potential conflicts with the default ports that will be used by using the **netstat** command and referring to the *IBM Tivoli Netcool/Impact Administration Guide*, SC23-8853.
- 5. Identify the settings for the E-mail Sender Service, if required.
- 6. Determine which version of CVS will be used.

Given the determined technical architecture and installation prerequisites and parameters, obtain, install, test, and document the IBM Tivoli Netcool/Impact solution so that a functioning IBM Tivoli Netcool/Impact solution is available and the deployment is documented, with emphasis on the following steps:

- 1. Download the IBM Tivoli Netcool/Impact server binaries.
- 2. Uncompress the IBM Tivoli Netcool/Impact server binaries.
- 3. Review the installation parameters.
- 4. Execute the IBM Tivoli Netcool/Impact installer.
- 5. Review the installation and installation logs for issues.
- 6. Start the IBM Tivoli Netcool/Impact server.
- 7. Test availability and the proper installation.
- 8. Document the installation procedure.

Given the architecture resiliency requirements, install two IBM Tivoli Netcool/Impact server cluster members, configure the necessary failover points, and set up self-monitoring of IBM Tivoli Netcool/Impact so that all components meet the requirements, with emphasis on the following steps:

- 1. Review the requirements and determine the appropriate resilient architecture.
- 2. Install the primary IBM Tivoli Netcool/Impact server in the cluster.
- 3. Install the secondary IBM Tivoli Netcool/Impact server cluster member.
- 4. Set the server clustering properties.
- 5. Configure an ObjectServer data source with failover.

- 6. Configure other data sources with failover.
- 7. Set up IBM Tivoli Netcool/Impact self-monitoring.
- 8. Recommend actions to be taken for self-monitoring events.

Given the various IBM Tivoli Netcool/Impact component scripts, configure application startup so that all necessary components are running and will start up automatically upon physical server restart, with emphasis on the following steps:

- 1. Select the IBM Tivoli Netcool/Impact services to automatically start up (such as Event Readers, Policy Activators, and E-mail Readers).
- 2. Start the GUI server (if deployed remotely).
- 3. Start the WebSphere Application Server Community Edition server (includes the auto-startup of the GUI server and IBM Tivoli Netcool/Impact server, unless they are deployed remotely).
- 4. Start the IBM Tivoli Netcool/Impact PostgreSQL database.
- 5. Start the IBM Tivoli Netcool/Impact server separately (if deployed remotely).
- Start the secondary IBM Tivoli Netcool/Impact Server (if clustering is deployed).
- 7. Start the JRExec process on all IBM Tivoli Netcool/Impact servers.
- 8. Deploy a script to auto-start all processes upon physical server startup (UNIX only).
- 9. Set up Windows Services for automatic startup (Windows only).

#### 1.2.3 Configuration

Given the appropriate data access and documented system architecture, configure and test IBM Tivoli Netcool/Impact data sources so that defined data sources are operational, with emphasis on the following steps:

- 1. Obtain access information (host name/IP, database name/instance, schema, login, and ports).
- Obtain Web Services definition library files and XML schema definitions, if applicable.
- 3. Configure IBM Tivoli Netcool/Impact data sources per the documented system architecture through IBM Tivoli Netcool/Impact GUI.
- 4. Test the connection to and from the IBM Tivoli Netcool/Impact data source (where applicable).

Given a properly configured and accessible data source and access to the IBM Tivoli Netcool/Impact GUI, create a data type based on that data source so that a configured data type with access to data items within the data type is available, with emphasis on the following steps:

- 1. Select the Data Types tab.
- 2. Select the data source required for the data type from the drop-down list, and then select **Add**.
- 3. In the DataType window, provide a unique name for the data type.
- 4. Select a different display icon, if required.
- 5. Enter the details for the base table of the database and then select **Refresh**.
- 6. Select at least one field of the returned table as the key field.
- 7. Remove the fields from the table if they are not required in the Data Type.
- 8. Select an appropriate field for the Display Name.
- 9. Apply a filter condition to apply to the data source's Data Items, as required.
- 10. Apply a sort order to apply to the data source's Data Items, as required.
- 11. Save the configuration by selecting the Floppy Disk icon.
- 12.Confirm the Data Type's configuration by selecting the View Data Items icon adjacent to the Data Type entry.

Given that IBM Tivoli Netcool/Impact has been properly installed, the name of the IBM Tivoli Netcool/Impact Server is known, a valid user name and password is known (optionally encrypted), a policy to execute is known, and (optionally) a "@" Identifier and "@" Serial if policy contains a call to the Returnevent function, configure command-line activation of policies requirements so that policies may be executed from the command line, with emphasis on the following steps:

- 1. Determine the name of IBM Tivoli Netcool/Impact server.
- 2. Determine a valid user name/password combination.
- 3. Determine a Policy Name.
- 4. Optionally, determine "@" Identifier and "@" Serial fields.
- 5. Optionally, determine an encrypted password.

Given the anticipated event volumes and multi-policy processing, set up an Event Processor so that processing of events is optimized with minimal conflicts, with emphasis on the following steps:

- 1. Determine if event volumes warrant additional multi-threading of policy processing.
- 2. Select the Event Processor Service from the Service Status pane.
- 3. Enter the desired number of simultaneous threads.

- 4. Enter the desired maximum number of events per query.
- 5. Enter the desired event fetch rate.
- 6. Select whether to enable logging.
- 7. Save the configuration and stop and restart the Event Processor.
- 8. Determine whether to enable event locking within the Event Readers, and update the Event Readers accordingly.

Given that access to data sources has been configured, data types have been created and policies written, identify the IBM Tivoli Netcool/Impact Service configuration requirements so that Event Readers can be created within IBM Tivoli Netcool/Impact, with emphasis on the following steps:

- ► Determine Service Name.
- ► Determine Data Source.
- ► Determine Polling Interval.
- Determine Restrict Fields.
- Determine the IBM Tivoli Netcool/Impact Server startup requirements.
- Determine the logging and reporting requirements.
- Determine the Event Mapping requirements.

Given that Services are available, configure Services so that logging is available, with emphasis on the following steps:

- 1. For any Service where logging is an option, select the Service to access the configuration window.
- 2. Review the Service Log: write to file option and select or de-select it as required and then click **OK**.
- 3. For the Policy Logger Service, select or de-select the following log options, as required:
  - Highest Log Level (0-3)
  - All SQL statements
  - Pre-execution Action Module Parameters
  - Post-execution Action Module Parameters
  - All Action Module Parameters
- 4. For the Service Log; write to file option, append Thread Name to Log File Name and append Policy Name to Log File Name. For any service where logging has been selected, check the log files at \$NCHOME/impact/log/.log.

#### 1.2.4 Development

Given functional requirements, an architectural design, and policy objectives, draft pseudo-code so that a document that fully describes the critical elements of a policy defined in the functional requirements is created, with emphasis on the following steps:

- 1. Review the requirements.
- 2. Formulate a development plan for accessing the appropriate data.
- 3. Formulate a development plan for testing the policy during development.
- 4. Verify access to test systems and data.
- 5. Define the event or trigger that will execute the policy.
- 6. Identify any relationships or interactions with other events or policies.
- 7. Create pseudo-code or a flow chart.

Given that the IBM Tivoli Netcool/Impact planning requirements have been met, proper installation and configuration of the IBM Tivoli Netcool/Impact server has occurred, and a Netcool ObjectServer is running and accessible through administrative access, determine and create Netcool ObjectServer fields to meet functional requirements so that additional ObjectServer fields are created, with emphasis on the following steps:

- 1. Determine the policy interaction requirements.
- 2. Determine the event suppression flow requirements.
- 3. Determine the communication states between ObjectServer and IBM Tivoli Netcool/Impact requirements.
- 4. Determine column name and data type.
- 5. Log in to Netcool ObjectServer with the administration application.
- 6. Use the administration tool to create ObjectServer columns.
- 7. Optionally, use SQL Interactive to create ObjectServer columns.

Given pseudo-code or policy flow statements with properly configured data sources and data types, develop and unit test IBM Tivoli Netcool/Impact policies so that functional requirements are met, with emphasis on the following steps:

- 1. Determine the functional category.
  - Event Enrichment policy
  - X in Y Policy
  - Event Notification
  - Event Gateway or third-party integration
  - Event Correlation
  - Event Consolidation or Service Event
  - Message (E-mail Reader or IM reader) policy
  - Scheduling policy

- 2. Develop IBM Tivoli Netcool/Impact policies.
  - Data types: User defined variables, arrays, and context
  - Control structures: IF statements, WHILE statements
  - Functions: action, parser, and user defined
  - Logging: policy context and checkpoints
- 3. Unit test functions or functional blocks.

Given the required tasks, configure IBM Tivoli Netcool/Impact services so that policies are automatically executed, with emphasis on the following steps:

- 1. Determine which services and methods are necessary to automate the policy execution.
- 2. Determine the triggers and filters needed to automate policy execution within a service, but avoid conflicts between policies.
- 3. Obtain the required data to enable IBM Tivoli Netcool/Impact Service.
  - E-mail server information (if an E-mail Reader service is necessary)
  - Event Listener Service information (if an Event Listener Service is necessary)
  - ObjectServer connectivity information (if an Event Reader Service is necessary)
  - Instant Messaging Jabber Server Information (If a Jabber Reader Service is necessary)
  - Java[™] Messaging Bus information (if a JMS Message Listener Service is necessary)
- 4. Determine the frequency of the policies' execution.
- 5. Configure new or existing services.
- 6. For an Event Reader Service, create event mapping filters, avoiding conflicts between policies, and achieve efficient processing.

Given that an Operator View is required, create and configure an Operator View, with emphasis on the following steps:

- 1. Select New Operator View from the Operator View tab.
- 2. In the Operator View configuration window, define the Operator View, as necessary, using the following tabs:
  - Name & Layout
  - Action Panel
  - Information Groups
- 3. Save the Operator View by selecting the Floppy Disk icon.

- 4. If a custom Operator View is required, modify the Operator View files as follows:
  - Edit the Operator View definition policy.
  - Edit the Operator View HTML file.
- 5. Review the Operator View from a Web browser.

Given the system requirements and a functioning IBM Tivoli Netcool/Impact system with deployed policies and services, plan, set up, implement, and review an IBM Tivoli Netcool/Impact systems test with relevant stakeholders so that properly configured test systems and data are verified, with emphasis on the following steps:

- 1. Review the technical architecture and requirements.
- 2. Map the requirements into system test cases.
- 3. Create a system test plan.
- 4. Get stakeholder approval for a system test plan.
- 5. Set up the required test systems.
- 6. Set up the required test data.
- 7. Execute the system test.
- 8. Document the test results.
- 9. Review the test results with all stakeholders and developers.

Given the complete project requirements, identify the appropriate parties and available mechanisms to formulate and document a policy exception plan so that a document that defines the project specific policy exception plan is created, with emphasis on the following steps:

- 1. Review the requirements.
- 2. Identify the available exception notification capabilities.
- 3. Identify the parties who must be notified.
- 4. Formulate and publish a plan for policy exceptions.
- 5. Define a standard, programmatic mechanism for handling policy exceptions.
- 6. Produce a document that defines how to handle exceptions within policies.

#### 1.2.5 Administration

Given that IBM Tivoli Netcool/Impact planning requirements have been met, proper installation and configuration of the IBM Tivoli Netcool/Impact Server has occurred, an IBM Tivoli Netcool/Impact Server is running and accessible, and project members have been determined, configure IBM Tivoli Netcool/Impact projects so that functional requirements are met, with emphasis on the following steps:

- 1. Determine the project members from the Global Repository based on functional requirements.
- 2. Log in to IBM Tivoli Netcool/Impact.
- 3. Select the **Project** tab.
- 4. Enter the name of the project and select the New button.
- 5. Select **Members** from the Global Repository for Data Sources, Data Types, Policies, Services, and Operator Views.
- 6. Select **OK** and save the project.

Given detailed functional requirements related to user access and roles, configure users and roles in Security Manager so that users have appropriate access to IBM Tivoli Netcool/Impact, with emphasis on the following steps:

- 1. Interview stakeholders to determine the desired user roles.
- 2. Determine that users that require access to IBM Tivoli Netcool/Impact.
- 3. Set up Security Manager with the appropriate role(s).
- 4. Create IBM Tivoli Netcool/Impact users in Security Manager.
- 5. Assign roles to users in Security Manager to match requirements.

Given the technical architecture and requirements for a functioning IBM Tivoli Netcool/Impact deployment, deploy IBM Tivoli Netcool/Impact objects so that deployment policy usage is documented and objects are deployed to other servers and clusters as needed, with emphasis on the following steps:

- 1. Determine the method for moving objects.
  - Use the nci import/export command.
  - Use the deployment function
- 2. Review the technical requirements.
- 3. Create deployment policies needed based on architecture.
- 4. Test the deployment policies.
- 5. Document the deployment policy usage.
- 6. Deploy objects as needed.

Given that IBM Tivoli Netcool/Impact Server A has been configured, export data from IBM Tivoli Netcool/Impact Server A so that the data can be Imported into IBM Tivoli Netcool/Impact Server B, with emphasis on the following steps:

- 1. Ensure that IBM Tivoli Netcool/Impact Server A is running.
- 2. From IBM Tivoli Netcool/Impact Server A's user interface, select the **Global** tab and then select the **Clear all Source Control File Locking** button.
- 3. Export the data from IBM Tivoli Netcool/Impact Server A by running the **nci_export** command and specifying a location for the export directory.
- 4. Copy the export directory to a temporary location within IBM Tivoli Netcool/Impact Server B's host server.
- 5. Ensure that IBM Tivoli Netcool/Impact Server B is running.
- 6. From IBM Tivoli Netcool/Impact Server B's user interface, select the **Global** tab and then select the **Clear all Source Control File Locking** button.
- Import data into IBM Tivoli Netcool/Impact Server B using the nci_import command and specify the location of IBM Tivoli Netcool/Impact Server A's exported data.
- 8. Check the IBM Tivoli Netcool/Impact Server B's log file for errors.
- 9. Shut down and re-start IBM Tivoli Netcool/Impact Server B in order to view the imported data in the user interface.

Given that you have a running IBM Tivoli Netcool/Impact server and any EAR file modification requirements, modify the IBM Tivoli Netcool/Impact EAR file and redeploy it so that the WebSphere Application Server Community Edition server is running the newly modified version, with emphasis on the following steps:

- 1. Determine if this is to be a *hot* or *cold* redeploy.
- 2. Execute the nci_ear_editor command.
- 3. Redeploy the EAR.
  - a. For a *hot* redeployment, verify that the EAR was redeployed to the wasce.log file.
  - b. For a *cold* redeployment, stop WebSphere Application Server Community Edition, remove the previous deployment files, restart WebSphere Application Server Community Edition and review the wasce.log file.

Given that the IBM Tivoli Netcool/Impact EAR file has been modified, deploy a new EAR file so that IBM Tivoli Netcool/Impact has the new functionality, with emphasis on the following steps:

- 1. Access and log in to the WebSphere Application Server Community Edition management console from a Web browser.
- 2. Select **Application EARs** in the Console Navigation pane.

- 3. Uninstall the IBM Tivoli Netcool/Impact component.
- 4. Select Deploy New in the Console Navigation pane.
- 5. Navigate to the location of the modified EAR file and then select the file.
- 6. Select **Start application after installation**, as required, and then select the **Install** button.
- 7. Confirm that the application has started within WebSphere Application Server Community Edition.
- 8. Log out of the WebSphere Application Server Community Edition console.
- 9. Check log files for errors and resolve them as appropriate.
- 10.Log in to IBM Tivoli Netcool/Impact and confirm that the new functionality is available.

#### 1.2.6 Troubleshooting and performance tuning

Given an installed IBM Tivoli Netcool/Impact system based on technical architecture specifications, review the deployment documentation and check various outputs of systems so that a properly installed IBM Tivoli Netcool/Impact system is verified, with emphasis on the following steps:

- Review the technical architecture and requirements.
- Review the installation logs for errors.
- Review the external systems logs for errors.
- Review the IBM Tivoli Netcool/Impact server logs for errors.

Given an operating IBM Tivoli Netcool/Impact server and performance requirements, analyze reports, identify potential bottlenecks, and create recommendations for those bottlenecks so that a document describing performance bottlenecks is created that recommends appropriate actions to mitigate the issues, with emphasis on the following steps:

- 1. Enable the appropriate monitoring reports.
- 2. Review the monitoring reports and identify problem areas.
- 3. Map problem performance areas to specific policies or services.
- 4. Review the policy code or service configuration.
- 5. Identify external system bottlenecks.
- 6. Document the potential bottlenecks and consult with the policy developers or system administrators for review and recommendations.
- 7. Consolidate and publish the recommendations.

Given that a functional IBM Tivoli Netcool/Impact Installation is running, and logging standards and IBM Tivoli Netcool/Impact Services are configured, utilize IBM Tivoli Netcool/Impact and the system tools so that performance issues may be assessed, with emphasis on the following steps:

- Utilize the server logs netcool.log, (Server)_(Service).log, (Server)_Policylogger.log, and (Server)_Policylogger_PolicyName.log.
- 2. Utilize the operating system tools **netstat**, **uptime**, **iostat**, **sar**, **vmstat**, and **top**.
- 3. Utilize the Self-Monitoring Service.
  - a. Determine when to utilize it.
  - Enable the service either using the GUI (IBM Tivoli Netcool/Impact Self-Monitoring Service) or CLI (IBM Tivoli Netcool/Impact Clustered mode) or (IBM Tivoli Netcool/Impact Service)_selfmonitoring.props.
  - c. Analyze the Monitoring Types: Memory Status, Queue Stat, Data Source Status, and Cluster Status.
- 4. Utilize the Reporting Tools.
  - a. Determine when to utilize them.
  - b. Enable them using the IBM Tivoli Netcool/Impact Policy Logger Service.
  - c. Analyze Network and Network Operators Reports: Node Efficiency Report and Operator Efficiency Report.
  - d. Analyze IBM Tivoli Netcool/Impact Configuration Reports: Action Efficiency Report, Action Error Report, Impact Profile Report, Impact Return on Investment Report, Policy Efficiency Report, and Policy Error Report.

Given known bugs or issues, test the system and review logs so that bugs and issues are isolated within the system, with emphasis on the following steps:

- 1. Review the bug tracking list or known issues.
- 2. Increase the logging level in the Policy Logger Service.
- 3. Replicate the bug.
- 4. Review policy logs, Event Reader logs, and server logs to determine where bug(s) exist in the policy.
- 5. Review the Event Reader configuration, if applicable.
Given that IBM Tivoli Netcool/Impact CVS is generating error messages in the log files, troubleshoot IBM Tivoli Netcool/Impact CVS using the **nci_version_control** command so that error messages are not generated, with emphasis on the following steps:

- If CVS is preventing a policy or other object (data source or data type) from being configured, use the IBM Tivoli Netcool/Impact CVS script, nci_version_control, to modify the status of the file. The script can be used for the following actions:
  - Check in a file.
  - Check out a file.
  - Add a file.
  - Remove a file.
  - Uncheck a file.
  - Rename a file.

## **1.3 Certification achieved**

The test IBM Tivoli Netcool/Impact V4.0 implementation (#938) is a prerequisite for achieving the following certifications:

- 1.3.1, "Tivoli Netcool/Impact V4.0" on page 23
- 1.3.2, "Tivoli Fault Management Solutions 2008" on page 25
- 1.3.3, "IBM Service Management Network and Service Assurance 2009" on page 26

#### 1.3.1 Tivoli Netcool/Impact V4.0

Tivoli Netcool/Impact V4.0 is an IBM Certified Deployment Professional certification.

#### Target audience

An IBM Certified Deployment Professional - Tivoli Netcool/Impact V4.0 is an individual who has demonstrated the ability to implement and support an IBM Tivoli Netcool/Impact solution. It is expected that this person is able to perform the following tasks independently a majority of the time, and in some situations, take leadership and provide mentoring to peers. It is expected that this person will be able to perform these tasks with limited assistance from peers, product documentation, and vendor support services.

#### **Recommended prerequisite skills**

Key areas of competency

- ► Describe the IBM Tivoli Netcool/Impact V4.0 architecture and components.
- Plan and design an IBM Tivoli Netcool/Impact V4.0 solution based on customer requirements/environment.
- ► Install and configure prerequisites to IBM Tivoli Netcool/Impact V4.0.
- Install and configure IBM Tivoli Netcool/Impact V4.0 infrastructure components.
- Use available interfaces to configure and administer the IBM Tivoli Netcool/Impact V4.0 environment.
- Perform performance tuning and problem determination for IBM Tivoli Netcool/Impact V4.0.
- Develop and deploy IBM Tivoli Netcool/Impact policies and services.

#### **Required prerequisites**

- ► Experience administering an IBM Tivoli Netcool/Impact V4.0 at skill level 4
- Knowledge of IBM Tivoli Netcool/Impact V4.0 at skill level 4
- Knowledge of scripting languages (shell scripting, rules files, regular expressions, and PERL) at skill level 3
- Knowledge of operating systems (UNIX and Windows) at skill level 3
- Knowledge of networks and network management at skill level 3
- Knowledge of IBM Tivoli Netcool/OMNIbus at skill level 3
- Knowledge of database structures at skill level 3
- Knowledge of operating system utilities (ftp, telnet, sftp, ssh, and text editors) at skill level 2
- Knowledge of SQL, PostgreSQL, and other ANSI compliant sources at skill level 2
- Knowledge of HTML at skill level 2
- Knowledge of WebSphere Application Server Community Edition at skill level 1
- Knowledge of Web services (XML, SOAP, and WSDL), if applicable, at skill level 1
- Knowledge of Netcool Security Manager at skill level 1
- Knowledge of CVS at skill level 1

Skill descriptions:

1	Basic Skill/Knowledge: Familiarity with basic functionality and concepts. May need to rely on assistance from documentation or other resources.
2	Working Skill/Knowledge: Working knowledge of functionality and concepts. Can use product or explain concepts with little or no assistance.
3	Advanced Skill/Knowledge: Substantial experience with functionality or concepts. Can teach others how to use functionality or explain concepts.
4	Expert Skill/Knowledge: Extensive and comprehensive experience with functionality or concepts. Can create or customize code, architecture, or processes.

This certification requires the IBM Tivoli Netcool OMNIbus V7.x Implementation or Tivoli Enterprise Console® 3.9 Implementation Certification as well as passing the IBM Tivoli Netcool/Impact V4.0 Implementation exam.

#### **Requirements**

This certification requires two tests.

- Any one of the following tests:
  - Test 594 IBM Tivoli Enterprise Console V3.9 Implementation
  - Test 901 IBM Tivoli Netcool/OMNIbus V7.1 Implementation
  - Test 933 IBM Tivoli Netcool/OMNIbus V7.2 Implementation
- ► Test 938 IBM Tivoli Netcool/Impact V4.0 Implementation

### 1.3.2 Tivoli Fault Management Solutions 2008

Tivoli Fault Management Solutions 2008 is an IBM Certified Advanced Deployment Professional certification.

#### **Target audience**

An IBM Certified Advanced Deployment Professional - Tivoli Fault Management Solutions 2008 is an individual who has demonstrated a higher level of implementation knowledge and skill both in breadth and in depth in the IBM Tivoli Fault Management solutions area.

#### **Requirements**

This certification requires four tests.

- Any one of the following tests:
  - Test 901 IBM Tivoli Netcool/OMNIbus V7.1 Implementation
  - Test 933 IBM Tivoli Netcool/OMNIbus V7.2 Implementation
- Test 922 IBM Tivoli Netcool/Webtop V2.0
- Any two of the following tests:
  - Test 890 IBM Tivoli Monitoring V6.1 Implementation
  - Test 897 IBM Tivoli Network Manager IP Edition V3.7 Implementation
  - Test 905 IBM Tivoli Composite Application Manager for WebSphere V6.1
  - Test 920 IBM Tivoli Composite Application Manager for Response Time V6.2 Implementation
  - Test ITIL® Information Technology Infrastructure Library --Foundations™
  - Test 436 IBM Tivoli Business Service Manager V4.1.1 Implementation
  - Test 938 IBM Tivoli Netcool/Impact V4.0 Implementation
  - Test 908 IBM Tivoli Monitoring V6.2 Implementation

#### 1.3.3 IBM Service Management Network and Service Assurance 2009

IBM Service Management Network and Service Assurance 2009 is an IBM Certified Advanced Deployment Professional certification.

#### **Target audience**

An IBM Certified Advanced Deployment Professional - IBM Service Management Network and Service Assurance 2009 is an individual who has demonstrated a higher level of implementation knowledge and skill both in breadth and in depth in the IBM Tivoli Network and Service Assurance solutions area.

#### **Requirements**

This certification requires 4 tests:

- ► Test 000-922 IBM Tivoli Netcool/Webtop V2.0
- Test 000-933 IBM Tivoli Netcool/OMNIbus V7.2 Implementation
- ► Test 000-938 IBM Tivoli Netcool/Impact V4.0 Implementation

- Any 1 of the following tests:
  - Test ITIL Information Technology Infrastructure Library -- Foundations
  - Test 000-430 IBM Tivoli Netcool Service Quality Manager V4.1.1 Implementation
  - Test 000-434 IBM Tivoli Netcool Performance Manager for Wireless V9.1.2 Implementation
  - Test 000-931 IBM Tivoli Netcool/Proviso V4.4.1 Implementation

## **1.4 Recommended study resources**

Courses and publications are offered to help you prepare for the certification tests. The courses are recommended, but not required, before taking a certification test. If you wish to purchase Web-based training courses or are unable to locate a Web-based course or classroom course at the time and location you desire, please feel free to contact one of our delivery management teams at:

- Americas: tivamedu@us.ibm.com
- EMEA: tived@uk.ibm.com
- ► AP: tivtrainingap@au1.ibm.com

Note that course offerings are continuously being added and updated. If you do not see the course(s) below listed in your geography, please contact the delivery management team.

#### 1.4.1 Courses

Course title: IBM Tivoli Netcool/Impact 4.0 Administration and Implementation

Course duration: 5 Days

Course number:

Abstract: This 5 day course is designed to address all aspects of IBM Tivoli Netcool/Impact. Students will undertake practical exercises throughout the course, built to teach both common policies in use and to cover key development concepts. Students should plan on attending the entire course, as concepts will be built upon throughout the length of the course. In the lab exercises, an IBM Tivoli Netcool/OMNIbus and sample relational database will be available in class to use with the Tivoli Netcool/Impact product. After some introductory material, the class will focus on learning a new concept then applying that concept in a policy building lab. Each lab will progressively build on previous labs, offering the students not just the opportunity to input policies, but to debug and explore how each works.

# 2

## Planning

This chapter discusses IBM Tivoli Netcool/Impact implementation planning issues. We discuss these issues in the following sections:

- ► 2.1, "IBM Tivoli Netcool/Impact architecture" on page 30
- ► 2.3, "IBM Tivoli Netcool/Impact policy planning" on page 41
- ► 2.4, "User and authorization scheme" on page 42
- ► 2.5, "Application dependency" on page 43
- ► 2.6, "Failover configuration" on page 44

## 2.1 IBM Tivoli Netcool/Impact architecture

The IBM Tivoli Netcool/Impact architecture is discussed in the following sections:

- ► 2.1.1, "IBM Tivoli Netcool/Impact components" on page 30
- ► 2.1.3, "IBM Tivoli Netcool/Impact deployment types" on page 35

#### 2.1.1 IBM Tivoli Netcool/Impact components

Figure 2-1 shows the IBM Tivoli Netcool/Impact components. Understanding the components' roles and functions is critical to understanding which components are needed for a specific solution.



Figure 2-1 Components

As shown in Figure 2-1 on page 30, IBM Tivoli Netcool/Impact solution has the following major components:

► IBM Tivoli Netcool/Impact server

The IBM Tivoli Netcool/Impact server is the primary component of an IBM Tivoli Netcool/Impact solution. This component manages the data model, services, and policies that make up an IBM Tivoli Netcool/Impact implementation and runs the policies in real time in response to events that occur in the environment.

An IBM Tivoli Netcool/Impact server has a runnable subcomponent called the JRExec server that you can use to run external commands, scripts, and applications from within a policy.

In Version 4.0, the IBM Tivoli Netcool/Impact server runs as an application instance inside a Java application server. By default, the IBM Tivoli Netcool/Impact server runs inside WebSphere Application Server Community Edition, which is the application server that is installed automatically as part of the IBM Tivoli Netcool/Impact installation.

IBM Tivoli Netcool/Impact GUI server

The IBM Tivoli Netcool/Impact GUI server is the component that is responsible for managing the IBM Tivoli Netcool/Impact GUI. The GUI server brokers HTTP requests from users' Web browsers and returns the graphical user interface views that you use to work with IBM Tivoli Netcool/Impact data models services and policies. Only administrators have a need to access the GUI.

The GUI server has a subcomponent called the IBM Tivoli Netcool/Impact nameserver that provides application registry functionality for IBM Tivoli Netcool/Impact components. IBM Tivoli Netcool/Impact components use the nameserver to locate and establish connections to each other. The IBM Tivoli Netcool/Impact server also uses the name server for server clustering.

Like the IBM Tivoli Netcool/Impact server, the GUI server runs as a hosted application inside a Java application server. By default, the GUI server runs inside WebSphere Application Server, which is the application server that is installed automatically as part of the IBM Tivoli Netcool/Impact installation.

Netcool Security Manager

The Netcool Security Manager is a component that provides user authentication for applications in Netcool applications. When a user logs into a Netcool application, it passes the login request to the Security Manager for authentication. Security Manager authenticates the request against IBM Tivoli Netcool/OMNIbus, LDAP, or NIS authentication sources and then returns the authentication status to the originating application. Security Manager is a runnable, stand-alone server application that you must install, run, and manage separately from IBM Tivoli Netcool/Impact. Unlike the IBM Tivoli Netcool/Impact server and the IBM Tivoli Netcool/Impact GUI server, the Security Manager does not run as an application instance inside WebSphere Application Server.

Netcool database

The Netcool database is a specially configured version of PostgreSQL that is prepared for use with IBM Tivoli Netcool/Impact and other Netcool products. IBM Tivoli Netcool/Impact uses the database to store the underlying data used by the GUI reporting tools. You can also use the database to store other kinds of information used by IBM Tivoli Netcool/Impact.

Data Source Adaptors

Data Source Adaptors (DSAs) are software components that IBM Tivoli Netcool/Impact uses to communicate with external data sources. The ObjectServer DSA is a required component for IBM Tivoli Netcool/Impact. Other DSAs are installed as needed. DSAs broker information to and from SQL databases, LDAP servers, JMS topics and queues, and software systems that allow communication using Web services APIs. DSAs also allow IBM Tivoli Netcool/Impact to parse XML strings and documents, communicate with Web servers using HTTP, and communicate with custom applications through generic socket transactions. For a complete list of available DSAs, refer to the *IBM Tivoli Netcool/Impact DSA Reference Guide*, SC23-8856, found at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm .netcool_impact.doc/im402dsa.pdf

The IBM Tivoli Netcool/Impact server runs some services that are specialized and perform specific tasks. These services has their own specific functions:

- IBM Tivoli Netcool/OMNIbus Event Listeners: Hooks IBM Tivoli Netcool/Impact into the IBM Tivoli Netcool/OMNIbus accelerated event notification (AEN) system. High-priority events coming in from the IBM Tivoli Netcool/OMNIbus AEN system go directly to the top of the event queue for immediate processing. An IBM Tivoli Netcool/OMNIbus Event Listener should not be used as a replacement for the IBM Tivoli Netcool/OMNIbus Event Reader, but instead as a way to ensure high priority events are processed first.
- IBM Tivoli Netcool/OMNIbus Event Readers: These are services that monitor an IBM Tivoli Netcool/OMNIbus ObjectServer event source. When an Event Reader discovers a new, updated, or deleted alert in the ObjectServer, it retrieves the alert and sends it to an event queue, where it waits to be handled by the Event Processor.

- E-mail Readers: These readers receive e-mail from a POP service to activate a policy.
- E-mail Senders: This component sends e-mail using an external SMTP service.
- ► Policy Activators: This activator runs policies at timed intervals.
- Hibernating Policy Activator: This is a special activator service that is responsible for waking hibernations at timed intervals. You use the Hibernating Policy Activator with X events in Y time solutions and similar solutions.
- Event Processor: This processor is responsible for managing events coming into IBM Tivoli Netcool/Impact from Event Reader, Event Listener, and E-mail Reader Services. The Event Processor manages the incoming event queue and is responsible for sending queued events to the Policy Engine for processing.
- Policy Logger: The service responsible for managing the policy log. The policy log is a text stream used by IBM Tivoli Netcool/Impact to record messages during the run time of a policy. The policy log contains both system messages issued by IBM Tivoli Netcool/Impact and messages that you define in a policy.
- Command-Line Manager: The service that manages the IBM Tivoli Netcool/Impact CLI.
- CORBA name service: The service that manages the CORBA interface used by some Mediator DSAs.
- Jabber Service: This service sends instant messages and receives them. An user-defined service called Jabber Reader can be created.

Another component that could be required is IBM Tivoli Netcool/OMNIbus, which is used whenever there is a need for event handling within the IBM Tivoli Netcool/Impact deployment, but this is not a absolute requirement, as the policies can be used to do other things, such as correlating between different databases.

### 2.1.2 Communication configuration

Some of the IBM Tivoli Netcool/Impact components communicate with each other using the TCP/IP port. When those components are installed on different servers, you must ensure that the firewall between those components will allow the communication. Figure 2-2 shows this configuration.



Figure 2-2 Communication interaction

Using Figure 2-2 as a reference, Table 2-1 on page 35 lists the ports that must be available for the solution to work.

Table 2-1 Required ports

	Usage	Description	Default
	HTTP port	HTTP port for accessing IBM Tivoli Netcool/Impact GUI and Netcool name server.	8080
	HTTPS port	HTTPS port for SSL based IBM Tivoli Netcool/Impact GUI.	8443
	Naming port	RMI naming port.	1099
-	EJB™ port	Enterprise Java Beans port.	4201
orts ⁶	COS Naming port	Common Object Services naming port.	5050
ed p	JMS port	Java Message System port.	61616
relat	Derby port	Apache Derby port.	1527
lere	Directory service port	Apache Directory Server port.	1389
bSpł	JAAS port	Java Authentication and Authorization Service port.	4242
We	Tomcat AJP port	Tomcat AJP connector port.	8009
	Security Manager port	Port used by the Security Manager.	8077
	Netcool database port	Port used by the Netcool database server.	5435
	IBM Tivoli Netcool/OMNIbus port	Default primary port for NCOMS ObjectServer.	4100

a. Except for the HTTP and HTTPS ports, WebSphere related ports are not visible for the solution operation; they are mainly for internal communication within WebSphere.

## 2.1.3 IBM Tivoli Netcool/Impact deployment types

An IBM Tivoli Netcool/Impact solution can be installed in several different configurations. An IBM Tivoli Netcool/Impact installation can be categorized as a single-system installations or a distributed installation:

Single-system deployments: A single-system deployment consists of the IBM Tivoli Netcool/Impact server, the IBM Tivoli Netcool/Impact GUI server, and the Netcool Security Manager installed on a single system in your environment. A single-system deployment is suitable for testing and demonstrating IBM Tivoli Netcool/Impact. Distributed deployments: A distributed deployment consists of one or more instances of the IBM Tivoli Netcool/Impact server, and the IBM Tivoli Netcool/Impact GUI server and the Netcool Security Manager installed across different systems in your environment. A distributed deployment is suitable for most real-world implementations of IBM Tivoli Netcool/Impact.

Once the individual server configuration has been decided, you must decide whether to establish clustering. A cluster implementation enables IBM Tivoli Netcool/Impact systems with redundant systems, which allows resiliency of the processes. There are two types of IBM Tivoli Netcool/Impact clustering systems:

- Basic IBM Tivoli Netcool/Webtop cluster: In this configuration, only the IBM Tivoli Netcool/Impact server is put into a cluster. The supporting subsystems, such as Security Manager and the ObjectServer, are not put into a cluster.
- Full IBM Tivoli Netcool/Webtop cluster: In this configuration, all components of the IBM Tivoli Netcool/Impact solution are put into a cluster, including the Security Manager, probes, and ObjectServer.

## 2.2 Important concepts

This section discusses the important concepts of IBM Tivoli Netcool/Impact. The definition of the terms are presented according to the IBM Tivoli Netcool/Impact categories. The content is covered in the following sections:

- ► 2.2.1, "IBM Tivoli Netcool/Webtop services" on page 36
- 2.2.2, "Data models" on page 38
- 2.2.3, "Data Service Adapter (DSAs)" on page 39

#### 2.2.1 IBM Tivoli Netcool/Webtop services

Although we have explained the major components of IBM Tivoli Netcool/Impact in 2.1, "IBM Tivoli Netcool/Impact architecture" on page 30, here we list the services that are available. A service is a runnable sub-component of IBM Tivoli Netcool/Impact that you control from within the IBM Tivoli Netcool/Impact GUI:

#### **Command Execution Manager**

Manages remote command execution using the CommandResponse function in the IPL.

#### **Command-Line Manager**

Manages the IBM Tivoli Netcool/Impact command-line interface.

Corba Name Service	Provides the CORBA naming functionality for mediator DSAs.
Database Listener	Listens for incoming messages from an SQL database data source and then triggers policies based on the incoming message data.
E-mail Reader	Polls a POP mail server at intervals for incoming e-mail and then triggers policies based on the incoming e-mail data.
E-mail Sender	Sends e-mail by way of an SMTP mail server.
Event Processor	Manages events coming into IBM Tivoli Netcool/Impact using Event Reader, Event Listener and E-mail Reader services. The Event Processor manages the incoming event queue and is responsible for sending queued events to the Policy Engine for processing.
Event Reader	Monitors an event source for new, updated, or deleted events and triggers policies based on the event data.
Event Source	A data source that stores and manages events. Most commonly, the event source used by IBM Tivoli Netcool/Impact is the ObjectServer database.
Generic Event Lister	ner
	Listens to an external data source for incoming events and triggers policies based on the event data.
Generic Event Read	er
	Monitors an SQL database event source for new or modified events and triggers policies based on the event information.
Hibernating Policy A	ctivator
	This component is responsible for waking hibernating policies.
Jabber Reader	Listens to external instant messaging servers for messages and triggers policies based on the incoming message data.
Jabber Service	Sends instant messages to instant messaging clients, such as AOL Instant Messenger and Yahoo! Messenger, by way of a Jabber server.
Policy Activator	Runs a specified policy at intervals that you define.
Policy Logger	Writes messages to the policy log.

#### **Precision Event Listener**

Listens to the Netcool/Precision application for incoming messages and triggers policies based on the message data.

#### Self-Monitoring Service

Monitors IBM Tivoli Netcool/Impact for memory and other status conditions and reports them to the IBM Tivoli Netcool/OMNIbus ObjectServer as events.

#### **Standard Event Reader**

A standard Event Reader is a IBM Tivoli Netcool/Impact service that monitors an IBM Tivoli Netcool/OMNIbus ObjectServer database for new, updated, or deleted events and triggers policies based on the event data.

#### 2.2.2 Data models

This section describes data models in IBM Tivoli Netcool/Impact. A data model is an abstract representation of the business data and meta data used in an IBM Tivoli Netcool/Impact installation. A data model contains data sources, data types, links, and event sources. The data models are shown below:

#### Data Source Adapter

	A Data Source Adaptor (DSA) is a component of IBM Tivoli Netcool/Impact that allows the application to access data stored in an external source of data.
Data source	A data source is an element of an IBM Tivoli Netcool/Impact data model that represents an external source of data (for example, a relational database). Each database must be defined as a separate data sources.
Data type	A data type is an element of an IBM Tivoli Netcool/Impact data model that represents a set of data stored in a data source (for example, a table or view in a relational database). Each distinct table must be defined as a separate data type.
Data item	A data item is an element of a IBM Tivoli Netcool/Impact data model that represents an actual unit of data stored in a data source (for example, a row in relational database table).
Link	A link is an element of an IBM Tivoli Netcool/Impact data model that defines a relationship between data types or data items. A link can be static or dynamic.

Event	An event is a set of data that represents a status condition or an activity that has occurred in your environment. Most commonly, events originate with Netcool probes and monitors and are stored in the IBM Tivoli Netcool/OMNIbus ObjectServer database.
Exception	An exception is an occurrence during run time that changes the normal flow of policy execution.
Field	A field is a single named unit of data in an IBM Tivoli Netcool/Impact event or data item.
Key field	A key field is a field that uniquely identifies a data item in a data type.
Data caching	This is storage of external data in memory for quick retrieval. This is useful for buffering small, heavily used lookup tables. Larger lookup tables may need to be queried as a database table; for example, a table in the PostgreSQL database. A policy can be written to automatically load tables from an external sources.

A filter is an expression that IBM Tivoli Netcool/Impact uses to select data (for example, data items in a data type) from a larger set of data:

LDAP filter	An LDAP filter is an expression that IBM Tivoli Netcool/Impact uses to select data elements located in an LDAP directory tree. The syntax for LDAP filters is specified in RFC 2254.
SQL filter	An SQL filter is an expression that IBM Tivoli Netcool/Impact uses to select rows in a database table. The syntax for the filter is similar to the contents of an SQL WHERE clause.

## 2.2.3 Data Service Adapter (DSAs)

A Data Service Adapter defines how a data source will be accessed. The DSAs are as follows:

Flat File DSA	The Flat File DSA allows retrieval of data stored in a flat file, including a Comma Separated Value (CSV) format. The information needed includes the file path, file name, and separator.
JMS DSA	The JMS DSA is a Data Source Adaptor that allows IBM Tivoli Netcool/Impact to send and receive Java Messaging System (JMS) messages.

LDAP DSA The LDAP DSA is a Data Source Adaptor that allows IBM Tivoli Netcool/Impact to read directory data managed by an LDAP server. Mediator DSAs Mediator DSAs are a type of Data Source Adaptor that allows IBM Tivoli Netcool/Impact to access data provided by third-party systems, devices, and applications. The Cramer Dimension DSA is an example of a mediator DSA. Precision DSA The Netcool/Precision DSA is a Data Source Adaptor that allows IBM Tivoli Netcool/Impact to access data managed by the Netcool/Precision application. SNMP DSA The SNMP DSA is a Data Source Adaptor that allows IBM Tivoli Netcool/Impact to set and retrieve management information stored by SNMP agents. It also allows IBM Tivoli Netcool/Impact to send SNMP traps and notifications to SNMP managers. The Management Information Base (MIB) file that describes the data item that can be retrieved by the DSA must be loaded into \$NCHOME/impact/dsa/snmpdsa/mibs. Socket DSA The Socket DSA is a Data Source Adaptor that allows IBM Tivoli Netcool/Impact to exchange information with external applications using a socket server as the brokering agent. SQL database DSAs SQL database DSAs are Data Source Adaptors that allow IBM Tivoli Netcool/Impact to retrieve information from relational databases and other data sources that provide a public interface by way of Java Database Connectivity (JDBC[™]). SQL database DSAs also allow IBM Tivoli Netcool/Impact to add, modify, and delete information stored in these data sources. Note that a Flat File DSA that uses the CSV format is also considered a Database DSA. Web services DSA The Web services DSA is a Data Source Adapter that allows IBM Tivoli Netcool/Impact to exchange information with external applications that provide a Web services API. Before you can use the Web services DSA, you must compile a Web Services Description Language (WSDL) file using the **nci** compilewsdl command found under \$NCHOME/impact/bin so that the DSA understands the

WSDL being used.

XML DSAThe XML DSA is a Data Source Adapter that allows IBM<br/>Tivoli Netcool/Impact to read XML data from strings and<br/>files and to read XML data from Web servers over HTTP.<br/>The XML DSA requires access to the XSD and DTD files<br/>that describes the XML schema.

## 2.3 IBM Tivoli Netcool/Impact policy planning

IBM Tivoli Netcool/Impact policies are the basic rules about how IBM Tivoli Netcool/Impact processes data. The policy can be defined in IBM Tivoli Netcool/Impact as follows:

- Use different projects for development, testing, and production.
- Determine policy naming convention: This is an import factor in order to distinguish between standard policies and the custom created ones. (this will make upgrading IBM Tivoli Netcool/Impact much easier). A practical example would be to prepend the policy name with an acronym of the company name.
- Have a naming convention for variables.
- Determine data models: data sources, data types, data items, links, and event sources. Refer to 4.1, "Data sources" on page 84 and 4.2, "Data types" on page 90 for more information.
- Determine policy types: event enrichment, x events in y time, event notification, and event gateway. Refer to 4.3, "Policies" on page 95 for more information.
- Determine services: Event Readers/Listeners, E-mail Sender/Readers, and Policy Activators. Refer to 4.5, "Event Processor" on page 104 and 4.4, "Event Reader" on page 98 for more information.
- Determine external requirements: data sources, ObjectServer fields, rules files, and scripts.
- Determine pre-conditions: suppression states, triggering events, and timing.
- Determine post-conditions: suppression states, synthetic/meta events creation, event fields, and notifications.
- Determine logging standards: Required, Info, and Debug. Refer to 4.6, "Logging configuration" on page 106 for more information.

## 2.4 User and authorization scheme

IBM Tivoli Netcool/Impact users are authenticated through the Tivoli Netcool Security Manager. Security Manager is a component of the IBM Tivoli Netcool suite that provides user authentication for applications like IBM Tivoli Netcool/Webtop, IBM Tivoli Netcool/Impact, Netcool/Precision, and Tivoli Business Service Manager. The Security Manager is a stand-alone, runnable server application designed for integration with these products and the Netcool GUI Foundation.

The Security Manager supports the following authentication types:

Native authentication

Native authentication is an authentication scheme where user and group information is stored locally in the Security Manager database. Logins to Netcool products and Tivoli Business Service Manager are authenticated against this internally-stored authentication information.

External authentication

External authentication is an authentication scheme where user and group information is stored externally in one or more authentication sources. Logins to Netcool products and Tivoli Business Service Manager are brokered through the Security Manager during run time. Security Manager supports IBM Tivoli Netcool/OMNIbus ObjectServer and LDAP (including Microsoft Active Directory®) as external authentication sources.

The Netcool Installer allows you to add one external authentication source during installation. To add an external authentication source, run the Security Manager configuration utility and follow the prompts that appear. This utility is launched by a script named ncsm_config located in the \$NCHOME/security/install directory. You can run the configuration utility in GUI mode or in console mode. In GUI mode, the utility presents a series of wizards that guide you through the installation process. In console mode, it prompts you for required information from the command line. If you are running the utility remotely using telnet or another command-line application, you must run it in console mode. After you have run the configuration utility, the external authentication source is enabled in Security Manager. There is no need to perform additional configuration steps.

For more detailed information, see the *Security Manager Installation Guide*, found at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.ne
tcool sm.doc/securitymanagerinstallationguide.pdf

When you configure the Security Manager to authenticate against LDAP authentication sources, it directly accesses the specified LDAP repository and does not require use of the ObjectServer PAM authentication feature as a proxy for authentication requests. The Security Manager can directly connect to all supported LDAP authentication sources, including Microsoft Active Directory.

To add users that have access to the IBM Tivoli Netcool/Impact GUI, add the user to the IMPACT_USER group that was created by the installer. IBM Tivoli Netcool/Impact users are not operators; they are system administrators that are responsible for defining and maintaining policies. Operators may be only interested in the Active Event List (AEL) view of the events that has been processed by IBM Tivoli Netcool/Impact. The AEL view is part of IBM Tivoli Netcool/Webtop instead of IBM Tivoli Netcool/Impact.

IBM Tivoli Netcool/Impact can be configured to use SSL to connect to several components:

- Event data using SSL connections to IBM Tivoli Netcool/OMNIbus
- An IBM Tivoli Netcool/Impact name server using SSL
- An IBM Tivoli Netcool/Impact GUI server using HTTPS

## 2.5 Application dependency

In order to create a good project/policy, you need to perform the following tasks.:

- Review detailed functional requirements: Translate the project requirements into a workflow.
- Assess the application infrastructure: Can we get through firewalls?
- Determine Netcool application dependencies: Precision, Tivoli Business Service Manager, ObjectServers, new fields in ObjectServer, database accesses, data sources, and data types.
- Determine other application dependencies, for example:
  - Are Oracle listeners required?
  - Web services: Do we have the WSDL file?
  - DB2: Which user access is required? (read only?)
  - Jabber: Which account is used?
  - XML: Do we have the XML schema?
  - E-mail: Can we send/receive e-mail?
  - Data Source Adapters: Do we have user name, password, and access information?

Document application dependencies.

## 2.6 Failover configuration

This section discusses some options for providing failover configuration for an IBM Tivoli Netcool/Impact server. The main components involved in the failover configuration are ObjectServer, Security Manager, and IBM Tivoli Netcool/Impact server.



The cluster failover configuration is shown in Figure 2-3.

Figure 2-3 IBM Tivoli Netcool/Impact failover configuration

You should be aware of the following parts of the configuration:

Startup

At startup, each server communicates with the Netcool Application Registry. If no other cluster member is currently registered, it registers itself as the primary server. If a primary server is already registered, it declares itself a secondary server. By default, each secondary server synchronizes its services, data types, data sources, policies, and configuration settings with the primary server before becoming active. Event monitoring

During the event monitoring phase, the primary server queries the ObjectServer at intervals for new and updated events, and (optionally) receives notification from the ObjectServer when an event is deleted. The primary server places the incoming events in an event queue and waits for requests from the secondary servers for event processing. Secondary servers do not query the ObjectServer at any time.

Event processing

During the event processing phase, a primary server processes the events. The primary server can share the load with secondary servers. Each secondary server queries the primary server for events to process. The minimum interval and maximum number of events to retrieve per query are specified in the Event Processor properties file. Similarly, the primary server requests events from its own event queue. For each retrieved event, the server runs the corresponding policy, depending on the filter conditions specified in the event broker. After all the events have been processed, the secondary server queries the primary server again for new events.

► Failover

The secondary servers ping the primary at intervals during run time. This assures the secondary servers that the primary is active and functioning.

If the primary server fails, the first secondary server to become aware of the failure contacts the Netcool Application Registry and registers itself as the new primary. When the original primary server is restarted, it becomes another secondary server. If a secondary server fails, there is no impact on the other servers in the cluster.

Shutdown

If the primary server is manually shut down, the first secondary server to become aware of the failure contacts the Netcool Application Registry and registers itself as the new primary, as in the failover phase above. If a secondary server is shut down, there is no impact on the other servers in the cluster.

The failover configuration for an Impact cluster is automatically propagated across the cluster. Any changes made to any server in the cluster are automatically replicated to all of the other servers. Most of the cluster configuration of the cluster is stored in the <servername>_server.props file under \$NCHOME/impact/etc. An excerpt of the server definition is shown in Example 2-1.

Example 2-1 Sample NCI2_server.props

```
impact.server.timeout=120000
impact.servicemanager.storelogs=true
impact.cluster.name=NCICLUSTER
impact.cluster.pinginterval=6000
impact.cluster.repingcount=3
impact.cluster.resyncbeforestandby=true
impact.replication.receiveupdatesfor.orgnodes=true
impact.replication.receiveupdatesfor.servicestates=true
impact.replication.receiveupdatesfor.types=true
impact.replication.receiveupdatesfor.policies=true
impact.security.host=rh4
impact.security.port=8077
```

impact.security.backup.host.1=rh4_2
impact.security.backup.port.1=8077

impact.logdir=/opt/netcool/impact/log/

# 3

## Installation

We discuss installation of IBM Tivoli Netcool/Impact in this chapter. We discussed in the following sections:

- ► 3.1, "Preparation" on page 48
- ► 3.2, "Basic installation" on page 50
- ► 3.3, "Cluster installation" on page 64
- ► 3.4, "Self-monitoring" on page 73
- ► 3.5, "Startup configuration" on page 79

## 3.1 Preparation

This section contains the following sections:

- ► 3.1.1, "Prerequisites" on page 48
- ▶ 3.1.2, "Parameters" on page 49

#### 3.1.1 Prerequisites

IBM Tivoli Netcool/Impact and Security Manager have different operating system support lists:

- Netcool Security Manager is supported on the following platforms:
  - AIX® 5L™
  - Red Hat® Linux 9.0 and Red Hat Enterprise Server 3.0
  - Sun[™] Microsystems Solaris[™] 7, 8, 9, 10, and 11
  - Microsoft Windows XP, Windows 2000 Server, and Windows 2003 Server
  - Hewlett-Packard HP-UX 11.11
- IBM Tivoli Netcool/Impact server and GUI server are supported on the following platforms:
  - Sun Microsystems Solaris 8, 9, and 10
  - Red Hat Enterprise Server 3.0 and 4.0
  - SUSE® Enterprise 9.3
  - Microsoft Windows 2000 Server, Windows 2003 Server, and Windows XP
  - IBM AIX 5L V5.2 and V5.3
  - Hewlett-Packard HP-UX 11i
- The IBM Tivoli Netcool/Impact GUI runs on the following Web browsers:
  - Microsoft Internet Explorer® 6 and later
  - Netscape 7 and later
  - Mozilla 1.67 and later

You must install the Netcool Security Manager before you install the IBM Tivoli Netcool/Impact server and IBM Tivoli Netcool/Impact GUI server. Netcool Security Manager is a mandatory prerequisite for installing IBM Tivoli Netcool/Impact.

IBM Tivoli Netcool/OMNIbus is a mandatory prerequisite *only* when event manipulation is a requirement of the IBM Tivoli Netcool/Impact deployment.

IBM recommends that the target systems satisfy the following minimum hardware requirements:

- Processor speed: 1 GHz and higher for Intel®, 500 MHz and higher for SPARC or POWER® architectures, and dual and quad configurations supported for all platforms.
- Memory: 1 GB of RAM. For high-load installations where you expect large event storms, you might need as much as 4 GB of RAM.
- Disk space: 1 GB disk space is required for production. An additional 500 MB of space is necessary during installation.

#### 3.1.2 Parameters

When installing on UNIX, refer to the *IBM Tivoli Netcool/Impact Administration Guide*, SC23-8853 for details about the installation process. You can find this book at the following address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.ne tcool_impact.doc/im402ag.pdf

There are special memory considerations that have to be fulfilled, and that book discusses them.

You must configure the operating system kernel so that it fulfills the memory requirements for the PostgreSQL database. In addition, if you are installing on HP-UX, you must also configure the kernel to accommodate the threading requirements of IBM Tivoli Netcool/Impact.

#### 3.1.3 Installation image

The IBM Tivoli Netcool/Impact installation package is distributed in the gzipped tar archive format. To explode the installation package, use one of the following commands:

```
tar -xzf impact40_platform.tar.gz
gunzip -c impact40_platform.tar.gz | tar -xf
```

## 3.2 Basic installation

In this section, we discuss the installation of IBM Tivoli Netcool/Impact in the following sections:

- ► 3.2.1, "Graphical installation" on page 50
- ► 3.2.2, "Other installation types" on page 63
- ▶ 3.2.3, "Installation log file" on page 64
- ▶ 3.2.4, "JRExec service on Windows" on page 64

#### 3.2.1 Graphical installation

The basic installation includes installing a single instance of an IBM Tivoli Netcool/Impact server on a single machine. Multiple machine installation will be discussed in 3.3, "Cluster installation" on page 64. Perform the following steps:

1. Download and unpack the IBM Tivoli Netcool/Impact binaries from Passport Advantage®

(http://www-01.ibm.com/software/howtobuy/passportadvantage/) into a temporary directory.

2. In UNIX, you have to run the installer as a *non*-root user. Run **setuparch.bin**. and click **Next** when the window shown in Figure 3-1 opens.



Figure 3-1 InstallShield Wizard

3. Read the license agreement shown in Figure 3-2 on page 51 and select the I Accept both the IBM and the non-IBM terms radio button and click Next.

	Netcool/Impact - InstallShield Wizard	×
1	Please read the following license agreement carefully.	1
	International Program License Agreement	
X	Part 1 - General Terms	
	BY DOWNLOADING, INSTALLING, COPYING, ACCESSING, OR USING THE PROGRAM YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ACCEPTING THESE TERMS ON BEHALF OF ANOTHER PERSON OR A COMPANY	
Tivoli. software	OR OTHER LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE FULL AUTHORITY TO BIND THAT PERSON, COMPANY, OR LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT AGREE TO THESE TERMS,	
	Read non-IBM terms	
	I accept both the IBM and the non-IBM terms	
	I do not accept the terms in the license agreement	
	Print	
Percentrality		
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel	ľ

Figure 3-2 License Agreement

 Select the destination path of your installation, as shown in Figure 3-3. The default path for UNIX is /opt/netcool and for Windows is C:\Program Files\ibm\netcool. IBM Tivoli Netcool/Impact itself is installed in the impact subdirectory of the installation path.

	Netcool/Impact - InstallShield Wizard
	Click Next to install "Netcool/Impact" to this directory, or click Browse to install to a different directory.
	Directory Name:
Tivoli, software	/opt/netcool
	B <u>r</u> owse
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

Figure 3-3 Installation destination directory

**Note:** Keep in mind that the installation directory has to be writable or owned by the user in which you want to run IBM Tivoli Netcool/Impact.

- Image: Netcool/Impact InstallShield Wizard

   Image: Netcool/Impact InstallGovernment of the suggested configuration.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Installation type that best suits your needs.

   Image: Netcool/Impact Impact -
- 5. Select the **Typical** radio button, as shown on Figure 3-4. Click **Next**.

Figure 3-4 Installation choices

6. For the WebSphere Application Server settings, complete the ports fields shown in Figure 3-5 on page 53. The default ports for these services are already entered in our example. Click **Next**.

≥	WebSphere Applicatio	on Server CE - InstallShield Wizard
- en	Netcool will install, co	nfigure, and manage its own embedded WASCE Server.
14. 7.	HTTP Port	8080
Aller	HTTPS Port	8443
	Naming Port	1099
Tivoli, software	EJB Port	4201
	COSNaming Port	1050
	JMS Port	61616
	Derby Port	1527
- Tratalishiek		
$\mathbf{\pi}$		< Back Next > Cancel

Figure 3-5 WebSphere Application Server settings

The ports are explained in Table 3-1. You can check the availability of these ports by running the **netstat** command (**netstat** -an | grep 8080) or the **1sof** command.

	Prompt	Description
	HTTP port	The HTTP port used by WebSphere Application Server Community Edition. When you log into IBM Tivoli Netcool/Impact using a Web browser, you use this HTTP port. The default is 8080.
	HTTPS port	The HTTPS port used by WebSphere Application Server Community Edition. IBM Tivoli Netcool/Impact components that are configured to communicate with each other using SSL use this port. The default is 8443.
	Naming port	The RMI naming port used by WebSphere Application Server Community Edition. The default is 1099.
	EJB port	The Enterprise Java Beans port used by WebSphere Application Server Community Edition. The default is 4201.
	COS Naming port	The Common Object Services naming port used by WebSphere Application Server Community Edition. The default is 5050.
	JMS port	The Java Message System port used by WebSphere Application Server Community Edition. The default is 61616.

Table 3-1 WebSphere Application Server options

Prompt	Description
Derby port	The Apache Derby port used by WebSphere Application Server Community Edition. The default is 1527.

7. For the WebSphere Application Server settings, complete the additional ports fields shown in Figure 3-6. The default ports for these services are already entered in our example.

	WebSphere Application S	Server CE - InstallS	hield Wizard	
- end	Netcool will install, config	ure, and manage its	own embedded WAS	CE Server.
un To	DirectoryService Port	1389		
Alle See	JAAS Port	4242	j	
	Tomcat AJP Port	8009		
Tivoli. software				
charallaniek				
*	< E	Back	Next >	Cancel
			_	

Figure 3-6 WebSphere Application Server settings

These ports are explained in Table 3-2.

Table 3-2WebSphere Application Server options

Prompt	Description
Directory service port	The Apache Directory Server port used by WebSphere Application Server Community Edition. The default is 1389.
JAAS port	The Java Authentication and Authorization Service port used by WebSphere Application Server Community Edition. The default is 4242.
Tomcat AJP port	The Tomcat AJP connector port used by WebSphere Application Server Community Edition. The default is 8009.

8. Figure 3-7 on page 55 shows the settings for the Netcool Security Manager Client. The default settings for these settings are already entered for our example.

	Netcool Security Manager Client - InstallShield Wizard
- Com	Configure the Security Manager Location
Le Th	Enter the host and port for your Security Manager
All Contraction	Security Manager Host localhost
10211 122	Security Manager Port 8077
Tivoli. software	
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

Figure 3-7 Security Manager settings

The default settings are explained in Table 3-3.

Table 3-3 Security Manager options

Prompt	Description
Security Manager host	The host name or IP address of the system where the Netcool Security Manager is located. The default is localhost.
Security Manager port	The port used by the Security Manager. The default is 8077.

- 9. For a Windows installation only, configure the Netcool database by performing these steps:
  - a. Provide a user (which will be used to run the PostgreSQL database) for the installer, as shown in Figure 3-8.

🛃 Netcool/Postgres - In	stallShield Wizard			
- Card	Service configuration			
Aust	Account name Account domain	postgres BASEIMAGE		
Tivoli. software	Account password			
	The service account is the acc must NOT be a member of the created an account, the instal nassword	Create User count that runs the Pr e local administrator ler can do so for you.	ostgreSQL database s s group. If you have no Enter an account nan	erver. It It already ne and
AnsialiShiel()	< <u>B</u> ack	<u>N</u> ext =	<u>C</u> an	cel

Figure 3-8 Database user

Table 3-4 provides the details about the defaults.

Table 3-4	Database user options
-----------	-----------------------

Prompt	Description
Account name	The Windows user that controls the database. The default is postgres.
Account domain	The Windows domain for the database user. the default is the name of the computer where you are installing IBM Tivoli Netcool/Impact component.
Account password	The Windows password for the database user. the default is a blank string.

b. Provide a suitable port (which will be used to run the PostgreSQL database) for the installer, as shown in Figure 3-9 on page 57.

**Note:** Provide a password for the PostgreSQL installer. There is no default password filled out, but installation will fail if you leave it blank.

	Configure the Netcool/Database
- end	Netcool will install, configure and manage its own embedded database.
	Port 5435
Tivoli. software	The database will run as the current user you are logged in as now. The database contains its own internal users. Please enter a username and password for this internal database user.
	Username postgres Password •••••••
the tellomies *	< <u>B</u> ack <u>Next &gt; Cancel</u>

Figure 3-9 Netcool database settings

Table 3-5 provides the details about the defaults.

Table 3-5 Netcool database options

Prompt	Description
Netcool database port	The port used by the Netcool database server. The default is 5435.
Netcool database user	The internal admin user for the Netcool database server. This is not an operating system user. The default is postgres.
Netcool database password	The password for the internal database admin user.

c. Name server configuration: Complete the two required fields (shown in Figure 3-10) for the name server that IBM Tivoli Netcool/Impact will use to publish its service.

	Netcool/Impact Server - InstallShield Wizard
- I -	Name Server Configuration
	Impact uses the Netcool Name Server to publish its services. By default the application name server is installed as part of the Netcool Impact Client. If you moved the "nameserver.war" file to a different J2EE server, please enter the correct http host and port below, otherwise use the host and port of your Netcool Impact Server.
Tivoli. software	Name Server Host
	Name Server Port 8080
-thetelliphies	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

Figure 3-10 Name server settings

Table 3-6 describes the default ports for these fields.

Table 3-6	Name server options
-----------	---------------------

Prompt	Description
Name server host name	The host name or IP address of the system where the IBM Tivoli Netcool/Impact name server is running. This is the same system where the IBM Tivoli Netcool/Webtop GUI server is located. The default is localhost.
Name server port	The port where the IBM Tivoli Netcool/Impact name server is running. The default is 8080.

**Note:** If you click **Next** too quickly, the installer will not find the name server, which is normal behavior.

d. Complete the three required fields (shown in Figure 3-11 on page 59) for the Instance Name and Cluster Group that IBM Tivoli Netcool/Impact will use.
$\mathbf{r}$	Netcool/Impact New Server - InstallShield Wizard		
- Com	Configure the Impact Instance		
	Netcool/Impact allows you to run multiple instances of the server with different configuration characteristics. The instance name is a unique string that identifies the server instance. The cluster group is used to identify server instances that form a cluster. If you don't want to run in clustering mode, just assign a unique name to every server instance.		
Tivoli, software	Instance Name NCI		
	Cluster Group NCICLUSTER		
	Command Line Port 2000		
the fall of the text	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel		

Figure 3-11 IBM Tivoli Netcool/Impact settings

Table 3-7 describes the default entries for these fields.

Table 3-7	IBM Tive	oli Netcool/	Impact of	options

Prompt	Description	
Instance name	The name of the new instance. The default is NCI.	
Cluster group	The name of the server cluster for the new instance. If you are not using server clustering, accept the default. Otherwise, enter the name of the cluster that the instance will participate in. The default is NCICLUSTER.	
Command-line port	The port used by the command-line service for the new instance. The default is 2000.	

e. In Figure 3-12, provide the information that IBM Tivoli Netcool/Impact will use to connect to the event source ObjectServer.

**Note:** Make sure your ObjectServer is running, as the installer will check for its availability

·	Netcool/Impact New Server - In	stallShield Wizard		
- Com	Configure the ObjectServer Location			
	Netcool/Impact will need an event source. By default, it will connnect to Netcool/OMNIbus. Other types of event sources require manual configuration. Please enter the host and port for your Netcool/OMNIbus server.			
1935564	ObjectServer Host	localhost		
Tivoli, software	ObjectServer Port	4100		
	ObjectServer User	root		
	ObjectServer Password			
	< <u>B</u> ack	<u>N</u> ext > <u>C</u> ancel		

Figure 3-12 IBM Tivoli Netcool/Impact ObjectServer settings

Table 3-8 lists the default values.

Table 3-8	IBM Tivoli Netcool/Impact ObjectServer options
-----------	------------------------------------------------

Prompt	Description	
ObjectServer host	The host name or IP address for the primary ObjectServer in your environment.	
ObjectServer port	The port used by the ObjectServer.	
ObjectServer user	The name of the user account that IBM Tivoli Netcool/Impact will use to connect as an ObjectServer client. The default is root.	
ObjectServer password	The password for the ObjectServer user.	

f. Configure the e-mail settings by providing the information IBM Tivoli Netcool/Impact needs to connect to an SMTP capable e-mail server, as shown in Figure 3-13 on page 61.

	Netcool/Impact New Server - InstallShield Wizard		
- Com	Configure the Email Information		
44	If you are using Impact's email sender service, please enter a valid smtp host that will accept connections from this machine. On most Unix hosts, it is safe to use "localhost"		
1030145	SMTP Host localhost		
Tivoli. software	Email Sender Impact		
	< Back Next > Cancel		

Figure 3-13 E-mail server settings

Table 3-9 lists the e-mail server details.

Table 3-9 E-mail server settings

Prompt	Description
SMTP host	The host to be used by the E-mail Sender Service. The default is localhost.
E-Mail sender name	The name of the user associated with the E-mail Sender Service. The default is impact.

g. Choose the version control system that you would like to use. The default is Impact CVS, as shown in Figure 3-14.



Figure 3-14 CVS server settings

10. You are now ready to install IBM Tivoli Netcool/Impact. Your installation settings are listed in Figure 3-15. Confirm that your settings are correct and click **Install** to continue.

	Netcool/Impact - InstallShield Wizard
- Com	Please read the summary information below.
Tivoli software	Netcool/Impact will be installed in the following location: /opt/netcool with the following features: GUI Server Impact Server Impact Database for a total size: 415.1 MB
	< <u>B</u> ack <u>I</u> nstall <u>C</u> ancel

Figure 3-15 Installation summary

11. The window shown in Figure 3-16 opens when the installation is complete. Click **Finish**.



Figure 3-16 Successful installation

**Note:** The IBM Tivoli Netcool/Impact installation log file is named log.txt and is located in the \$NCHOME directory. This log file contains runtime messages generated during the installation process. We advise that you look at this file if you have any installation problems.

### 3.2.2 Other installation types

Apart from the graphical installation, you can perform the installation using the following methods:

Console installation: When an X Window System is not available on a UNIX platform, you can install IBM Tivoli Netcool/Impact using a console mode text only interface. Start the console installation by running the setuparch.bin -console command.

- Silent installation: Allows predefined options to be created and entered before the installation process, thus making the installation non-interactive. To perform a silent installation, perform the following steps:
  - a. Delete the InstallShield directory from the installation user's home path.
  - b. Create an option file by running the setuparch.bin -options-record
     <optionsfile> command. The installer runs in GUI mode and you can enter your choices, as shown in 3.2.1, "Graphical installation" on page 50. The resulting file can be used for silent installation.
  - c. Using our options file, run the following command:

```
setup<arch>.bin -options <optionsfile> -silent
```

#### 3.2.3 Installation log file

The IBM Tivoli Netcool/Impact installation log file is named log.txt and is located in the \$NCHOME directory. This log file contains runtime messages generated during the installation process.

You can use this log to verify that you have installed IBM Tivoli Netcool/Impact successfully. You can also use it to troubleshoot installation problems.

#### 3.2.4 JRExec service on Windows

For a Windows based server, the JRExec component has to be installed separately. The IBM Tivoli Netcool/Webtop JRExec server is a component that allows IBM Tivoli Netcool/Impact to run external commands, scripts, and applications from within a policy. On Windows based systems, you must also manually create the JRExec service after installation. To create the JRExec service, enter the following command:

%NCHOME%/impact/bin/nci_jrexec -i nci_jrexec.conf

# 3.3 Cluster installation

A cluster installation allows IBM Tivoli Netcool/Impact servers to fail over, thus providing redundancy for fault tolerance. We discuss this topic in the following sections:

- 3.3.1, "Simple IBM Tivoli Netcool/Webtop cluster installation" on page 65
- 3.3.2, "Complete failover IBM Tivoli Netcool/Impact cluster installation" on page 69

# 3.3.1 Simple IBM Tivoli Netcool/Webtop cluster installation

A simple IBM Tivoli Netcool/Webtop cluster is a redundant IBM Tivoli Netcool/Impact server installation on two machines. This allows failure on an IBM Tivoli Netcool/Impact server process to be resumed by the other instance. This simple cluster does not provide a redundancy for other components, such as a name server, ObjectServer, or security server.

To install a simple IBM Tivoli Netcool/Webtop cluster, perform the following steps;

- 1. Install the first server, as described in 3.2, "Basic installation" on page 50.
- 2. On the other servers, start a custom installation, as shown in Figure 3-17.



Figure 3-17 Installation types

3. Select only the Impact Server options, as shown in Figure 3-18.



Figure 3-18 Cluster installation

4. As the Impact Database is not selected in Figure 3-18, a warning appears in the window that informs that you must manually configure Impact Reporting, as shown in Figure 3-19. This discussion can be found in 7.2, "Performance reporting" on page 163.



Figure 3-19 Database warning

5. Assign the appropriate WebSphere port numbers using the same information that you used while installing the primary server, as shown in Figure 3-20.

	WebSphere Applicatio	on Server CE - I	nstallShield Wizard
- end	Netcool will install, cor	nfigure, and mar	nage its own embedded WASCE Server.
the To	HTTP Port	8080	
Alle See	HTTPS Port	8443	
	Naming Port	1099	
Tivoli. software	EJB Port	4201	
	COSNaming Port	1050	
	JMS Port	61616	
	Derby Port	1527	
	WebSphe	re Application	Server CE - InstallShield Wizard
	Netcool	will install, confi <u>c</u>	gure, and manage its own embedded WASCE Server.
chistaliani	Direct	toryService Port	1389
	JAAS I	Port	4242
	Tomca		8009
Tivoli, so	ftware		

Figure 3-20 WebSphere Application Server settings

6. In the Name Server Configuration window, enter the data of the primary servers name server host name and port, as shown in Figure 3-21.



Figure 3-21 IBM Tivoli Netcool/Impact name server settings

7. In the Security Manager Client window, enter the data for the primary server's Security Manager host name and port, as shown in Figure 3-22.

	Netcool Security Manager Client	- InstallShield Wizard
	Configure the Security Manager Loc Enter the host and port for your Sec Security Manager Host Security Manager Port	ation curity Manager YourPrimaryHost 8077
Tivoli. software		
	< <u>B</u> ack	<u>N</u> ext > <u>C</u> ancel

Figure 3-22 Security Manager settings

8. When installing the second IBM Tivoli Netcool/Impact server, it is important that you provide another instance name for this server, as shown in Figure 3-23.



Figure 3-23 IBM Tivoli Netcool/Impact settings

9. Provide the ObjectServer address and port for the first installation, as shown in Figure 3-24.

	Netcool/Impact New Server - In	stallShield Wizard
Tivoli software	Configure the ObjectServer Location Netcool/Impact will need an event s Netcool/OMNIbus. Other types of e Please enter the host and port for y ObjectServer Host ObjectServer Port ObjectServer User ObjectServer Password	ource. By default, it will connnect to vent sources require manual configuration. our Netcool/OMNIbus server. YourPrimaryHost 4100 root
	< <u>B</u> ack	Next > Cancel

Figure 3-24 IBM Tivoli Netcool/Impact ObjectServer settings

10. Assign the SMTP server and Impact CVS system as appropriate.

11.Complete the installation.

# 3.3.2 Complete failover IBM Tivoli Netcool/Impact cluster installation

The complete failover IBM Tivoli Netcool/Impact cluster creates two almost identical servers that both have the full functionality to be a stand-alone IBM Tivoli Netcool/Impact server.

To install a complete failover IBM Tivoli Netcool/Impact cluster, perform the following steps:

1. Install both servers, as described in 3.2, "Basic installation" on page 50.

**Note:** We suggest using different IBM Tivoli Netcool/Impact instance names on both servers. This configuration assumes that a redundant ObjectServer cluster is available.

- 2. Configure the Security Manager for failover. Both Security Managers must not be running. Perform the following steps:
  - a. Manually create a identical database on both servers. This is only needed if the primary servers security information has changed.

- b. Copy \$NCHOME/security/db/security.script from the primary host to a temporary directory on the secondary host (such as /tmp).
- c. On the secondary server, load the copied security.script file by running the following command:

```
$NCHOME/security/bin/ncsm_db restore -backupfile
/tmp/security.script
```

d. Update the \$NCHOME/security/etc/db.properties file and comment the line for server.address= localhost that it becomes the following:

```
#server.address= localhost
```

- 3. Configure the Security Manager data source by performing these steps:
  - a. Edit \$NCHOME/security/etc/SM_smDataSource.ds on both servers and change the lines for smDataSource.HSQL.BACKUPHOST and smDataSource.HSQL.BACKUPPORT to reflect the other server.
  - b. Edit \$NCHOME/security/etc/SM_smParentType.ds on both servers and add the lines shown in Example 3-1.

```
Example 3-1 Modification to SM_smParentType.ds
```

```
smParentType.SQL.NUMDSPROPERTIES=1
smParentType.SQL.DSPROPERTY.1.NAME=IMPACT_REPLICATE_CHANGES
smParentType.SQL.DSPROPERTY.1.VALUE=true
```

- c. Edit \$NCHOME/security/etc/SM_server.props:
  - On the primary server, add the lines shown in Example 3-2.

```
Example 3-2 Modification to SM_server.props in primary server
```

```
impact.security.failover=true
impact.security.controlport=5995
impact.security.failover.other.host=YourSecondaryServer
impact.security.failover.other.port=5996
impact.security.failover.ResyncRateInSec=10
```

• On the secondary server, add the lines shown in Example 3-3.

```
Example 3-3 Modification to SM_server.props.in secondary server
```

```
impact.security.failover=true
impact.security.controlport=5996
impact.security.failover.other.host=YourPrimaryServer
impact.security.failover.other.port=5995
impact.security.failover.ResyncRateInSec=10
```

YourSecondaryServer and YourPrimaryServer should be replaced with the host names of your machines.

**Note:** The ports used in this installation (5995 and 5996) have been chosen from any of the free ports on the servers.

- 4. Modify the IBM Tivoli Netcool/Impact GUI servers to use the failover security manager:
  - a. Edit \$NCHOME/guiserver/etc/server.props on the primary server, as shown in Example 3-4.

Example 3-4 Modification to server.props in primary server

```
security.backup.host.1=YourSecondaryServer
security.backup.port.1=8077
impact.security.backup.host.1=YourSecondaryServer
impact.security.backup.port.1=8077
```

b. Edit \$NCHOME/guiserver/etc/server.props on the secondary server, as shown in Example 3-5.

Example 3-5 Modification to server.props in secondary server

```
security.backup.host.1=YourPrimaryServer
security.backup.port.1=8077
impact.security.backup.host.1=YourPrimaryServer
impact.security.backup.port.1=8077
```

- Modify IBM Tivoli Netcool/Impact servers to use the failover Security Manager:
  - a. Edit \$NCHOME/impact/etc/NCI_server.props on the primary server, as shown in Example 3-6.

Example 3-6 Modification to NCI_server.props on primary server

security.backup.host.1=YourSecondaryServer
security.backup.port.1=8077
impact.security.backup.host.1=YourSecondaryServer
impact.security.backup.port.1=8077

b. Edit \$NCHOME/impact/etc/NCI2_server.props on the secondary server, as shown in Example 3-7.

Example 3-7 Modification to NCI_server.props on primary server

```
security.backup.host.1=YourPrimaryServer
security.backup.port.1=8077
impact.security.backup.host.1=YourPrimaryServer
impact.security.backup.port.1=8077
```

**Note:** NCI and NCI2 point to the IBM Tivoli Netcool/Impact instance created on both servers, and should be replaced with your instance names.

- 6. Configure the IBM Tivoli Netcool/Impact Name server for failover by performing these steps:
  - a. Edit \$NCHOME/guiserver/etc/nameserver.props and \$NCHOME/impact/etc/nameserver.props on the primary name server, as shown in Example 3-8.

Example 3-8 Name server modification for the primary server

```
nameserver.1.host=YourSecondaryServer
nameserver.1.port=8080
nameserver.1.location=/nameserver/serviceS
nameserver.count=2
```

 b. Edit \$NCHOME/guiserver/etc/nameserver.props and \$NCHOME/impact/etc/nameserver.props on the secondary name server, as shown in Example 3-9.

```
Example 3-9 Name server modification for secondary server
```

```
nameserver.1.host=YourPrimaryServer
nameserver.1.port=8080
nameserver.1.location=/nameserver/services
nameserver.count=2
```

7. Reconfigure

\$NCHOME/guiserver/install/stage/nameserver/WEB-INF/web.xml on both the primary and secondary name servers. The XML file contains a set of context parameters; the ones that are shown in Table 3-10 on page 73 must be customized.

Table 3-10 Web.xml changes

Context parameter	Value			
REPLICANT.COUNT	2 (number of IBM Tivoli Netcool/Impact servers, which is required for a failover scenario).			
REPLICANT.0.HOST REPLICANT.1.HOST	This contains the list of servers in the cluster. The values should be YourPrimaryServer and YourSecondaryServer.			
REPLICANT.0.PORT REPLICANT.1.PORT	This contains the list of ports for each cluster members. We use 8080 for all members.			
SELFINDEX	1.			

- 8. The failover cluster is now ready. To start the cluster, perform these steps:
  - a. Make sure your failover ObjectServers are running.
  - b. Start the primary Security Manager.
  - c. Start the secondary Security Manager.
  - d. Start the primary IBM Tivoli Netcool/Impact server and wait until it is running.
  - e. Start the secondary IBM Tivoli Netcool/Impact server.

# 3.4 Self-monitoring

IBM Tivoli Netcool/Impact contains a self-monitoring tool to ensure the operation of the IBM Tivoli Netcool/Impact engine. The self-monitoring tool monitors queue status, data source status, cluster status, and memory status from IBM Tivoli Netcool/Impact services and reports its findings to the IBM Tivoli Netcool/OMNIbus ObjectServer. We discuss this tool in the following sections:

- 3.4.1, "Configuring the self-monitoring tool using the GUI" on page 74
- ► 3.4.2, "Configuring self-monitoring using the CLI" on page 75

# 3.4.1 Configuring the self-monitoring tool using the GUI

To configure self monitoring using the Web-based interface, perform the following steps:

 Open a Web browser and enter the URL of your primary server (http://YourPrimaryServer:8080/nci). The window shown in Figure 3-25 opens.

NETCOOL [®] Suite ^{**}		logged in as:admin   NCI:NCICLUSTER (9.42.171.40:43457:NCI	I) • Logout
Projects	Global		N N
Projects: Default	🖉 🗱 🗗 📲		
Data Sources And Types	i V		
Operator Views	= 9		
O Policies	<b>T</b>		
Reports			
Services			10
🐼 Wizards			2
Service Status 🗹 Auto Refresh		IBM. <b>NETCOOL</b> [®] /Impact [™]	Java
DefaultEmailReader			COMPATIBLE
🔴 DefaultEventReader			
🔴 DefaultJabberReader			
DefaultPolicyActivator			
🔴 EmailSender			
EventProcessor			
HibernatingPolicyActivator		IBM Netcool®/Impact™ v4	
🔴 JMSMessageListener		Netcool and IBM are registered trademarks or International Business Machines Corporation	
JabberService			
PolicyLogger			
PrecisionEventListener			
SelfMonitoring			
			_

Figure 3-25 Service status

2. Click the **SelfMonitoring** entry in the Service Status window in the bottom left pane, which will open a new window, shown in Figure 3-26 on page 75, in which you can turn on self-monitoring.

🕗 http://9.42.170.179:8080 - SelfMonitoring Configuration - Microsoft In 🖃 🔲 🔀	
SelfMonitoring Service:	
ObjectServer Data Source: defaultobjectserver 💌	
Memory Status 🗹 Enable	
Interval: 60 📮	
✓ Deduplication	
Queue Status <b>∨</b> Enable	
Interval: 60	
✓ Deduplication	
Cluster Status: 🕑 Enable	
Data Source Status: 🔽 Enable	
Startup: 🗹 Automatically when server starts	
Service Log: Write to file	
? DK Cancel	
🕘 🔹 İnternet 🛒	

Figure 3-26 Self-monitoring configuration

- 3. Turn on the service by clicking the Start ( 🗈 ) button for the SelfMonitoring service.
- 4. You can click the view log ( 🗉 ) button for the SelfMonitoring service to see its status.

### 3.4.2 Configuring self-monitoring using the CLI

The server configuration can be modified using a command-line interface (CLI) through a telnet connection to YourPrimaryServer on port 2000. The interface allows you to issue SQL statements that checks and updates the IBM Tivoli Netcool/Impact configuration. Here are some of those commands:

- Generic self monitoring commands:
  - To start the Self-Monitoring Service, run the following command:
    - UPDATE Service SET Running = true WHERE Name = 'SelfMonitoring';
  - To stop the Self-Monitoring Service, run the following command:

UPDATE Service SET Running = false WHERE Name = 'SelfMonitoring';

- Memory status monitoring commands:
  - To check if memory status monitoring is enabled, run the following command:

SELECT IsMemoryStatusEnabled FROM Service WHERE Name =
'SelfMonitoring';

This will return a value of true if memory status monitoring is enabled.

– To enable memory status monitoring, run the following command:

UPDATE Service SET EnableMemoryStatus = true where Name='SelfMonitoring';

- To disable memory status monitoring, run the following command:

UPDATE Service SET EnableMemoryStatus = false where Name='SelfMonitoring';

To check if memory status monitoring is enabled, run the following command:

SELECT MemoryStatus FROM Service WHERE Name='SelfMonitoring';

To check the memory status monitoring history, run the following command:

SELECT MemoryStatusHistory FROM Service WHERE Name =
'SelfMonitoring';

- To view the total JVM[™] heap size, run the following command:

SELECT TotalVMHeapSize FROM Service WHERE Name =
'SelfMonitoring';

- To view the maximum JVM heap size, run the following command:

SELECT MaxVMHeapSize FROM Service WHERE Name = 'SelfMonitoring';

– To view the free system memory, run the following command:

SELECT FreeSystemMemory FROM Service WHERE Name =
'SelfMonitoring';

To check if memory status events deduplication is enabled, run the following command:

SELECT MemoryDeduplication FROM Service WHERE Name =
'SelfMonitoring';

This returns a value of true if memory status event deduplication is enabled.

- To disable memory status event deduplication, run the following command:

UPDATE Service SET MemoryDeduplication = false WHERE Name= 'SelfMonitoring';  To view the intervals at which IBM Tivoli Netcool/Impact checks the memory status, run the following command:

SELECT MemoryInterval FROM Service WHERE Name = 'SelfMonitoring';

 To change the interval at which IBM Tivoli Netcool/Impact server checks the memory status, run the following command:

```
UPDATE Service SET MemoryInterval = interval WHERE Name =
'Self-Monitoring';
```

The interval is the number of seconds at which the server checks the queue status.

- Queue status monitoring commands:
  - To check if queue status monitoring is enabled, run the following command:

SELECT IsQueueStatusEnabled FROM Service WHERE Name =
'Self-Monitoring';

This returns a value of true if queue status monitoring is enabled.

- To enable queue status monitoring, run the following command:

UPDATE Service SET EnableQueueStatus = true WHERE Name = 'Self-Monitoring';

- To disable queue status monitoring, run the following command:

UPDATE Service SET EnableQueueStatus = false WHERE Name =
'Self-Monitoring';

- To view the current queue status, run the following command:
  - SELECT QueueStatus FROM Service WHERE Name = 'Self-Monitoring';
- To view the queue status history, run the following command:

SELECT QueueStatusHistory FROM Service WHERE Name =
'Self-Monitoring';

To check if queue status event deduplication is enabled, run the following command:

SELECT QueueDeduplication FROM Service WHERE Name =
'Self-Monitoring';

This returns a value of true if queue status event deduplication is enabled.

- To enable queue status event deduplication, run the following command:

UPDATE Service SET QueueDeduplication = true WHERE Name =
'Self-Monitoring';

- To disable queue status event deduplication, run the following command:

```
UPDATE Service SET QueueDeduplication = false WHERE Name =
'Self-Monitoring';
```

 To view the interval at which the IBM Tivoli Netcool/Impact server checks the queue status, run the following command:

SELECT QueueInterval FROM Service WHERE Name = 'Self-Monitoring';

 To change the interval at which the IBM Tivoli Netcool/Impact server checks the queue status, run the following command:

```
UPDATE Service SET QueueInterval = interval WHERE Name =
'Self-Monitoring';
```

The interval is the number of seconds at which the server checks the queue status,

- Data source status monitoring:
  - To check if data source status monitoring is enabled, run the following command:

SELECT IsDataSourceStatusEnabled FROM Service WHERE Name =
'Self-Monitoring';

This returns a value of true if data source status monitoring is enabled.

- To enable data source status monitoring, run the following command:

UPDATE Service SET EnableDataSourceStatus = true WHERE Name =
'Self-Monitoring';

- To disable data source status monitoring, run the following command:

UPDATE Service SET EnableDataSourceStatus = false WHERE Name =
'Self-Monitoring';

- To view the current data source status, run the following command:

SELECT DataSourceStaths FROM Service WHERE Name =
'Self-Monitoring';

- To view the data source status history, run the following command:

SELECT DataSourceStatusHistory FROM Service WHERE Name =
'Self-Monitoring';

- Cluster status monitoring commands:
  - To check if cluster status monitoring is enabled, run the following command:

SELECT IsClusterStatusEnabled FROM Service WHERE Name =
'Self-Monitoring';

This returns a value of true if cluster status monitoring is enabled.

- To enable cluster status monitoring, run the following command:

UPDATE Service SET EnableClusterStatus = true WHERE Name =
'Self-Monitoring';

- To disable cluster status monitoring, run the following command:

UPDATE Service SET EnableClusterStatus = false WHERE Name =
'Self-Monitoring';

- To view the current cluster status, run the following command:

SELECT ClusterStatus FROM Service WHERE Name = 'Self-Monitoring';

- To view the cluster status history, run the following command:

```
SELECT ClusterStatusHistory FROM Service WHERE Name =
'Self-Monitoring';
```

# 3.5 Startup configuration

Services can be controlled and set from the IBM Tivoli Netcool/Impact Web interface. Select the services that are needed for using IBM Tivoli Netcool/Impact from the Service Status window, as shown in Figure 3-27.

Service Status 🗹 Auto Refresh	
🔴 DefaultEmailReader	
DefaultEventReader	
🔴 DefaultJabberReader	
DefaultPolicyActivator	
🔴 EmailSender	
EventProcessor	
HibernatingPolicyActivator	
JMSMessageListener	
JabberService	
PolicyLogger	
PrecisionEventListener	
🔵 SelfMonitoring	
	•

Figure 3-27 Service Status

To turn on the service, click the **b** button. If you want to have this service autostart at server startup, click the service name and a window will appear. In this window, select **Startup Automatically when server starts**. The services that can be started automatically are:

- E-mail Reader
- Event Reader
- Jabber Reader
- Policy Activator
- Hibernating Policy Activator
- ► JMS Message Listener
- ► Jabber Service
- Event Listener
- Self Monitoring

There are other startup methods for various IBM Tivoli Netcool/Impact components:

Starting the GUI server:

On UNIX: Run the **\$NCHOME/bin/wasce** start command.

On Windows: Use the services applet from the Control Panel. Start the service called WebSphere Application Server CE.

**Note:** If the IBM Tivoli Netcool/Impact server is deployed on the same server, this also launches the IBM Tivoli Netcool/Impact instance, and the name server, as the PostgreSQL database.

If the IBM Tivoli Netcool/Impact instance is running on a separate machine (a distributed installation or a secondary server), launch the instance as follows:

On UNIX: Run the **\$NCHOME/impact/bin/nci_server INSTANCENAME** command.

On Windows: Use the services applet from the Control Panel. Start the service called WebSphere Application Server CE.

Import the reporting schema. Run the following commands:

\$NCHOME/bin/nc_db_postgres setupinit \$NCHOME/impact/bin/nci_db setupinit

Start the IBM Tivoli Netcool/Impact database server, if this is located on a separate server (this is required only for the GUI reporting feature).

On UNIX: Run the **\$NCHOME/bin/nc_db_postgres** start command.

• Start the JRExec server as follows:

On UNIX: Run the **\$NCHOME/impact/bin/nci_jrexec** command.

On Windows: Use the services applet from the Control Panel. Start the service called Netcool JRExec Server.

► Starting all IBM Tivoli Netcool/Impact services on bootup:

On Windows: Change the startup type of the services to Automatic by using the services applet from the Control Panel.

On UNIX: Copy \$NCHOME/impact/install/startupscripts/impact.rc to the UNIX startup directory, which is usually located in /etc/init.d/. Link this script to the necessary runlevels, and make it executable by the root user.

When IBM Tivoli Netcool/Impact is installed on the same machine as IBM Tivoli Netcool/OMNIbus, you can use IBM Tivoli Netcool/OMNIbus' process control to start and stop the IBM Tivoli Netcool/Impact server.

IBM Tivoli Netcool/Impact can be stopped by running the nci_shutdown script under \$NCHOME/impact/bin. It gracefully shuts down the PostgreSQL database and the IBM Tivoli Netcool/Impact server.



# 4

# Configuration

This chapter describes configuration items that you can perform for IBM Tivoli Netcool/Impact. We discuss this topic in the following sections:

- ▶ 4.1, "Data sources" on page 84
- 4.2, "Data types" on page 90
- ▶ 4.3, "Policies" on page 95
- ► 4.4, "Event Reader" on page 98
- ► 4.5, "Event Processor" on page 104
- ► 4.6, "Logging configuration" on page 106
- ▶ 4.7, "Operator View configuration" on page 109

# 4.1 Data sources

This section discusses IBM Tivoli Netcool/Impact data sources. The topics are discussed in the following sections:

- 4.1.1, "Data source definition" on page 84
- 4.1.2, "Configuring a new data source" on page 86

#### 4.1.1 Data source definition

A data source is an element of an IBM Tivoli Netcool/Impact data model that represents real-world information. IBM Tivoli Netcool/Impact provides the following categories of data sources:

SQL database data sources

SQL database data sources represent relational databases. IBM Tivoli Netcool/Impact supports most of the popular commercial relational databases, such as Oracle, Sybase, and Microsoft SQL Server®. In addition, it also supports freely-available databases like MySQL[™] and PostgreSQL. The IBM Tivoli Netcool/OMNIbus ObjectServer is also supported as an SQL data source.

LDAP data sources

LDAP data sources represent read-only data on LDAP directory servers. IBM Tivoli Netcool/Impact supports Netscape, iPlanet, and OpenLDAP servers.

Mediator data sources

Mediator data sources represent third-party applications that are integrated with IBM Tivoli Netcool/Impact through the DSA Mediator. These data sources include a wide variety of network inventory, network provisioning, and messaging system software. In addition, providers of XML and SNMP data can also be used as mediator data sources. The SNMP DSA adapter allows you to send SNMP traps and retrieve SNMP variables from a device. The XML DSA allows you to interpret XML format data for use in your policies. Other mediator data sources include Web Services, JMS, Precision (IBM Tivoli Network Manager), and socket.

Internal data repository

The internal data repository is a built-in data source for IBM Tivoli Netcool/Impact. The primary responsibility of the internal data repository is to store system data. However, you can also use it to store the data in predefined and user-defined internal data types. Data sources are accessed using an adapter. The adapter is commonly known as the Data Source Adaptor (DSA). For a complete list of supported data sources, refer to the *IBM Tivoli Netcool/Impact DSA Reference Guide*, SC23-8856, which can be found at the following address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.ne
tcool_impact.doc/im402dsa.pdf

Additional information about data sources can be found in Chapter 4, "Working with Data Sources", in the *IBM Tivoli Netcool/Impact Version 4.0 Solution Guide*, SC23-8857, which can be found at the following address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.ne
tcool_impact.doc/im402sg.pdf

## 4.1.2 Configuring a new data source

To configure a new data source, perform these steps:

1. Log in (using the Web GUI) to your IBM Tivoli Netcool/Impact GUI server on port 8080 by using the URL http://yourGuiServer:8080/nci/, as shown in Figure 4-1.

Image: Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second	✓ ↔ × Google ✓ → × Google ✓ → × Page × ⊙ Tgols × ×
Elle     Edit     View     Favorites     Iools     Help	M + N → H + Page + () Tgols + *
😪 🎄 🌈 Micromuse Netcool	Tools +
Technical Surrent 1, Sector 19	
Please enter your Netcool username and password Username: admin Password: Log In	
IBM NETCOOL® Solutions [™] IBM is a trademark of International Business Machines Corporation	IBM.

Figure 4-1 Login page

 Log in with credentials that have the necessary rights to manipulate the data sources and data types. The default administrator is user admin, with a password of netcool. Figure 4-2, the default IBM Tivoli Netcool/Impact page, then appears. The GUI is an application development tool for defining IBM Tivoli Netcool/Impact processing.



Figure 4-2 IBM Tivoli Netcool/Impact GUI

3. Create a new data source at either the Project or Global tab. The data source connects to the database you want to use. Select your database from the drop-down menu, as shown in Figure 4-3.

	Projects	G	ilobal			
Clear All Source Control File Locking:					•	
🖉 Data	Sources					
Source:	Choose One	~		÷		
D Precis	Choose One CorbaMediator			×		
Report	DB2 DirectMediator		7	×		
B Socke	Flat File Informix	Flat File Informix				
🖉 XmlDs	LDAP MS-SOLServer			×		
🖉 defaul	MySQL		<del>7</del>	×		
🔊 test	ODBC Oracle		7	×		
🕅 Data	PostgreSQL SNMPDirectMed	iator				
🕲 Opera	Sybase Tor views					
O Polici	es			-		
Repo	rts				_	
				1	•	

Figure 4-3 New data source database type

- 4. To define the database properties, click the + sign next to it. This opens up a window that is relevant to the selected database. Figure 4-4 on page 89 is an example of a DB2 data source. The required information is:
  - The host name and port number
  - The user and password to access the database

Specifically for the DB2 data source, you must also specify the DB2 database name.

🛃 http://9.42.170.179:8081 - New '	'DB2' Data Source - Microsoft I 🖃 🗖 🔀	
General Settings:	* required field	
Data Source Name:*		
Username:*		
Password:*		
Maximum SQL Connection:*	5	
Database Failure Policy:		
	● Fail over	
	◯ Fail back	
	O Disable Backup	
	Use JDBC Type 4 Driver	
Primary Source:		
Host Name:*	localhost	
Port:*	6789 单	
Database:*	database	
	Test Connection	
Backup Source:		
Host Name:	localhost	
Port:	6789	
Database:	database	
	Test Connection	
?	OK Cancel	
6 Done	🔮 Internet	

Figure 4-4 DB2 data source settings

Other types of databases have their own required fields, such as:

- An Oracle data source needs the Oracle SID.
- Microsoft SQL Server and MySQL data sources need a database name.
- 5. The data source settings define a unique data source name, access credentials, and connection methods to be used. It also provides the failover configuration of the data source, if applicable.
- 6. Test both connections by clicking the **Test Connection** button.
- 7. Click **OK** to add this data source to the IBM Tivoli Netcool/Impact data sources list.

# 4.2 Data types

This section discusses data types. We discuss these topics in the following sections:

- ▶ 4.2.1, "Data type definition" on page 90
- ► 4.2.2, "Configuring a new data type" on page 92

#### 4.2.1 Data type definition

Data types are elements of the data model that represent sets of data stored in a data source. The structure of this data type depends on the category of the data source where it is stored. For example:

- ► For an SQL database, each data type corresponds to a database table.
- For an LDAP server, each data type corresponds to a type of node in the LDAP hierarchy.

A data type definition contains the following information:

- The name of the underlying table or other structural element in the data source
- A list of fields that represent columns in the underlying table or another structural element (for example, a type of attribute in an LDAP node)
- Settings that define how IBM Tivoli Netcool/Impact caches data in the data type

IBM Tivoli Netcool/Impact supports a data type for each of the data sources that are available. Here are the categories of data types:

SQL database data types

SQL database data types represent data stored in a database table. Each data item in an SQL database data type corresponds to a row in the table and each field in the data item corresponds to a column. An SQL database data type can include all of the columns in a table or just a subset of the columns.

LDAP data types

LDAP data types represent data stored at a certain base context level of an LDAP hierarchy. Each data item in an LDAP data type corresponds to an LDAP node that exists at that level and each field corresponds to an LDAP attribute. LDAP data types are read-only. This means that you cannot add, update, or delete data items in an LDAP data type.

Mediator data types

Mediator data types represent data that is managed by third-party applications, such as a network inventory manager or a messaging service. Typically, mediator data types do not represent data stored in database tables. Instead, they represent collections of data that are stored and provided by the data source in various other formats, such as sets of data objects, or as messages.

Internal data types

IBM Tivoli Netcool/Impact provides the following categories of internal data types:

- System data types

System data types are used to stored and manage data used internally by IBM Tivoli Netcool/Impact.These types include Policy, Service, and Hibernation. In most cases, you do not directly access the data in these data types. However, there are some occasions in which you can use them in a policy. Some examples are when you start a policy from within another policy or work with hibernating policies.

- Predefined internal data types

IBM Tivoli Netcool/Impact provides the following predefined internal data types:

- Schedule
- TimeRangeGroup
- Doc

You use the Schedule and TimeRangeGroup data types to manage IBM Tivoli Netcool/Impact scheduling. You can use the Doc data type to store information about URLs located on your intranet.

User-defined internal data types

User-defined internal data types are internal data types that you create. The data items in these data types are stored in IBM Tivoli Netcool/Impact internal data repository, rather than in an external data source. User-defined data types function in much the same way as SQL database data types. IBM recommends that you use internal data types solely for testing and demonstrating IBM Tivoli Netcool/Impact, or for very low-load tasks. User-defined internal data types are slower than external SQL database data types

# 4.2.2 Configuring a new data type

To define a new data type, perform these steps:

1. Create a new data type by selecting the **Data Types** tab in your Project or Global tab, as shown in Figure 4-5.

Projec	ts	Globa		
		٠		
🖉 Data Sour	ces		=	
🕅 Data Type	s			
Data Source:	Choose On	e 🔽	*	
Doc	Choose On Internal	e		
FTEST_Java	ReportsPGS	SQL	×	
STEST_Jav.	Schedule SocketMedi	atorDataSouro	×	
FTEST_Jav	URL XmlDsaMed			
FTEST_Jav	defaultobje test	ctserver	×	
FTEST_JavionBreak				
FTEST_Java	XML_Title		×	
FTEST_Java	XML_Topic	F 🕅	×	
FailedEvent		Eŧ		
🖻 Filter		e		
O HTEST_Java	XML_Book		×	•
Figure 4-5	New dat	ta type		

2. The drop-down menu shows all the available data source that have been created. Select your data source and click the + sign. This opens a window showing the necessary information to configure this data type. Figure 4-6 shows a new data type based upon the default ObjectServer data source. Give the data type a unique name and choose an icon.

🔞 defaultobjectserver 🗙	
Table Description Dynamic Links Cache Settings	
* required field	
General Settings:	
Data Type Name: *	
Data Source Name: defaultobjectserver 💌	
Display Icon: 🏾 🎆 Browse	
State: 🔽 Enabled	
Table Description:	
Base Table: Refresh	
New Field:	
Select: ID Field Format Display Description Key Move Edit	
Data Filter and Ordering:	
Filter:	
Order By:	

Figure 4-6 New data type

3. You must enter the table name in your data source that you would like to use. If you entered an existing table, click **Refresh**. This will populate the table information, as shown in Figure 4-7.

🔞 defaultobj	jectserve	er 🗙						
8								
Table Descrip	otion Dy	namic Links	Cache Setting	s				
Ger	neral Se	attings:						* required
			Data Typ	e Name: *				
			Data Sourc	e Name:	defaultobiectserve	r 🗸		
			Displ	av Icon:	Browse			
			Dispi	Chatas E				
Tal	hle Dec	cription		State: [	Enabled			
10	bie bes	cription.						
			Bas	e Table: a	alerts.status	Refresh		
			Ne	ew Field:	*			
	Select: (all)	ID		Field Name	Format	Display Name	Description	Key Field Move Edit
		Identifier	Iden	tifier	LONG_STRING	Identifier	Identifier	- 🗉 🥒
		Serial	Seria	al	INTEGER	Serial	Serial	🗆 🕂 🌽
		Node	Node		LONG_STRING	Node	Node	ti 🥒
		NodeAlias	Node	Alias	LONG_STRING	NodeAlias	NodeAlias	🗌 ti 🥒
		Manager	Mana	ager	LONG_STRING	Manager	Manager	🗌 ti 🥒
		Agent	Ager	nt	LONG_STRING	Agent	Agent	🗌 ti 🥒
		AlertGroup	Alert	Group	LONG_STRING	AlertGroup	AlertGroup	ti 🥒
		AlertKey	Alert	:Key	LONG_STRING	AlertKey	AlertKey	ti 🎤
		Severity	Seve	erity	INTEGER	Severity	Severity	ti 🎤
		Summary	Sum	mary	LONG_STRING	Summary	Summary	🗌 tų 🥒
		StateChang	ge State	eChange	INTEGER	StateChange	StateChange	ti 🥒
		FirstOccurr	ence First	Occurrence	INTEGER	FirstOccurrence	FirstOccurrence	ti 🥒
		LastOccurre	ence Last	Occurrence	INTEGER	LastOccurrence	LastOccurrence	ti 🎤
		InternalLas	t Inter	rnalLast	INTEGER	InternalLast	InternalLast	ti 🥒
		Poll	Poll		INTEGER	Poll	Poll	ti 🥒
		Туре	Туре	e	INTEGER	Туре	Туре	🗆 TU 🥒

Figure 4-7 ObjectServer data type configuration

- 4. Select at least one field from the database, and remove the fields that you do not use. Selecting only the field that you used speeds up your processing.
- 5. Select the appropriate field for the Display Name.
- 6. Apply a filter or a sorting order
- 7. Save the data type by clicking the floppy icon at the top left.
- 8. Once you have successfully created a new data type, you can use the "Data item for..." view next to the new item in the data types listing.
When you have a table that has been modified by a database administrator, you do not need to recreate the data type; you only need to refresh the data type definition.

# 4.3 Policies

This section discusses IBM Tivoli Netcool/Impact policies. The discussion is covered in the following sections:

- ► 4.3.1, "Policy definition" on page 95
- 4.3.2, "Policy configuration" on page 96

#### 4.3.1 Policy definition

A policy is a set of operations that you want IBM Tivoli Netcool/Impact to perform. These operations are specified using a programming language called IBM Tivoli Netcool/Impact policy language (IPL).

IBM Tivoli Netcool/Impact policy language (IPL) is a scripting language similar to programming languages like C/C++ and Java. It provides a set of data types, built-in variables, control structures, and functions that you can use to perform a wide variety of event management tasks. It also allows you to create your own variables and functions, just as in other programming languages.

The policy language gives you the means to perform the following core operations:

- Handle events; You can use the policy language to handle incoming events, send new events, update existing events, and delete events in an event source.
- Handle data: You can use the policy language to retrieve data from a data source. You can also use the policy language to add new data, update existing data and delete data.
- Send e-mail: You can use the policy language to send e-mail to administrators and users through an SMTP server.
- Execute commands: You can use the policy language to execute external commands, scripts, and applications on local and remote systems.
- Perform other tasks: You can use the policy language to perform many other tasks, such as calling database stored procedures and functions, and manipulating strings and arrays.

For more detailed information regarding IBM Tivoli Netcool/Impact policies and their capabilities, refer to Chapter 5, "Development" on page 113. Other references for policy can be found in:

 IBM Tivoli Netcool/Impact Solutions Guide, SC23-8857, found at the following address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm .netcool_impact.doc/im402sg.pdf

 IBM Tivoli Netcool/Impact Policy Reference Guide, SC23-8855, found at the following address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm .netcool_impact.doc/im402pr.pdf

#### 4.3.2 Policy configuration

The following steps describes our policy configuration:

1. The complete list of available policies can be retrieved by way of the GUI by clicking the **Policies** tab, as shown in Figure 4-8.



Figure 4-8 Policy frame

2. When logged onto the Web GUI. you can see the cluster member to which you are connected. Figure 4-9 shows this instance connection setting to NCI2, which is the backup cluster.



Figure 4-9 IBM Tivoli Netcool/Impact instance

3. IBM Tivoli Netcool/Impact provides a command-line tool named nci_trigger to start a policy from a command line or from another policy. This tool is located in \$NCHOME/impact/bin.

The syntax for invoking this tool is:

\$NCHOME/impact/bin/nci_trigger [-version] | server_name [ user_id |user_id/password |-e/user_id/encrypted_password | NULL ] policy_name field value field value

The options for running the nci_trigger command are listed in Table 4-1. All fields passed through the nci_trigger commands will be available in the policy as @<field> variables. Without the @ sign, the parameter passed would be seen as NULL. You can also hard code the parameters in the body of the policy or use the **Set Runtime Parameters** button from the IBM Tivoli Netcool/Impact GUI before clicking the **Trigger Policy** button.

If you want to run a policy that contains a call to the ReturnEvent function, you must include Identifier and Serial fields in the event container passed to the policy.

Arguments	Description	
-version	Causes <b>nci_trigger</b> to print the IBM Tivoli Netcool/Impact version number, platform, and command syntax to standard output and then exit.	
server_name	This is the instance of an IBM Tivoli Netcool/Impact server where you want the policy to run.	
user_id/password	UNIX only. The user name and password of a valid Netcool Security Manager user who has access to IBM Tivoli Netcool/Impact.	
user_id password	Windows only. the user name and password of a valid Netcool Security Manager user who has access to IBM Tivoli Netcool/Impact.	

Arguments	Description	
-e/user_id/encrypted_ password	UNIX only. The user name and encrypted password of a valid Netcool Security Manager user who has access to IBM Tivoli Netcool/Impact. You can encrypt passwords using the <b>nci_crypt</b> command.	
-e user_id encrypted_password	Windows only. The user name and encrypted password of a valid Netcool Security Manager user who has access to IBM Tivoli Netcool/Impact. You can encrypt passwords using the <b>nci_crypt</b> command.	
policy_name	Name of the policy to run.	
field	Name of a field in the event container passed to the policy. Optional.	
value	Value of a field in the event container passed to the policy. Optional	

# 4.4 Event Reader

This section discusses the Event Reader. The discussion is covered in the following sections:

- 4.4.1, "Event Reader definition" on page 98
- ► 4.4.2, "Event Reader configuration" on page 100

#### 4.4.1 Event Reader definition

An Event Reader is a IBM Tivoli Netcool/Impact service that reads events from an ObjectServer event source. An Event Reader works by querying the ObjectServer at intervals. During each query, the Event Reader retrieves those new or updated events that match its filter conditions. The Event Reader then places the retrieved events in an event queue, where they can be processed by the Event Processor Service. An Event Reader cannot be triggered by a deletion in ObjectServer.

You must have an Event Reader for each instance of the ObjectServer that you want to monitor. Most uses of IBM Tivoli Netcool/Impact require only one Event Reader. It is very uncommon to use IBM Tivoli Netcool/Impact monitor more than one ObjectServer

If the Event Reader is configured to retrieve all unprocessed events at startup, it queries the ObjectServer during this phase for new or updated events that have occurred since the last query. If the Event Reader is not configured to retrieve all unprocessed events, it proceeds to the event polling phase without querying the ObjectServer.

During the event polling phase, the Event Reader queries the ObjectServer at intervals for all new or unprocessed events. You set the polling interval when you configure the Event Reader.

It is important to understand that the Event Reader processes the events one by one, so each event gets to be processed against the relevant policy. If you need to process multiple events at the same time, it is necessary to use separate ObjectServer data sources.

The Event Reader provides a feature called event locking that allows a multi-threaded Event Processor to categorize incoming alerts based on the values of specified alert fields and then to process them within a category one at a time. With event locking enabled, if more than one event exists with a certain lock value, then these events are not processed at the same time. These events are processed in a specific order in the queue.

#### 4.4.2 Event Reader configuration

You can configure an event read by performing the following steps:

1. To create a new Event Reader, click the **Services** tab in the Web GUI, select the Event Reader type, and click the + sign. The window that opens is shown in Figure 4-10.

http://9.42.170.179:8081 - EventReader Configuration - Microsoft Internet Explo	rer 💶 🖂
General Settings Event Mapping	
General Settings:	* required field
Service Name:*	
ObjectServer Data Source: defaultobjectserver 💌	
Polling Interval:* 3000 🖕 (milliseconds)	
Restrict Fields: Fields	
Startup: Automatically when server starts	
Service Log: 🗌 Write to file	
Collect Reports: Enable	
Clear State: Clear	
?	OK Cancel
Done, but with errors on page.	Internet 🚲

Figure 4-10 Event Reader configuration

In the window shown in Figure 4-10, perform the following steps:

- a. Supply a unique name.
- b. Select the ObjectServer source that you want to use. The list of ObjectServer information comes from the data sources definition.
- c. Select the polling interval (this is in milliseconds).

Note: The default polling interval is 3 seconds.

2. Select which fields you want to use in this reader. Figure 4-11 shows the default selection, which uses all the fields.

Line 2014 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 12:00 10	tServicer fields for EventReader - Microsoft In 🖃 🗖 🔀
Alert Status Fields	
Available	Selected Add > Acknowledged
	Agent     AlertGroup     AlertKey     Class     Customer     EventId     ExpireTime     ExtendedAttr     ExtendedAttr
Optimize List	
?	OK Cancel
6 Done	🔮 Internet 🧮

Figure 4-11 Available fields

3. Configure the event mapping using the Event Mapping tab shown in Figure 4-10 on page 100. The event mapping configuration is shown in Figure 4-12.

街 http://9.42.170.179:8081	- EventReader Configuration - Microsoft Internet Explorer	
General Settings Event Mapping		
Event Mapp	ping:	
Event Matching: O Test events with all filters		
	<ul> <li>Stop testing after first match</li> </ul>	
,	Actions: 🗌 Get updated events	
Get status events		
	Run policy on deletes	
Policy: AddPolicyProcessMapping		
Event Locking: 🗌 Enable		
Expression:		
New M	apping: 🐺	
Select: (all)	Restriction Filter Policy Name Active Chain Move Edit	
Delete		
Order By:		
	Ţ	
Error on page.	Internet	

Figure 4-12 Event mapping

- 4. Selecting the **Get updated events** check box ensures that new and changed events are processed. Changed events are identified in the ObjectServer alerts.status table StateChange column, while new events are identified by the Serial column.
- 5. Select the appropriate options and check them.
- 6. Now add a new mapping by clicking **New Mapping** and give it a filter expression relevant to the events against which you want your policy to run. Refer to Figure 4-13 on page 103 for more information. Without the filter expression and with the process updating events, you may be processing any changes to events, even those that are not necessary.

http://9.42	.170.179:8081 - Create New Mapping - Microsoft Internet E 🖃 🗆 🔀
	Create a New Event Filter:
	Filter Expression:
	Policy to Run: AddPolicyProcessMapping
	Active:
	Chain: 🗌
	? OK Cancel
A Doop	
C DOLE	

Figure 4-13 Event filter

7. Select the policy you wan to run against the filtered events, as shown in Figure 4-14. The policies can be defined as chained to ensure the correct order of execution of the policies.

http://9.42.170.179:8081 - Cree	ate New Mapping - Microsoft I	nternet E 📮 🗖 🔀
Create a New Event Filter:		
Filter Europeier		
Filter Expression	•	
Policy to Run	: AddPolicyProcessMapping	¥
Active	AddPolicyProcessMapping	
,	DeployProject	
Chain	DiscreteOidSetTest	
	DiscreteOidTest	
	Export	-
	PackedOidSetTest	
?	PackedOidTest	
	PrecisionSampleListenerPolicy	
	PrecisionSamplePolicy	
	SNMPGetActionTest	
Done	SNMPGetNextActionTest	Internet
bone	SNMPSetActionTest	incernee .
	SNMPTableSetTest	
	SNMPTableTest	
	SNMP1rapAction1est	
	XmlFileTestPolicy	
	XmlHttpTestPolicy	
	XmlStringTestPolicy	
	XmlXsdFileTestPolicy	

Figure 4-14 New event mapping

8. Check the Active box.

# 4.5 Event Processor

This section describes the Event Processor. The discussion is covered in the following sections:

- ▶ 4.5.1, "Event Processor definition" on page 104
- ► 4.5.2, "Event Processor configuration" on page 104

#### 4.5.1 Event Processor definition

The Event Processor is the service responsible for managing events coming into IBM Tivoli Netcool/Impact through the Event Reader, Event Listener, and E-mail Reader Services. The Event Processor manages the incoming event queue and is responsible for sending queued events to the Policy Engine for processing.

#### 4.5.2 Event Processor configuration

The Event Processor in IBM Tivoli Netcool/Impact is configured from the EventProcessor entry in the Service Status window, as shown in Figure 4-15.

Service Status Auto Refresh	
🔴 DefaultEmailReader	
DefaultEventReader	
🔴 DefaultJabberReader	
DefaultPolicyActivator	
🔴 EmailSender	
EventProcessor	
🥏 HibernatingPolicyActivator	
JMSMessageListener	
🔴 JabberService	
PolicyLogger	
PrecisionEventListener	
SelfMonitoring	
	×

Figure 4-15 Service Status window

Clicking the configuration icon (I) will open a window like the one shown in Figure 4-16 on page 105. You can make changes in this window.

🕙 http://9.4	2.170.179:8081 - EventProcessor Configuration - Microsoft 💷 🗖 🔀
	* required field EventProcessor Service:
	Number of Simultaneous Threads: * 2
	Number of Events per Query:*10
	Event Fetch Rate:* 3000 🖕 (millisecs)
	Clear Queue: Clear
	Service Log: Write to file
	? OK Cancel
🛃 Done	🔮 Internet 🦽

Figure 4-16 Event Processor configuration

The Event Fetch Rate field indicates the interval that the Event Processor waits after finishing its processing before fetching the next block of events. The Event Processor is capable of running as a multi-threaded service, and each thread can process a single alert at one time. Adding threads may allow events to be processed faster. The multi-threaded Event Processor is prevented from having a race condition by an event locking mechanism that ensures events with the same keys (such as Node and AlertKey) are serialized.

# 4.6 Logging configuration

All of the services that are listed in the Service Status part of the GUI have a logging configuration option on their configuration page. For example, see the Event Reader configuration in Figure 4-17. If you click the **Service Log** check box, this will start the logging to a file.

🔄 http://9.42.170.179:8081 - EventReader Configuration - Microsoft Internet Explo	rer 💶 🖂
General Settings Event Mapping	
General Settings:	* required field
Service Name:* DefaultEventReader	
ObjectServer Data Source: defaultobjectserver 💌	
Polling Interval:* 3000 🛱 (milliseconds)	
Restrict Fields: Fields	
Startup: Automatically when server starts	
Service Log: Write to file	
Collect Reports: Enable	
Clear State: Clear	
?	OK Cancel
실 Done, but with errors on page. 🧉	Internet 💥

Figure 4-17 Event Reader configuration

The log file is stored in the \$NCHOME/impact/log/. The log files use the naming convention <Impact_instance>_<servicename>, so for the IBM Tivoli Netcool/Impact server, the log file for the Event Reader is NCI EventReader.log.

The IBM Tivoli Netcool/Impact components create a log file with their status and activity. By default, this file is named netcool.log and is located in the \$NCHOME/log directory. The server log contains status, debugging, and error messages generated during run time by the IBM Tivoli Netcool/Impact server and the GUI server. By default, IBM Tivoli Netcool/Impact components print messages to the netcool.log file until it reaches 10 MB in size. This file is set by running \$NCHOME/wasce/var/log/server-log4j.properties. The most detailed log level is DEBUG.

An excerpt of server-log4j.properties is shown in Example 4-1.

Example 4-1 Content of server-log4j.properties

```
# set the default priority to INFO for all micromuse/ibm loggers
log4j.category.com.ibm.tivoli=INFO,NETCOOL
log4j.category.com.micromuse=INF0,NETCOOL
# NETCOOL appender information
log4j.appender.NETCOOL=org.apache.log4j.RollingFileAppender
log4j.appender.NETCOOL.threshold=DEBUG
log4j.appender.NETCOOL.layout=org.apache.log4j.PatternLayout
log4j.appender.NETCOOL.layout.ConversionPattern=%d{ABSOLUTE} %-5p
[%c{1}] %m%n
log4j.appender.NETCOOL.append=true
log4j.appender.NETCOOL.file=${org.apache.geronimo.base.dir}/../log/netc
ool.log
log4j.appender.NETCOOL.bufferedIO=false
log4j.appender.NETCOOL.maxBackupIndex=3
log4j.appender.NETCOOL.maxFileSize=10MB
# override any specific loggers
log4j.category.com.micromuse.common.util.Debug=DEBUG
log4j.additivity.com.ibm.tivoli=false
log4j.additivity.com.micromuse=false
```

The PolicyLogger Service provides a configurable way of logging, as shown in Figure 4-18.

🛃 http://9.42.170.179:8081 - PolicyLogger Configuration - Microsoft Internet Explo 🗔 🔲 🛛
A Delicul accor Comica
PolicyLogger Service:
Error-handling Policy: DefaultExceptionHandler
Highest Log Level: 0 💌
Log What: All SQL statements
Pre-execution Action Module Parameters
Post-execution Action Module Parameters
All Action Module Parameters
Policy Profiling: Enable
Service Log: 🗌 Write to file
Append Thread Name to Log File Name
Append Policy Name to Log File Name
Collect Reports: Enable
OK Cancel 🗸
🕘 Done 🔮 Internet

Figure 4-18 PolicyLogger

Important policy logging properties are:

- All SQL statements
- Pre-execution action parameters
- Post-execution action parameters
- All action module parameters

**Note:** The Append check boxes allow you to create separate log files for the different policies and threads.

The log for the Policy Logger can have the following names:

- The default name is <impact_server>_policylogger.log.
- If the Append Policy Name to Log File Name check box is checked, the name is <impact_server>_policylogger_<Policyname>.log.
- If the Append Thread Name to Log File Name check box is also checked, the name is <impact_server>_policylogger_<thread>.log.

Policy profiling calculates the amount of time a policy takes to process and output it to the log file. In general, logging for IBM Tivoli Netcool/Impact can be selected from the IBM Tivoli Netcool/Webtop GUI by selecting the service **Write to file** and clicking **OK**.

# 4.7 Operator View configuration

The Operator View is a tool that allows users in a network operations environment to run a policy from within the IBM Tivoli Netcool/OMNIbus event list and then view the output from the policy. The Operator View allows users of IBM Tivoli Netcool/OMNIbus to view additional information stored in IBM Tivoli Netcool/Impact relating to the event selected in the event list. It provides the operator with a custom view that can include external data for speeding up work. Typical examples for operator views are customer details, quick fix, documentation, similar problems, and so on.

Basic operator views allow users to view real-time IBM Tivoli Netcool/Impact data in a pre-formatted Web page and to run policies based on that data. A basic operator view has the following display elements:

- An event window, which displays incoming event information from IBM Tivoli Netcool/OMNIbus or information from another application that can be expressed in name/value pairs.
- An actions window, which allows users to run one or more policies from within the operator view.
- An information groups window, which displays sets of data that you define when you create the view, or when you manually edit the operator view policy. The data can be displayed based on a selected Key (GetByKey function) or by specifying a Filter (GetByFilter function).

Operator views can be configured from the IBM Tivoli Netcool/Impact GUI by clicking the **Operator Views** tab, as shown in Figure 4-19.

Projects Global	
Clear All Source Control File Locking:	8
Data Sources	-
🕲 Data Types	=
Operator Views	
New Operator View:	÷

Figure 4-19 Operator View configuration

The default configuration window for the Operator View opens, as shown in Figure 4-20.

2
,
* required field

Figure 4-20 Operator View configuration

It is also possible to create custom, advanced operator views. You can modify the operator view by:

- Editing the Operator View definition policy. The operator view is generated based on a policy with a name of Opview_<viewname>.
- Editing the Operator View HTML file.

For an example of a custom operator view that is viewed by way of a Web browser, refer to Figure 4-21 on page 111. This view can be accessed from the list of operator views at http://hostname:port/opview.

Server	Dashb	oard								1	Server	: d02rdb108
							Contact/Support D	etails				
Configuratio	on Manage	ement					Contact Group	Role	e	D	etails	
Serv	er	Туре	Oper	ating System	n	IP Address	Ima Loof	DBA South	- bury	Office: 555	-555-5	555
d02rdb108 Pseries AIX					9.45.74.171				Cell: 567-8	90-21	34	
	Purpose Application									Pager: 123	-456-7	7890
Purpose Application			Location			Kevin Morris	CCE DPE		Office: 545	-565-5	765	
DB2	DB2 CCE South		Southbu	ry						Cell: 567-9	89-98	98
										Pager: 545	-555-5	5665
							Ken Knapp	SDC North	AVM	Office: 536	-535-9	934
Change Ma	nagement									Cell: 227-9	65-99	07
Recent Chan	nes Risk	Schedu	led Start	Scheduled	End	Abstract				Pager: 535	-333-5	335
848981	Medium	06-10 21	1:00	06-10 23:00	)	APAR Updates						
848324	High	06-11 15	5:00	06-11 19:00	ţ.	Disk Replacement						
Number 29762127	SORT (Sale: Tracking) ap ("STAGE" db processes tl EVENTAPPL	Service Im s out Repo plication: b 2 server). t nat are dov Y and RPT	nting and 03edrdb0 the DPROI vn are call r_APPLY2	01 delay ed availa	ed, B ase a able, 1	Im Reporting and Tracking partner incentive paymusiness Partner and IE nd Business Partner s nowever, with the DPR	bact Statement - If the SORT application ints and IBM sales rep co M users will experience a atisfaction will decline. Co P applications not function	is not avail mmission: i significan urrently, the oning, the S	able, s will be t workload site is SORT	Cent Poughke	er epsie	AHE IBM S&D
Omnibus Ev	vents Node			Cobbu		Summary		Tally	Severity	Customer	Last	Occurrence
d02rdb108.so	uthbury.ibm.c	om		DVC Failed -	Ping	s Complete: Timed ou		302	5	Lenovo	1162	579115
d02rdb108.so	uthbury.ibm.c	om		Event based service d02r	attrib db10	ute IsmIcmpStatusRu 8.southbury.ibm.com h	es of template Host and as value Bad	1	5		1162	525167
Application:DB	2:d02rdb108	.southbury	.ibm.com	Overall Attrib DB2:d02rdb	ute of 108.s	f the Application tag of outhbury.ibm.com is B	ad.	1	5		1162	525 <mark>1</mark> 67
Application:DB2:d02rdb108.southbury.ibm.com			Host children of DB2:d02rdb108.southbury.ibm.com (d02rdb108.southbury.ibm.com) are Bad.			1	5		1162	525167		
d02rdb108.so	uthbury.ibm.c	om		Overall Attrib Bad.	ute o	f the Host tag of d02rdl	108.southbury.ibm.com i	s 1	5		1162	525167
Application:DE	2:d02rdb108	.southbury	.ibm.com	Overall Attrib	ute of	f DB2:d02rdb108.soutl	bury.ibm.com is Bad.	1	5		1162	525167
d02rdb108.so	uthbury.ibm.c	om		Overall Attrib	ute of	rd02rdb108.southbury	ibm.com is Bad.	1	5		1162	525167

Figure 4-21 Custom operator view

Because an operator view operates by basically running a policy, debugging and analysis of operator view errors can be performed by looking at the how the policy executes. You can find debugging information for an operator view from the Policy Logger output.



# 5

# **Development**

This chapter discusses policy development. We discuss this topic in the following sections:

- ► 5.1, "IBM Tivoli/Netcool Impact policy language (IPL)" on page 114
- ► 5.2, "Development requirements" on page 117
- ► 5.3, "Sample development" on page 119
- ► 5.4, "Verifying and testing the policy" on page 136
- ► 5.5, "Built-in functions" on page 138
- ► 5.6, "Exceptions in policy" on page 143
- ► 5.7, "Auto execute of policy" on page 145

# 5.1 IBM Tivoli/Netcool Impact policy language (IPL)

This section discusses IBM Tivoli/Netcool Impact policy language. A policy is a set of rules and actions that IBM Tivoli/Netcool Impact is required to perform when certain events or status conditions occur in your environment. Policies are implemented using IBM Tivoli/Netcool Impact policy language. IBM Tivoli/Netcool Impact policy language is the programming language that you use to write policies. We discuss important terms and concepts in IBM Tivoli/Netcool Impact policy language in the following sections:

- ► 5.1.1, "Common elements" on page 114
- ► 5.1.2, "Function" on page 114
- ► 5.1.3, "Operator" on page 115

#### 5.1.1 Common elements

The common elements of the IPL are:

Control structure	A control structure is a statement block in the IPL that is executed when the terms of the control condition are satisfied. The IPL supports If Then Else and When control structures.
Variable	A variable is an IPL keyword that represents a value or a set of values.
Key expression	A key expression is an expression that specifies the value that one or more key fields in a data item must have in order to be retrieved by the GetByKey function in the IPL.

#### 5.1.2 Function

A function is a named set of instructions in the IPL that accepts certain predefined input parameters and optionally returns a value or set of values. The functions of the IPL are:

Action function An action function is a built-in IPL function that performs a high-level task, such as retrieving data from a data source or sending e-mail. Action functions are predefined by the IPL and cannot be modified or extended when you write a policy.

Parser functionA parser function is a built-in IPL function that performs a<br/>low-level task, such as converting numeric and date<br/>formats or extracting a substring from a string. Parser<br/>functions are predefined by the IPL and cannot be<br/>modified or extended when you write a policy.

#### **User-defined function**

A user-defined function is a custom function that you use to organize code in a IBM Tivoli/Netcool Impact policy.

#### 5.1.3 Operator

An operator is a built-in IPL function that assigns a value to a variable, performs an operation on a value, or specifies how two values are compared in a policy. The operators of the IPL are:

#### Assignment operator

	The assignment operator is a built-in IPL function that assigns a value to a variable. The assignment operator is =.
Boolean operator	A Boolean operator is a built-in IPL function that specifies a logical operation of AND, OR, or NOT when IBM Tivoli/Netcool Impact evaluates sets of operations. The Boolean operators are &&, II, and !.
Comparison operato	r
	A comparison operator is a built-in IPL function that IBM Tivoli/Netcool Impact uses to compare two values. The comparison operators are $==$ , $!=$ , $<$ , $>$ , $<=$ , and $>=$ .
Mathematical operat	or
	A mathematical operator is a built-in IPL function that performs a mathematical operation on two values. The mathematical operators are +, -, *, /, and %.
String operator	A string operator is a built-in IPL function that performs an operation on two strings. IBM Tivoli/Netcool Impact supports one string operator that you can use for string concatenation. The string concatenation operator is +.

#### 5.1.4 IBM Tivoli Netcool/Webtop policy development types

When assessing the policy development requirements, you must consider the following items:

- Determine the functional category.
  - Event Enrichment policy: Adds new fields or data to existing fields within the event structure (alert.status).
  - X in Y Policy: Suppresses events until a certain number of identifiable events occur within a specified period of time. (The events will be stored in an internal data type.)
  - Event Notification: Generates a notification based on the event filter. Some notifications rely on updated events only, which is indicated in the StateChange field.
  - Event Gateway or Third-party integration: Forwards or transforms and forwards events to an external subsystem.
  - Event Correlation: Analyzes and connects two or more events to make problem determination easier for an operator.
  - Event Consolidation or Service Event: Combines events to generate an overview event for a certain service.
  - Message (E-mail Reader or IM reader) policy: Handles e-mail or instant messaging input.
  - Scheduling policy: A policy that is invoked based on a schedule.
- Develop IBM Tivoli Netcool/Impact policies.
  - Data types: User defined variables, arrays, and context.
  - Control structures: If statements and while statements.
  - Functions: Action, parser, and user defined.
  - Logging: Policy context and checkpoints.

# 5.2 Development requirements

Policy development is based on the policy's functional requirements. The policy must be used in an architectural design and policy objectives. You should draft pseudo-code or a flow chart to show the critical elements of a policy. This section explores a sample development scenario that describes the main tasks in policy development. We cover this topic in the following sections:

- ► 5.2.1, "Scenario" on page 117
- 5.2.2, "Development plan" on page 117
- 5.2.3, "Data information" on page 119

#### 5.2.1 Scenario

With multiple operators working on events, you want to reduce the time to resolve events in general. You want to provide the operators with all the information they need for their events so that they can make quick, decisive, and correct decisions about how to handle events. Using several databases that hold the customer details, SLA levels, penalties, and so on, you must enrich the event to use this information.

The requirements for the policy are:

- There must be read and write access to the event source (ObjectServer).
- There must be read access to the customer database (PostgreSQL).
- There must be data in the customer database or other database that links a customer name to a given node event field.
- There must be new ObjectServer fields added to the alerts.status table.

#### 5.2.2 Development plan

The development plan for the events is as follows:

- Only unprocessed events are processed.
- ► The node field of the event is checked against the customer table, and the customer details are fetched from the database.
- Previous customer data is parsed into the new ObjectServer fields.
- Events are marked as having been processed.

We then develop a pseudo-code for the policy by performing the following steps:

- 1. Check the event in order to determine whether ImpactFlag1 exists. ImpactFlag1 would be inserted at the end of this policy's processing to indicate that the events have been processed.
- 2. If ImpactFlag1 is not found, retrieve the Node value.
- 3. Use the Node value to inquire the Customer table about ImpactFlag1.
- 4. If the entry is not found, enter Unknown as the customer entry.
- 5. If the entry exists, retrieve the customer data and enrich the element.

The flowchart for the policy is shown in Figure 5-1.



Figure 5-1 Flow chart

#### 5.2.3 Data information

This section explains the data information that would be needed by the policy:

 The customer database schema has five tables defined, as shown in Figure 5-2.



Figure 5-2 Customer database schema and logical links

- The ObjectServer table alerts.status must be modified to contain the following extra fields:
  - Customer varchar(64)
  - WorkingHours varchar(64)
  - Department varchar(64)
  - ImpactFlag1 int

# 5.3 Sample development

This section discusses the development of the policy in the following sections:

- 5.3.1, "Create database fields in the ObjectServer" on page 120
- ► 5.3.2, "Define the data source for a customer table" on page 125
- ▶ 5.3.3, "Create the policy" on page 131
- ► 5.3.4, "Configure a service to use the policy" on page 132

#### 5.3.1 Create database fields in the ObjectServer

The following steps create database fields in the ObjectServer. You can only display columns of type CHAR, VARCHAR, INCR, INTEGER, and TIME in the event list. Do not add columns of other types in the alert.status table.

Perform these steps:

- 1. Create the required fields (Customer, WorkingHours, Department, and ImpactFlag1) in the alerts.status table of your ObjectServer(s).
- 2. Start the administrator configuration program for IBM Tivoli Netcool/OMNIbus:

UNIX: Run **\$NCHOME/omnibus/bin/nco_config**.

Windows: Launch the Administrator program.

The program window is shown in Figure 5-3.

B NETCOOL/O	MNIbus Administrator - Proce	ess Agents								
File Edit View	Item Tools Window Help				_					
Tivoli Netco	ol/OMNIbus									IBM.
- e e a	18 🗢 🗉 🗎 🛍 🗙	1								
Repo	ObjectServers									
	🗑 ObjectServer Report							3		
	Namez	Н	lost		P	ort	S	SL		
	DEMO	192.168.253.10			4100		🗶 false			
	ITNM38	172.16.2.65			4100		🗶 false			
	LOCAL	192.168.80.250			4100		X false			
	NCOMS	172.16.2.40	_		4100		X faise			
	NCOM32 NCOM32	192 168 80 129	_		4100		X false			
	NCOMS PREC	172.16.2.31			4100	-	X false			
	PRECISION	172.16.2.31			4100		🗙 false			
	TEST	172.16.2.58	_		4100		🗶 false			
	Bows: 9									
			_	-		_	_			
	Process Agents								×	
	ObjectServer Report       Image: Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point Point									
	Name /	Host	Port	SSL	Running	Pending	Dead	Save Req	u	
	NCO_PA	72.16.2.36	4200	🔀 false	•			🔀 false		
	Bows: 1									
Ready										

Figure 5-3 IBM Tivoli Netcool/OMNIbus Administrator

Double-click the NCOMS entry in the ObjectServer Report window and provide the credentials (the user has to have the DatabaseAdmin role) needed for database editing. The configuration panes that are available for the NCOMS ObjectServer will appear.



Figure 5-4 NCOMS configuration

3. Click **Databases**, click the + next to the alerts database, and select the status table, as shown in Figure 5-5.

Sector 2001/0MNIbus Administrator - Configuration of NCOMS on 172.16.2.40:4100										
🗱 File Edit View It	em Tools Window He	elp				_ 8 ×				
Tivoli Netcool/OM	Nibus					IBM.				
9 🔍 🖽 🕱	9 & Ħ 🕱 🖩 T I/I, ■ 🛛 🗮 🗈 🏠 🔧									
Navigator	🕨 👬 User	🗧 Databases, Tables	and Columns			3				
P Reports	Automation	🔚 🖓 alerts	Column Definitions Dat	ta View						
	🕨 👸 Visual	application_t	Name 🗠	Data Type	Length PrimaNo D No M System	n Hidden Ordin				
	🔻 🗱 System	backup_state	Acknowledged Agent	Integer VarChar	4 X f X f X f X f 64 X f X f X f X f	X f 24 ∧ X f 7				
	200	col_visuals	AlertGroup AlertKey	VarChar VarChar	255 X f., X f., X f., X f., X f., 255 X f., X f., X f., X f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f., K f.	¥ f 8 ★ f 9				
	Properties	colors	Class Customer	Integer VarChar	4 X f X f X f X f 64 X f X f X f X f	X f 19 X f 30				
		conversions	Department EventId	VarChar VarChar	64 X f., X f., X f., X f.,	× f 54 × f 26				
		details	ExpireTime FirstOccurrence		4 X E X E X E X E 4 X F X F X F X F X F	X f 27				
	Log Files	iduc_messag	Flash	Integer	4 <b>X</b> E <b>X</b> E <b>X</b> E	X f 25				
		journal	Grade Identifier	Integer VarChar	4 X L X L X L X L 255 √ t √ t √ t X f	X f 20 X f 2				
		login_failures	ImpactFlag1 ImpactFlag2	Integer Integer	4 X f X f X f X f 4 X f X f X f X f	X f 57 X f 58				
	Databases	objclass	InternalLast		4 X f X f X f X f 4 X F X F X F X F	X f 15				
	<u> </u>	objmenuitem:	LocalNodeAlias	VarChar	64 X f X f X f X f	X f 39				
	SQL	objmenus	LocalRootObj	VarChar	255 X f X f X f X f	× f 40				
		problem_eve	LocalSecUb _l Location	VarChar VarChar	255         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f <th f<="" th="" th<=""><th>X f 41 X f 21</th></th>	<th>X f 41 X f 21</th>	X f 41 X f 21			
		status	Manager NmosCauseType	VarChar Integer	64         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f         X f	X f 6 X f 38				
	Connections	v 10ac2802	NmosObjinst	Integer	4 X f X f X f X f	¥ f 37 ♥				
			Rows: 59							
	Session NCOMS on 17	72.16.2.40:4100 as root								
Connected										

Figure 5-5 alerts.status table

4. Right-click inside the column definitions pane and select **Add Column**, as shown in Figure 5-6 on page 123.

ns		Department	VarChar	64 🗶 f 🗶 f 🗶 f	🗶 f 🎗				
		EventId	VarChar	255 🗶 f 🗶 f 🗶 f	🗶 f 🔉				
		ExpireTime	Department	4 🗶 f 🗶 f 🗶 f	🗶 f 🎗				
20		FirstOccurrence	UTC	4 🗶 f 🗶 f 🗶 f	🗶 f 🔉				
ag		Flash	Integer	Сору	( f., 🎽				
		Grade	Integer	Parte	🕻 f.i. 🔉				
		Identifier	VarChar		( f., 🎽				
es		ImpactFlag1	Integer	Delete	📜 🕻 f.i. 🎽				
		ImpactFlag2	Integer	📔 Create Database	( f., 🎽				
		InternalLast	UTC		🕻 f.i. 🎾				
-mail	-	LastOccurrence	UTC		( f., 🎽				
ank		LocalNodeAlias	VarChar	🗙 Drop Table	🕻 f.i. 🎾				
		LocalPriObj	VarChar	Refresh Table	( f., 🎽				
		LocalRootObj	VarChar	Delete Data From Table	, 🕻 f 🎽				
ve		LocalSecObj	VarChar		î 🕴 🕻 f 🎽				
		Location	VarChar	Add Column	🕻 f.i. 🎾				
		Manager	VarChar	VarChar T Zedit Column					
		NmosCauseType	Integer	Jie Containin	🕻 f.i. 🎾				
		NmosObjInst	Integer	🕠 Drop Column	( f., 🕽				
J2'		<							
		Bauer EQ		12 Refresh					
>				🔜 🔿 Filter	<u>۲</u>				

Figure 5-6 Database options

5. Create a new VarChar field called Customer, as shown in Figure 5-7, and click **OK**.

Edit Column (NCOMS on 172.16.2.40:4100)										
📱 Column Details 💡										
Column Name: Customer										
Data Type:	VarChar	Length:	64 💭							
Attributes:	Primary key	📃 No Modify	No Default							
		ОК	Apply Cancel							

Figure 5-7 Customer field

6. Create a new VarChar field called WorkingHour, as shown in Figure 5-8, and click **OK**.



Figure 5-8 WorkingHours field

7. Create the Department and ImpactFlag1 fields and click OK for each one.

The alert.status table can also be updated by using the SQL command nco_sq1. The command to add a column is:

ALTER TABLE alert.status ADD COLUMN <column name> <column type>

#### 5.3.2 Define the data source for a customer table

The data source for the customer table described in Figure 5-2 on page 119 is defined as follows:

1. Create a new PostgreSQL data source called Customer that points to the customer database, as shown in Figure 5-9.



Figure 5-9 Choose database type

2. Click the + sign to create a new data source. Enter the connection information in the window, as shown in Figure 5-10.

General Settings:	required field	
Data Source Name:*	Customer	
Username:*	postgres	
Password:*	:	
Maximum SQL Connection:*	5	
Database Failure Policy:		
	0.5-1	
	Disable Backup	
Primary Source:		
· ·		
Host Name:*	172.16.2.40	
Port:*	5434	
Database:*	Customer	
	Test Connection	
Backup Source:		
Host Name:	Incalhost	
Dett	5432	
Port:		
Database:	database	
	Test Connection	
?	OK Cancel	

Figure 5-10 New data source details

3. Test the connection, as shown in Figure 5-11. If you have failover capabilities in the database, remember to test both connections and make sure that the databases are identical.

Microsoft Internet Explorer						
1	PostgreSQL: Connection is OK.					
	ОК					

Figure 5-11 Connection successful

4. Create a new data type called Node based on the previously created Customer data source, referring to the Node table. Figure 5-12 shows the data types that are available.



Figure 5-12 Data types

5. In the data type window shown in Figure 5-13, enter a unique data type name, enter the table name in the Base Table field, and then select the Key field.

General Se	ettings								
	Data T	ype Nam	e: * Node						
D	ata Sou	rce Nam	e: Custo	omer	~				
	Dis	splav Ico	n: 🙉 🗌	Browse					
		Chat	· •						
Table Dec	rintio	Stat	e: 🗹 En	abled					
Table Des	riptio								
	В	ase Tabl	e: Node		Re	fresh			
		New Fiel	d: 🐺						
Select: (all)	ID	Field Name	Format	Display Name	Description	Key Field	Move	Edit	
	node	node	STRING	node	node	✓	Ŧ	N	
	ip	ip	STRING	ip	ip		ŤŦ	N	
	nodeid	l nodeid	INTEGER	nodeid	nodeid		ŤŦ	Ì	
	custid	custid	INTEGER	custid	custid		ŤŦ		
	deptid	deptid	INTEGER	deptid	deptid		ŤŦ		
	servid	servid	INTEGER	l servid	servid		t		
Delete									
D	isplay N	ame Fiel	d: node	~					

Figure 5-13 New data type

In the Display Name Field, select the database field that represents the field that you want returned from the query. Click 🔚 to save.

6. Create a second data type for the customer table similar to the Node table, as shown in Figure 5-14 on page 129.

	Data Ti	ma Namai * Cu	tomor					
	Data Ty	pe Name: Cu	stomer					
	Data Sou	rce Name: Cu	stomer	×				
	Dis	play Icon: 🔇	Browse.					
		State: 🔽	Enabled					
able Description:								
				Det	fur ala			
Base Table: customer Refresh								
		New Field: 🛞						
Select: (all)	ID	Field Name	Format	Display Name	Description	Key Field	Move	Edit
	customer	customer	STRING	customer	customer		Ŧ	
	customerid	customerid	INTEGER	customerid	customerid		ŤŦ	N
	contactname	contactname	STRING	contactname	contactname		t.	
	contactnumber	contactnumber	STRING	contactnumber	contactnumber		ŤŦ	1
	servicelevel	servicelevel	STRING	servicelevel	servicelevel		tŧ	
	workinghours	workinghours	STRING	workinghours	workinghours		ŤŦ	
	locid	locid	INTEGER	locid	locid		Ť	

Figure 5-14 Data type customer fields

- 7. Create a department data type that refers to the Department table.
- 8. You can verify if the data type works correctly by clicking **I** to view data items. This will open the window shown in Figure 5-15, which shows the content of the table.

	A							
🔇 DataItem	ns: Node 🗙							
10 🖓 🌮	Filter:							
Data Ty	pe Name: Node							
Number	r of Objects: 43							
Select: (all)	node (key)	ip	nodeid	custid	deptid	servid	Links	Edit
	gambit	10.10.10.10	0	1	1	1	٩	Ì
	wombat	10.10.10.1	1	2	2	2	٩	Ì
	orac	10.10.10.2	2	3	3	3	Q	Ì
	muppet	10.10.10.3	3	4	4	0	Q	Ì
	moose	10.10.10.4	4	5	5	5	Q	
	hal	10.10.10.5	5	6	6	6	٩	Ì
	vixen	10.10.10.6	6	7	7	7	Q	Ì
					-			

Figure 5-15 View customer table data

- 9. Having created the data types, we now must create a link between these data types. We show some links in Figure 5-2 on page 119. Here we show the link between the Node table and Customer table based on the custld and customerId fields.
- 10.Select the node data type from the Data Type tab, and click the **Dynamic links** tab, as shown in Figure 5-16.

Edit: Node 🗙			
3			
ble Description Dynamic Links Cache Settings			
	Links From This Data Type:		
	New Link By Filter:	-	
	Select: Target (all) Target Type Category Value	e Link Edit Type	
	Delete		
	New Link By Key:	*	
	Select: Target (all) Target Type Category Value	e Exposed Link Edit Type	
	Delete		
	New Link By Policy:	-	
	Select: Target Data Type Category Value	e Exposed Link Edit Type	
	Delete		

Figure 5-16 Dynamic links

11. Click the 📰 button next to the New Link By Key and select the Target Data Type and foreign key expression that refers to the custid filed from the Node table (see Figure 5-17 on page 131).
|                      | Target Data Type: Customer                                                                                                                                                                                                                                                                                   |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| E                    | xposed Link Type: Customer 💌                                                                                                                                                                                                                                                                                 |
| Fo                   | preign key expression:                                                                                                                                                                                                                                                                                       |
| C                    | rustid                                                                                                                                                                                                                                                                                                       |
| -                    | he exercises entered here will be evaluated against the                                                                                                                                                                                                                                                      |
| ta<br>ta<br>th<br>da | Trent object. Its result will be used as a key into the<br>arget data type to dynamically discover this object's link<br>he expression may be built from one or more fields from<br>is object. For example, if you want the key into the target<br>at type to be a field in this object called "NodeName", I |

Figure 5-17 New Link By Key

- 12. Save the node data type by clicking the 🔚 button.
- 13. Verify if the links work by viewing the data for the node data type, and click the 'magnifying glass" called links, which provides a view of the selected linked data, as shown in Figure 5-18.

gambit(Node) View Data Item

Figure 5-18 Dynamic linked data types

The view shows the node name and its corresponding customer name from the other data type.

#### 5.3.3 Create the policy

This section explains how we create our policy.

Perform these steps:

 We write a policy in IBM Tivoli/Netcool Impact called EventEnrichmentOne that will complete the Customer, WorkingHours, and Department fields in IBM Tivoli Netcool/OMNIbus whenever a new event is entered into IBM Tivoli Netcool/OMNIbus. We use the ImpactFlag1 field to indicate whether IBM Tivoli/Netcool Impact has already processed this event. 2. Click the **Policies** tab and select the custom template from the drop-down menu. Name the policy EventEnrichmentOne and enter the policy code, as shown in Example 5-1.

Example 5-1 The EventEnrichmentOne policy

```
log("EventEnrichmentOne Starting");
@WorkingHours = "Unknown";
@Customer = "Unknown":
// Lookup corresponding entry in Node Table based on Node Field
NodeEntries = GetByKey("Node",@Node,1);
if (Length(NodeEntries) != 0) {
    log("EventEnrichmentOne - Nodes");
     // Lookup Customer entry based on link from PG Node to PG Customer
     CustomerEntries= GetByLinks({"Customer"},null,1000,NodeEntries);
     if(Length(CustomerEntries) !=0) {
        log("EventEnrichmentOne - Customer");
         @Customer=CustomerEntries[0].customer;
        @WorkingHours= CustomerEntries[0].workinghours;
     }
}
// Return information back to ObjectServer
// Set Flag as to not re-process
@ImpactFlag1=1;
ReturnEvent(EventContainer);
```

3. Save the policy by clicking the 🔚 button.

#### 5.3.4 Configure a service to use the policy

In this section, we associate the defined policy with a reader service. The reader service is responsible for associating incoming events with a policy that will process the event. We use the default Event Reader Service that only uses the ObjectServer as an event source. Another option is to configure the E-mail Reader Service, which requires defining the POP host, Polling Interval, and Login credentials to get to the POP host.

Perform these steps;

1. Click the default Event Reader in the Service Status pane, as shown in Figure 5-19 on page 133.

Service Status 🗹 Auto Refresh	
🔴 DefaultEmailReader	E > = -
🔴 DefaultEventReader	
🔴 DefaultJabberReader	
🔴 DefaultPolicyActivator	E > -
🔴 EmailSender	
EventProcessor	
🔴 HibernatingPolicyActivator	
🔵 JMSMessageListener	
🔴 JabberService	
PolicyLogger	
PrecisionEventListener	
SelfMonitoring	
	-

Figure 5-19 Service Status pane

2. The Event Reader setting includes a polling interval that specifies how often the Event Reader checks the ObjectServer for processing events, as shown in Figure 5-20.

General Settings Event Mapping	
General Settings:	* required f
Service Name:* DefaultEventReader	
ObjectServer Data Source: defaultobjectserver 💌	
Polling Interval:* 3000 🖕 (milliseconds)	
Restrict Fields: Fields	
Startup: Automatically when serve	r starts
Service Log: 🗌 Write to file	
Collect Reports: Enable	
Clear State: Clear	

Figure 5-20 Event Reader setting

3. Click the Event Mapping tab.

4. In the Event Mapping tab, add a new mapping by clicking the 📰 icon, as shown in Figure 5-21.

General Settings					
Event Mapping:					
Event Matching:  Test events with all filters Stop testing after first match					
Actions: 🗹 Get updated events					
Get status events					
Run policy on deletes					
Policy: AddPolicyProcessMapping					
Event Locking: Enable					
Expression:					
New Mapping: 🗰					
Select: (all) Restriction Filter Policy Name Active Chain Move Edit					
Delete					
Order By:					

Figure 5-21 Event mapping configuration

5. In the window shown in Figure 5-22, create a new event filter to only allow events from IBM Tivoli Netcool/OMNIbus that are filtered using ImpactFlag1=0 (the filter must be active when you start the reader), select the EventEnrichmentOne policy, select the Active check box, and click OK. The policy must already exist and you should have the correct scope.

Create a New Event Filter:
Filter Expression: ImpactFlag1=0
Policy to Run: EventEnrichmentOne
Active: 💌
Chain:
? OK Cancel

Figure 5-22 New event filter

The chain options in Figure 5-22 indicate that the selected policy is chained to other policies for the same filter. The policies are guaranteed to be executed serially.

6. The event mapping is added to the list, as shown in Figure 5-23.

Tag teral Settings Event Mappin	ng					
Event Map	ping:					
Event M	Aatching: ⊙ Test events with all filters ○ Stop testing after first match					
	Actions: 🗹 Get updated events					
Get status events						
Run policy on deletes						
Policy: AddPolicyProcessMapping						
Event	Locking: Enable					
	Expression:					
New N	Mapping: 🐺					
Select: (all)	Restriction Filter Policy Name Active Chain Move Edit					
	ImpactFlag1=0 EventEnrichmentOne 🔽 🗌 🥒					
Delete						
0	)rder By:					
?	OK Cancel					

Figure 5-23 Event mapping

# 5.4 Verifying and testing the policy

Policies can be tested once you start the Event Reader. This testing can be performed first in a test environment to ensure that the policy processing works. You can verify that the policy is working in several ways:

 By viewing the events in the IBM Tivoli Netcool/OMNIbus event console, as shown in Figure 5-24.

🔽 Netcool/OMNIbus Event List : Filter="All Events", View="Default"						
File Edit View Alerts Tools Help						
8833	All Events 🗖	Q Default 🗖	# # -	Top [ OFF ]		
WorkingHours	Customer	ImpactFlag1	Node	Alert Group	Summ	
12:00-15:00	BT	1	link4	Link	Link Down on port	
9:00-17:30	Netscape	1	hal	Stats	Diskspace alert	
9:00-12:00	Acme Ltd	1	link3	Link	Link Down on port	
9:00-13:00	AOL	1	vixen	<b>Stats</b>	Diskspace alert	
9:00-17:30	Microsoft	1	link6	Link	Link Down on port	
17:30-23:00	Sun	1	orac	Systems	Machine has gone offline	
12:00-14:00	Adobe	1	moose	Systems	Machine has gone offline	
9:00-17:30	Microsoft	1	muppet	Systems	Machine has gone offli <mark>ne</mark>	
6:00-12:00	France Telecom	1	wombat	Systems	Machine has gone offline	
Unknown	Unknown	1	link1	Link	Link Down on port	
9:00-17:30	Netscape	1	hal	Stats	Diskspace alert	
9:00-13:00	AOL	1	vixen	Stats	Diskspace alert	
10:00-14:00	UUNET	1	dewey	Link	Port failure : port reset	
12:00-14:00	Adobe	1	moose	Systems	Machine has gone online	
9:00-17:30	Microsoft	1	muppet	Systems	Machine has gone online	
6:00-12:00	France Telecom	1	wombat	Systems	Machine has gone online	
Unknown	Unknown	1	angel	Link	Port failure : port reset	
17:30-23:00	Sun	1	orac	Systems	Machine has gone online	
Unknown	Unknown	1	netcool	Administrator	A Administrator process running on netco	
Unknown	Unknown	1	netcool	Unix Event List	A e@AC100228@AC100228:1.0 process e@	
Unknown	Unknown	1	netcool	Unix Conductor	A c@AC100228@AC100228:1.0 process c@	
Unknown	Unknown	1	netcool	Administrator	A Administrator process running on netco	
8	2	12	2	10	0 All Events	
34 row(s) matched. 05/29/09 17:20:06 root NCOMS[PRI]						

Figure 5-24 IBM Tivoli Netcool/OMNIbus event list

▶ By looking at the Policy Logger log, as shown in Figure 5-25 on page 137.

PolicyLogger 🗙				
> • • • • • •				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
Parser log: EventEnrichmentOne Starting				
Parser log: EventEnrichmentOne - Nodes				
Parser log: EventEnrichmentOne - Customer				
r Parser iou; eventenrichmentone Starting				

Figure 5-25 Policy log

• By looking at the Event Reader log, as shown in Figure 5-26.

BefaultEventReader     A     Beau     Compare StateChange >= 129303393 AND Class == 10500 AND ((Impathag1=0)) order by stateChange ;     Read: 0 Wew Read: 0 Updates: 0 OSQueue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 37383112     Getting count where StateChange >= 124361338 AND Class == 10500 AND ((Impathag1=0))	
B      B     B     Control and based where StateChange >= 1243513937 And Class (= 10300 And ((timpetchag)=0)) order by StateChange ; Read: 0 New Read: 0 Updates: 0 OSQueue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 37383112     Getting count where StateChange >= 1243513938 AND Class (= 10500 AND ((Impetfiag1=0))	
l query setes. I nom are basedus merre suesciange > € (24501397 / mu class i≡ lubou Ang (umparingi≡u)) order by satechange ; Read: 0 New Read: 0 Updates: 0 OSqueue: 0 ReadBuffer: 22 mine: 0 Events Read(Sec: 0 New Events Read/Sec: 0 Memory; 37383112 Getting count where StateChange >= 1243613938 AND Class I= 10500 AND ((ImpactFlag1=0))	
Query: select "from alerts.status where StateChange >= 1243613930 AND Class I= 10500 AND ((ImpactFig1=0)) order by StateChange ; Read: 0 New Read: 0 Updates: 0 OSQueue: 0 Read/Buffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 3943528 Guery: select "from alerts.status where StateChange >= 1243613930 AND Class I= 10500 AND ((ImpactFig1=0)) order by StateChange ; Read: 0 New Read: 0 Updates: 0 OSQueue: 0 Read/Buffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 3943528 Guery: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 1243613939 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Query: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 124361394 AND Class I= 10500 AND ((ImpactFig1=0)) Guery: select "from alerts.status where StateChange >= 12436139	Lambor Langer 3= 1295013927 APU Langer 10000 APU (LINDBUT-RUJ = 0) Order by State-Lindber ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 37383112 45613938 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange > 1243513938 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 38422320 43513938 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange >= 1243513938 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 3942288 43513939 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange >= 124351399 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 39427968 4351399 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange >= 124351399 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 4015384 4051399 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 4015384 40513940 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 4050504 43513940 AND Class I= 10500 AND ((ImpactFlag1=0)) order by StateChange ; ue: 0 ReadBuffer: 22 Time: 0 Events Read/Sec: 0 New Events Read/Sec: 0 Memory: 4065248 43513940 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange >= 1243513940 AND Class I= 10500 AND ((ImpactFlag1=0)) StateChange >= 124351340 AND Class I= 10500 AND ((ImpactFlag1=0)) StateCh

Figure 5-26 Event Reader log

# 5.5 Built-in functions

This section lists the built-in function of IBM Tivoli/Netcool Impact. There are four types of functions in IBM Tivoli/Netcool Impact, as shown in the following sections:

- ► 5.5.1, "Action functions" on page 138
- ► 5.5.2, "Parser functions" on page 140
- ► 5.5.3, "Web services functions" on page 142
- 5.5.4, "SNMP functions" on page 143

#### 5.5.1 Action functions

The action functions are:

Activate: Runs another policy. The syntax is:

```
Activate([DataItem], [PolicyName])
```

 ActivateHibernation: Resumes a policy that has been suspended by the hibernate function. The syntax is:

ActivateHibernation(Hibernation)

AddDataItem: Adds a data item into a data type. The syntax is:

[DataItem =] AddDataItem(DataType, ContextToCopy)

 BatchDelete: Deletes a set of data items from a data type, typically for an SQL data type. The syntax is:

BatchDelete(DataType, [DeleteFilter], [DeleteDataItems])

 BatchUpdate: Updates a set of data items from a data type, typically for an SQL data type. The syntax is:

BatchUpdate(DataType, Filter, UpdateExpression)

CallDBFunction: Runs a column function from an SQL data item. The syntax is:

result = CallDBFunction(DataType, Filter, Metric)

CallStoredProcedure: Runs a database stored procedure. The syntax is:

[array] = CallStoredProcedure(DataSource, ProcName, Sp_Parameter)

 CommandResponse: Sends a series of commands using telnet, ssh, or tn3270. The syntax is:

Session = CommandResponse(Host, UserName, Password|UserCredentials, InitialPrompt|Options, Port, Timeout, Expiration)

• DeleteDataItem: Deletes a single data item from a data type.

 Deploy: Copies data sources, data types, policies, or services between server clusters. The syntax is:

Deploy(TargetCluster, Username, Password, Elements, ElementsOfType, CheckpointID)

 DirectSQL: Runs SQL commands directly to the database. It does not need to define an IBM Tivoli/Netcool Impact data type to represent a table. The syntax is:

[Array =] DirectSQL(DataSource, Query, CountOnly)

 GetByFilter: Retrieves data items from a data type using a filter (a WHERE SQL clause); optionally, it can only return the number of rows. The syntax is:

[Array =] GetByFilter(DataType, Filter, [CountOnly])

 GetByKey: Retrieves data items from a data type using a key field value. The syntax is:

```
[Array =] GetByKey(DataType, Key, [MaxNum])
```

 GetByLinks: Retrieves data items from a data type that has a link from a source data item. The syntax is:

[Array =] GetByLinks(DataTypes, [LinkFilter], [MaxNum], DataItems)

 GetHibernatingPolicies: Retrieves data items from a Hibernation data type. The syntax is:

[Array =] GetHibernatingPolicies(StartActionKey, EndActionKey, [MaxNum])

 GetScheduleMember: Retrieves schedule members associated with a particular time range or group of time. The syntax is:

[Array =] GetScheduleMember(Schedule, [TimeToMatch], [ReturnAll], Time)

Hibernate: Suspends the execution of a policy. The syntax is:

Hibernate(ActionKey, [Reason], TimeOut)

 JRExecAction: Executes an external command using the JRExec server. The syntax is:

JRExecAction(Command, Parameters, ExecuteOnQueue, TimeOut)

ReceiveJMSMessage: Receives a JMS message from a specific JMS destination. The syntax is:

ReceiveJMSMessage(JNDIProps, ConnectionFactory, Destination, MsgProps) RemoveHibernation: Deletes a data item from the Hibernation data type. The syntax is:

RemoveHibernation(ActionKey)

ReturnEvent: Inserts, updates, or deletes an event from an event source. For deleting an event, you set the @DeleteEvent member variable in the event container to True. The syntax is:

ReturnEvent(event)

► SendEmail: Sends an e-mail. The syntax is:

SendEmail([User], [Address], [Subject], Message, [Sender], ExecuteOnQueue)

SendInstantMessage: Sends an instant message. The syntax is:

SendInstantMessage(To, Group, Subject, TextMessage, ExecuteOnQueue)

SendJMSMessage: Sends a JMS message. The syntax is:

SendJMSMessage(JNDIProps, ConnectionFactory, Destination, MsgProps, Message)

 UpdateEventQueue: Updates or deletes events in the Event Reader event queue. The syntax is:

[Integer = ] UpdateEventQueue(EventReaderName, Filter, UpdateExpression, IsDelete)

#### 5.5.2 Parser functions

The parser functions are:

ClassOf: Provides the variable type class. The syntax is:

string = ClassOf(var)

- CurrentContext: Returns the current policy context. The policy context consists of all of the currently-defined variables in the policy, including EventContainer, DataItems, DataItem, and Num.
- Decrypt: Decrypts the string of a nci_crypt command or encrypt function. The syntax is:

string = Decrypt(expression)

Distinct: Returns a array of distinct content of another array. The syntax is:

```
array = Distinct(array, [UniqueClause])
```

Encrypt: Encrypts a string, with a result that is similar to the output of the nci_crypt command. The syntax is:

```
string = Encrypt(expression)
```

 Eval/EvalArray: Evaluates an expression based on a context complex variable or an array of complex variables. The syntax is:

```
<result> = Eval(expression, context)
<result> = EvalArray(expression, array)
```

- ► Exit: Terminates processing.
- Extract: Extracts words in a certain position based on the delimiter that separates the words. The index is counted from 0. The syntax is:

```
String = Extract(Expression, Index, [Delimiter])
```

- ► Float: Converts an expression into a float.
- ► FormatDuration: Converts a number of seconds into a date and time duration.
- GetDate: Returns the date and time as the number of seconds expired since the start of the UNIX epoch.
- ► Int: Converts an expression into an integer.
- Keys: Returns an array of strings that contains the field names of a data item.
- ► Length: Returns the number of elements of a variable or the length of a string.
- LocalTime: Returns the locally formatted time stamp from an integer number of seconds since the start of the UNIX epoch.
- ► Log: Logs a string into a Policy Logger.
- Merge: Merges two context variables. The syntax is:

[Target] = Merge(Target, Source, [Exclude])

► NewEvent: Creates a new event container. The syntax is:

EventContainer = NewEvent(EventReader)

- NewObject: Creates a new context variable.
- > ParseDate: Converts a time stamp value into an integer number of seconds.
- Random: Generates a random number between 0 and an upper bound.
- Replace: Replaces a substring of a given string. The syntax is:

String = Replace(Expression, Pattern, Substitution, MaxNum)

 RExtract: Uses a regular expression to extract a substring from a string. The syntax is:

```
String = RExtract(Expression, Pattern)
```

- RExtractAll: Performs RExtract multiple times to fill in an array. The syntax is: Array = RExtract(Expression, Pattern)
- Split: Returns an array of substrings from a string using a given delimiter. The syntax is:

```
Array = Split(Expression, Delimiters)
```

- String: Converts an expression into a string. The syntax is:
- Strip: Removes all characters of a mask string from another string. The syntax is:

```
String = Strip(Expression, Characters)
```

 Substring: Returns a substring using positional pointers. The index is counted from 0. The syntax is:

String = Substring(Expression, Start, End)

- ► ToLower: Converts an expression to lower case.
- ► ToUpper: Converts an expression to upper case.
- ► Trim: Trims leading and trailing whitespaces from a string.

#### 5.5.3 Web services functions

The Web services functions are:

► WSInvoke: Invokes a Web services message. The syntax is:

WSInvoke(WebServiceName, WebServicePort, EndPoint, MethodName, MethodParams, [Username], [Password])

- WSNewArray: Defines a complex array data from a WSDL file. The syntax is: WSNewArray(ElementType, ArrayLength)
- WSNewObject: Defines a complex element from a WSDL file. The syntax is: WSNewObject(ElementType)
- WSSetDefaultPKGName: Sets a default Java package name for the WSNewObject and WSNewArray functions. The syntax is:

WSSetDefaultPKGName(PackageName)

#### 5.5.4 SNMP functions

The SNMP functions represent existing SNMP commands:

SnmpGetAction: Runs SNMP Get. The syntax is:

SnmpGetAction(TypeName, [HostId], [Port], [VarIdList], [Community], [Timeout], [Version], [UserId], [AuthProtocol], [AuthPassword], [PrivPassword], [ContextId], [ContextName])

SnmpGetNextAction: Runs SNMP GetNext. The syntax is:

```
SnmpGetNextAction(TypeName, [HostId], [Port], [VarIdList],
[Community], [Timeout], [Version], [UserId], [AuthProtocol],
[AuthPassword], [PrivPassword], [ContextId], [ContextName])
```

SnmpSetAction: Runs SNMP Set. The syntax is:

SnmpSetAction(TypeName, [HostId], [Port], [VarIdList], ValueList, [Community], [Timeout], [Version], [UserId], [AuthProtocol], [AuthPassword], [PrivPassword], [ContextId], [ContextName])

SnmpTrapAction: Generates SNMP Trap. The syntax is:

SnmpTrapAction(HostId, Port, [VarIdList], [ValueList], [Community], [Timeout], [Version], [SysUpTime], [Enterprise], [GenericTrap], [SpecificTrap], [SnmpTrapOid])

# 5.6 Exceptions in policy

The IPL provides the means to raise and handle policy-level exceptions. You can also handle Java exceptions that are raised internally by IBM Tivoli/Netcool Impact during the execution of a policy.

### 5.6.1 Raising exceptions

To raise an exception, use the Raise keyword. Here is the syntax for Raise:

```
Raise ExceptionName(ExceptionText);
```

ExceptionName is a unique name for the exception and ExceptionText is the text output of the exception. This output is printed to the server log when the error is encountered. It can also be accessed inside an error handler using the ErrorMessage variable.

Example 5-2 shows an exception called IntOutOfRangeException.

Example 5-2 Sample exception

```
Function MyFunction(Param1, Param2) {
    If (Param1 < 0) {
        Raise IntOutOfRangeException("Param1 must be greater than 0");
    }
    . . .
}</pre>
```

#### 5.6.2 Handling exceptions

To handle an exception, you declare an exception handler. The handler is a function that is called each time a specific exception is raised at the policy level, or a specific Java exception is raised by IBM Tivoli/Netcool Impact during the execution of a policy.

Exception handlers must be declared in advance of any position where they are triggered in a policy. You should insert error handlers at the beginning of a policy before any other operations are specified. The following is the syntax for exception handlers:

```
Handle ExceptionName { statements ... }
```

ExceptionName is the name of the exception raised using the Raise keyword, or the name of the Java exception class raised by IBM Tivoli/Netcool Impact during the execution of the policy.

Example 5-3 shows how to handle policy-level exceptions using an exception handler.

Example 5-3 Sample policy level exception handler

```
Handle IntOutOfRangeException {
   Log("Error: Value of Param1 to MyFunction is less than 0");
}
```

Example 5-4 shows a Java exception handler for NullPointerException.

Example 5-4 Sample Java exception handler

```
Handle java.lang.NullPointerException {
    log("Null pointer exception in policy.");
}
```

### 5.6.3 Using exceptions

Exceptions can be used in a variety of different ways. Some of the useful constructs to use exceptions are:

- Trapping asynchronous errors (such as SQL problems, external script errors, and so on)
- Performing repetitive error handling functions
- Trapping Java exceptions and performing a recovery
- Generating a new event type in IBM Tivoli Netcool/OMNIbus as a result of an exception

# 5.7 Auto execute of policy

Determine the tasks that have to be performed in order to allow IBM Tivoli/Netcool Impact to autoexecute a policy:

- Determine which services and methods are necessary to automate policy execution.
- Determine the triggers and filters to automate policy execution within a service.
- ► Obtain the required data to enable the IBM Tivoli Netcool/Impact service.
  - E-mail server information (if an E-mail Reader Service is necessary)
  - Event Listener Service information (if an Event Listener Service is necessary)
  - ObjectServer connectivity information (if an Event Reader Service is necessary)
  - Instant messaging Jabber server information (If a Jabber Reader Service is necessary)
  - Java messaging bus information (if a JMS message Listener Service is necessary)
- > Determine the frequency at which to execute the policies.
- Configure new or existing services.
- For an Event Reader Service, create event mapping filters, making sure to avoid conflicts between policies and to achieve efficient processing.



# 6

# Administration

The administration of IBM Tivoli Netcool/Impact is discussed in the following sections:

- ► 6.1, "IBM Tivoli Netcool/Impact project creation" on page 148
- ► 6.2, "Configure users and roles" on page 149
- ► 6.3, "IBM Tivoli Netcool/Impact object deployment" on page 149
- ► 6.4, "Data export/import" on page 150
- ► 6.5, "Enterprise application redeployment" on page 151

# 6.1 IBM Tivoli Netcool/Impact project creation

Projects are a feature of IBM Tivoli Netcool/Impact that group together all the components for a particular piece of development work. The project allows you to organize and manage related:

- Data sources
- Data types
- Operator views
- Policies
- Services
- Wizards

When you create a project, you give it a name and save it. You can add to it any already existing applicable policies, data, documents, schedules, and so on. You can view all the project members in the Projects tab, as shown in Figure 6-1.

Projects Global			
Clear All Source Control File Locking:			
Data Sources			

Figure 6-1 Project tab

Apart from the project specific resources, IBM Tivoli Netcool/Impact also provides a global repository. The global repository is the storage area for all the IBM Tivoli Netcool/Impact policies, data, and services, and so on, for the server to which you are connected.

The only safe way to delete an item from a project is to remove it from the New or Edit Project window. This preserves the object in the IBM Tivoli Netcool/Impact repository. Deleting it from a list in the Navigation window removes it permanently from the IBM Tivoli Netcool/Impact database, which also removes it from all projects it is a member of and from the global repository.

# 6.2 Configure users and roles

The following tasks need to be performed in order to define the user level access that has to be created for each individual user or group (roles). Keep in mind the roles that were provisioned during installation.

Perform these steps:

- 1. Interview stakeholders about the desired user roles.
- 2. Determine the users that require access to IBM Tivoli Netcool/Impact.
- 3. Set up the appropriate role(s) in Security Manager.
- 4. Create the IBM Tivoli Netcool/Impact users in Security Manager.
- Assign roles to users in Security Manager to match your requirements. To use the IBM Tivoli Netcool/Impact GUI, the user has to be assigned the IMPACT_USER role.

# 6.3 IBM Tivoli Netcool/Impact object deployment

You can use the Automated Project Deployment Feature to perform deployment of IBM Tivoli Netcool/Impact objects. This will copy the data sources, data types, policies, and services in a project between two running server clusters on a network, as long as the servers are using the same IBM Tivoli Netcool/Impact name server. You can use this feature when moving projects from test environments into real-world production scenarios. It copies the latest version of the project. The Automated Project Deployment Feature uses a special IBM Tivoli Netcool/Impact policy called DeployProject to do its work. It can be invoked from the IBM Tivoli Netcool/Impact GUI or by running the **nci_deploy** command. To run the DeployProject policy, you must supply the target cluster name, user name, password, and the name of the project to copy, as shown in Figure 6-2.

Policy Runtime Parameters:
TargetCluster: NCI_CLUSTER_01
Username: admin
Password:
Project: MyProject
CheckpointID: MyProject_Ver_1_0
? Execute Close

Figure 6-2 DeployProject

You should check the policy log to ensure that the deployment completed without errors.

### 6.4 Data export/import

Data from a running instance of Tivoli Netcool Impact can be imported and exported and used on another instance of the same version of IBM Tivoli Netcool/Impact. Keep in mind that the files will still need to be manually sent to the other machine! The command-line tools for these operations are:

Configuration import

The nci_import command imports data that was previously exported from an instance of the IBM Tivoli Netcool/Impact server. This data includes all data sources, data types, policies, and services currently defined in the server instance. This tool is located in the \$NCHOME/impact/bin directory.

To run this tool, run the following command:

\$NCHOME/impact/bin/nci_import server_name directory

server_name is the name of the server instance where you want to import the data, and directory is the location that contains data exported by running the **nci_export** command.

Configuration export

The **nci_export** command exports data source, data type, service, and policy information from an instance of the IBM Tivoli Netcool/Impact server to a specified directory. This tool is located in the \$NCHOME/impact/bin directory.

To run this tool, run the following command:

\$NCHOME/impact/bin/nci_export server_name directory

server_name is the name of the server instance whose data you want to export, and directory is the path where you want to store the exported data.

## 6.5 Enterprise application redeployment

Whenever you make changes to the name server descriptor file (nameserver.props), it is necessary to redeploy the EAR files on the GUI server. You can perform the following types of GUI server redeployment:

- ► Hot redeployment, which does not require a restart of the application server.
- ► Cold redeployment, which requires a restart of the application server.

#### 6.5.1 Hot redeployment

A hot redeployment of the GUI server does not require you to restart the application server. WebSphere Application Server Community Edition automatically detects the new GUI server EAR file, and loads and starts the application. To perform a hot redeployment of the GUI server, perform the following steps:

1. Enter the following command to create a new GUI server EAR file:

\$NCHOME/guiserver/install/ncgui_createear

- 2. The ncgui_createear command creates a new EAR file named NC-guiserver-4.0.ear and moves it to the \$NCHOME/guiserver/deploy directory. The command overwrites any existing GUI server EAR file in that directory.
- 3. WebSphere Application Server Community Edition automatically refreshes the EAR file.

#### 6.5.2 Cold redeployment

To perform a cold redeployment of the GUI server, perform the following steps:

- 1. Stop WebSphere Application Server Community Edition, and run \$NCHOME/bin/wasce stop or stop the Windows service WebSphere Application Server Community Edition.
- Create a new GUI server EAR file by, under \$NCHOME/guiserver/install, running \$NCHOME/guiserver/install/ncgui_createear, or generate an existing EAR using the command \$NCHOME/impact/bin/nci_ear_editor old_ear new_ear.
- Remove the previous GUI server deployment files from \$NCHOME/wasce/config-store/n, where n is the deployment index. Remove the GUI server properties:
  - From index.properties under \$NCHOME/wasce/config-store, remove the line ibm/tivoli/guiserver/1.0/car=n.
  - From config.xml under \$NCHOME/wasce/var/config, remove the line
     <configuration</li>
     name="ibm/tivoli/guiserver/1.0/car"></configuration>.
- 4. Restart WebSphere Application Server Community Edition by running \$NCHOME/bin/wasce start or start the Windows service WebSphere Application Server Community Edition.

# 6.5.3 WebSphere Application Server Community Edition administration interface

The WebSphere Application Server Community Edition configuration can be viewed by using the URL http://<server>:8080/console, as shown in Figure 6-3 on page 153. To see which modules are installed, list the Application EAR files that are installed.



Figure 6-3 IBM Tivoli Netcool/Impact WebSphere Application Server Community Edition console



# 7

# Troubleshooting and performance tuning

In this chapter, we discuss some common troubleshooting and performance tuning issues regarding IBM Tivoli Netcool/Impact. The discussion is covered in the following sections:

- ► 7.1, "System verification" on page 156
- ► 7.2, "Performance reporting" on page 163
- 7.3, "Logging and testing" on page 166
- 7.4, "Impact CVS" on page 168

# 7.1 System verification

After successfully deploying IBM Tivoli Netcool/Impact, it is important to verify its requirements against the live system. IBM Tivoli Netcool/Impact provides a tool called the IBM Tivoli Netcool/Impact Configuration Documenter through the Web GUI that provides details about the items that have been configured.

The Configuration Documenter can be accessed form your Web browser by entering the URL http://hostname:port/clustername_servername_documenter. In our sample, IBM Tivoli Netcool/Impact is installed on YourPrimaryServer on the default port 8080. If we also use the default cluster name of NCICLUSTER and server name of NCI, the Configuration Documenter can be found at the following address:

http://YourPrimaryServer:8080/NCICLUSTER_NCI_documenter

The Configuration Documenter present four configuration items for your IBM Tivoli Netcool/Impact installation:

► The list of data sources, as shown in Figure 7-1.



Figure 7-1 Configuration Documenter data sources

►	The data	types,	as shown	Figure	7-2.
---	----------	--------	----------	--------	------

	COL Database
S Customer	SQL Database
W Node	SQL Database
VSDFTEST_JavaXML_Contents	DSA Mediator
SSDFTEST_JavaXML_Book	DSA Mediator
• XmlXsdFileTOC	DSA Mediator
SocketInfo	DSA Mediator
SocketData	DSA Mediator
Precision	DSA Mediator
XmlStringTOC	DSA Mediator
SmiHttpTOC	DSA Mediator
XmlFileTOC	DSA Mediator
STEST_JavaXML_Topic	DSA Mediator
STEST_JavaXML_Title	DSA Mediator
STEST_JavaXML_SectionBreak	DSA Mediator
STEST_JavaXML_Heading	DSA Mediator
STEST_JavaXML_Contents	DSA Mediator
REPORT_PolicyProfile	SQL Database
REPORT_EventStatus	SQL Database
REPORT_ImpactROI	SQL Database
REPORT_ROIProcess	SQL Database
REPORT_PolicyProcessMappings	SQL Database
REPORT_PolicyEfficiency	SQL Database
REPORT_OperatorEfficiency	SQL Database
REPORT_NodeEfficiency	SQL Database
REPORT_NodeEfficiencyDetails	SQL Database
REPORT_OperatorEfficiencyDetails	SQL Database
📧 LinkType	Internal Stored
🛍 Filter	Internal Stored
Doc	URL Document
C TimeRangeGroup	Internal Stored
Schedule	Schedule

Figure 7-2 Configuration Documenter data types 1

• The policies, as shown in Figure 7-3.

Policy Name	2
SNMPSetActionTest	
SNMPTableSetTest	
DeployProject	
AddPolicyProcessMapping	
XmlHttpTestPolicy	
PackedOidSetTest	
Export	
ReprocessFailedEvent	
TestSocketDSA	
Import	
DefaultExceptionHandler	
SNMPGetNextActionTest	
XmlXsdFileTestPolicy	
XmlStringTestPolicy	
SNMPTableTest	
SNMPTrapActionTest	
PrecisionSamplePolicy	
PackedOidTest	
DiscreteOidSetTest	
XmlFileTestPolicy	
SNMPGetActionTest	
EventEnrichmentOne	
PrecisionSampleListenerPolicy	
DiscreteOidTest	

Figure 7-3 Configuration Documenter policies

► The services, as shown in Figure 7-4.

Service Name	Service Class Name	
DatabaseListener	DatabaseListener	
DefaultPolicyActivator	PolicyActivator	
DefaultJabberReader	JabberReader	
CorbaNameService	CorbaNameService	
CommandExecutionManager	ExecuteQueueService	
SelfMonitoring	SelfMonitoring	
PolicyLogger	ActionTreeLogger	
DefaultEventReader	EventBroker	
PrecisionEventListener	EventListenerService	
JabberService	JabberService	
EventProcessor	MessageAcceptor	
CommandLineManager	CmdLineService	
HibernatingPolicyActivator	EscalationRooster	
DefaultEmailReader	EmailReader	
JMSMessageListener	JMSMessageListener	
EmailSender	EmailSenderService	

Figure 7-4 Configuration Documenter services

Clicking any of the listed items in these windows provides the relevant information for the item.

Here are some samples of the data that can be viewed:

► Figure 7-5 shows a sample customer data source.



Figure 7-5 Customer data source

 Figure 7-6 shows a sample customer data type from the customer data source.

GQL Database	Info							
	Name: 《	Customer						
Data Se	ource Name: 🤇	Customer						
DB	Table Name: o	ustomer						
Display	Name Field: o	ustomer						
	Key Fields: o	ustomerid						
	Enabled: 🕅	/						
						_		
ields								
						Default		
ield Name	Display Name	Description	Туре	Alias		Default Expression	Choices	<b>S</b> 1
<b>ield Name</b> ustomer	Display Name	Description	<b>Type</b> String	Alias customer		Default Expression	Choices	51 1
<b>ield Name</b> ustomer ustomerid	Display Name customer customerid	e Description customer customerid	<b>Type</b> String Integer	Alias customer customeri	id	Default Expression	Choices	51 1 X
<b>ield Name</b> ustomer ustomerid ontactname	Display Name customer customerid contactname	Description     customer     customerid     contactname	Type String Integer String	Alias customer customeri contactna	id	Default Expression	Choices	51 1 X X
ield Name ustomer ustomerid ontactname ontactnumber	Display Name customer customerid contactname contactnumbe	e Description customer customerid contactname er contactnumber	Type String Integer String String	Alias customer customeri contactna contactnu	id ame ımber	Default Expression	Choices	
ield Name ustomer ustomerid ontactname ontactnumber ervicelevel	Display Name customer customerid contactname contactnumbe servicelevel	e Description customer customerid contactname er contactnumber servicelevel	Type String Integer String String String	Alias customer customer contactna contactnu servicelev	id ame imber vel	Default Expression	Choices	ST 1 X X X X X X
ield Name ustomer ustomerid ontactname ontactnumber ervicelevel vorkinghours	Display Name customer customerid contactname contactnumbe servicelevel workinghours	e Description customer customerid contactname er contactnumber servicelevel workinghours	Type String Integer String String String String	Alias customer customer contactna contactnu servicelev workingh	id ame imber vel ours	Default Expression	Choices	ST 1 X X X X X X X

Figure 7-6 Customer data type

► Figure 7-7 shows a sample event enrichment policy logic.



Figure 7-7 EventEnrichmentPolicy

 Figure 7-8 on page 163 shows the Event Reader process with the changed policy information.

Service Info								
5	Service Name: 🔵	DefaultEventR	eader					
Servio	e Class Name: Eve	entBroker						
Service Current Status: Running: Input:0/min								
Service In Auto S	tart Up Mode: 💢							
	Log To File: X							
	-							
Service Propertie	5							
ObjectServe	r Poll Time (ms):	3000						
	Get Updates:	<b>V</b>						
	Fields:	*						
	lumber of Filters:	1						
IN IN		1						
IN .	Run All Filters:	V						
Object Server Da	Run All Filters: ta Source Name:	<pre>defaultobje</pre>	ctserver					
Object Server Da	Run All Filters: Ita Source Name: Collect Report:	defaultobje	ctserver					
Nobject Server Da	Run All Filters: ita Source Name: Collect Report:	defaultobje	ctserver					
Object Server Da Filters	Run All Filters: nta Source Name: Collect Report:	defaultobje	ctserver					
Nobject Server Da	Run All Filters: ata Source Name: Collect Report:	defaultobje	ctserver					
Object Server Da Filters Filter Number	Run All Filters: ata Source Name: Collect Report: Policy Name	defaultobje	ctserver Restriction Clause	Active?				

Figure 7-8 DefaultEventReader

# 7.2 Performance reporting

IBM Tivoli Netcool/Impact includes eight performance reporting tools:

- ► Two reports provide information about your network and network operators:
  - Node Efficiency Report

The Node Efficiency Report records information about the number of alerts generated by a node.

Operator Efficiency Report

The Operator Efficiency Report records how quickly operators respond to events that enter IBM Tivoli Netcool/Impact. It reports on the time between when the event occurs and when the operator owns it and acknowledges it.

- Six reports help you assess the efficiency of your IBM Tivoli Netcool/Impact configuration:
  - Action Efficiency Report

The Action Efficiency Report shows the total number of actions that have been processed over a time range that you select.

Action Error Report

The Action Error Report shows how many action errors have occurred in IBM Tivoli Netcool/Impact over a time period that you select.

- Impact Profile Report

The Impact Profile Report provides information about the efficiency of your IBM Tivoli Netcool/Impact configuration.

Impact Return on Investment (ROI) Report

The Impact Return on Investment (ROI) Report shows the operator time saved by IBM Tivoli Netcool/Impact to solve problems compared with the time it would take an operator to solve the identical problem manually. The manual times are calculated from industry statistics for common tasks. You associate the relevant policies with these calculations before you turn on report data collection in the Policy Logger Service.

Policy Efficiency Report

The Policy Efficiency Report records historical information about the performance of all your policies. Each time a policy executes, the time taken to execute is recorded.

- Policy Error Report

The Policy Error Report records historical information about how many errors have occurred in a policy over a time range that you select.

Some generic performance tips:

- When configuring an SQL database data source, SQL connections should be set equal to the number of policies that use the data source.
- Use data caching for improving access to heavily used tables.
- Provide a restriction filter to limit the number of rows to be processed.
- An Event Reader must be started and configured efficiently to process events, or the event queue may grow.

Additional tools are available to perform analysis (mainly on UNIX/Linux platforms), such as:

The top command shows the overall resource usage from processes. A sample output is shown in Figure 7-9 on page 165.

top - Tasks Cpu(s) Mem: Swap:	10:21:04 : 88 tota ): 1.2% ( 802860) 5931000	up al, us, k tot k tot	5:20 1 1 1.7 ⁹ al, al,	5, 3 u running % sy, 7800	users g, 80 0.0% 016k u 0k u	, loa 5 slea ni, 9 used, used,	ad epi 96.	avera ing, .8% ic 2284 593100	age: ( 0 st 1, 0. 14k fi 00k fi	0.09, 0.04 topped, .2% wa, 0 ree, 54 ree, 454	, 0.00 1 zombie .2% hi, 0.0% si 064k buffers 496k cached
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6090	root	15	0	39072	17m	5368	S	2.3	2.2	3:12.55	Х
30356	root	15	0	26448	13m	9540	S	1.0	1.7	0:00.49	gedit
6709	root	25	10	29928	16m	9844	S	0.7	2.0	2:37.21	rhn-applet-gui
5306	root	16	0	5492	2640	1308	S	0.3	0.3	0:23.94	hald
6725	root	15	0	16748	4552	3696	S	0.3	0.6	0:24.22	vmware-user
28030	root	17	0	651m	136m	2576	S	0.3	17.4	1:00.65	java
30371	root	16	0	3348	984	768	R	0.3	0.1	0:00.06	top
1	root	16	0	2820	552	472	S	0.0	0.1	0:01.05	init
2	root	RT	0	0	0	0	S	0.0	0.0	0:00.73	migration/0

Figure 7-9 The top command

- The iostat command shows the input/output performance.
- The vmstat command shows the overall system virtual memory statistic. The command takes two arguments: the interval between samples in seconds and the number of samples. See Figure 7-10 for an example.

[ro	ot@r	•h4_2 1	og]# vm	stat 3	2										
pro	cs -		mem	ory		SWa	ip	io		sys	tem		-сри		
r	b	swpd	free	buff	cache	si	S0	bi	bo	in	CS	us	sy i	d v	va
1	0	0	27036	54308	454512	0	0	13	6	508	136	1	29	6	1
0	0	0	27108	54316	454504	0	0	0	33	1020	320	1	1 9	8	0

Figure 7-10 The vmstat command

The netstat command shows the network connections between processes and the listening ports that are opened. Figure 7-11 shows checking the HTTP port for the IBM Tivoli Netcool/Impact GUI.

[root@rh4_2	2 log]#	netstat -an	grep 8080		
tcp	0	0 :::8080		:::*	LISTEN

Figure 7-11 The netstat command

The ps command allows you to list the running processes. It is the quickest way to see whether a process is running or not. Some of the IBM Tivoli Netcool/Impact processes are running in a Java virtual machine. Figure 7-12 shows the Java processes.

```
[root@rh4 2 log]# ps -ef | grep java
        28030
                  1 1 09:11 pts/1
root
                                      00:01:02
/opt/netcool/platform/linux2x86/jdk 1.5.0/bin/java -Xmx512M -Xms64M
-Xss512k -Xoss512k -XX:PermSize=512m -DXmx=512M
-Djava.awt.headless=true
-Djava.endorsed.dirs=/opt/netcool/wasce/lib/endorsed:/opt/netcool/pl
atform/linux2x86/jdk 1.5.0/jre/lib/endorsed
-Dorg.apache.geronimo.base.dir=/opt/netcool/wasce
-Djava.io.tmpdir=/opt/netcool/wasce/var/temp -jar
/opt/netcool/wasce/bin/server.jar --long
         30733 7238 0 10:31 pts/1
                                      00:00:00 grep java
root
```

Figure 7-12 The ps command

## 7.3 Logging and testing

This section discusses some logging and testing issues. We discuss these issues in the following sections:

- 7.3.1, "Log files" on page 166
- 7.3.2, "Event processing" on page 167
- 7.3.3, "Other troubleshooting tips" on page 167

#### 7.3.1 Log files

IBM Tivoli Netcool/Impact also provides extensive logging facilities. You can use IBM Tivoli Netcool/Impact and system tools so that performance issues may be assessed. The available tools are:

- An installation log that resides in \$NCHOME/log.txt.
- A centralized log for the IBM Tivoli Netcool/Impact server and IBM Tivoli Netcool GUI server resides in \$NCHOME/log/wasce.log.
- An IBM Tivoli Netcool/Impact components status and activity log resides in \$NCHOME/log/netcool.log.
#### 7.3.2 Event processing

IBM Tivoli Netcool/Impact mainly processes automation of events. The events come from IBM Tivoli Netcool/OMNIbus. This section discusses some general considerations for event processing flow troubleshooting. You can perform the following actions:

- 1. Check whether IBM Tivoli Netcool/OMNIbus and IBM Tivoli Netcool/Impact are running
- 2. Check whether the Event Reader and Event Processor services are running.
- 3. Check the Event Reader logging window to see if events are being retrieved from IBM Tivoli Netcool/OMNIbus.
- 4. Check the Policy Logger Log to determine if IBM Tivoli Netcool/Impact is processing the events.
- 5. Check the Write policy log to file to trace the execution of the policy.
- 6. Analyze which policy may be causing the problem by separating the Policy Logger by policy by appending the policy name to the log file.
- 7. Turn on policy profiling for the policy to get the run profile of the policy and provide a baseline of the policy execution.

#### 7.3.3 Other troubleshooting tips

Some general troubleshooting tips:

- Utilize server logs, such as netcool.log, (Server)_(Service).log, (Server)_Policylogger.log, and (Server)_Policylogger_PolicyName.log.
- Utilize operating system commands, such as netstat, uptime, iostat, sar, vmstat, and top.
- Utilize a Self-Monitoring Service.
  - Determine when to utilize it.
  - Enable it either by way of a GUI (IBM Tivoli Netcool/Impact Self-Monitoring Service) or CLI (IBM Tivoli Netcool/Impact Clustered mode) or by running (IBM Tivoli Netcool/Impact Service)_selfmonitoring.props.
  - Analyze monitoring types: Memory Status, Queue Status, Data Source Status, and Cluster Status.

- Utilize reporting tools.
  - Determine when to utilize them.
  - Enable them using the IBM Tivoli Netcool/Impact Policy Logger Service.
  - Analyze Network and Network Operators Reports: Node Efficiency Report and Operator Efficiency Report.
  - Analyze IBM Tivoli Netcool/Impact Configuration Reports: Action Efficiency Report, Action Error Report., Impact Profile Report, Impact Return on Investment Report, Policy Efficiency Report, and Policy Error Report.

Other activities that are needed for troubleshooting bugs and issues that are isolated within the system are:

- Review the bug tracking list or known issues.
- Increase the logging level in the Policy Logger Service, which is described in 4.6, "Logging configuration" on page 106.
- ► Replicate the bug.
- Review policy logs, Event Reader logs, and server logs to determine where bug(s) exist in the policy, which is described in 4.6, "Logging configuration" on page 106.
- Review the Event Reader configuration, if applicable. The IBM Tivoli Netcool/Impact Configuration Documenter, described in 7.1, "System verification" on page 156, can assist you in this task.

#### 7.4 Impact CVS

IBM Tivoli Netcool/Impact provides a version control interface that allows you to save IBM Tivoli Netcool/Impact policies, data sources, data types, and configuration properties as revisions in a source control archive. This interface supports the following version control systems:

- IBM Tivoli Netcool/Impact CVS
- External CVS
- RCS
- ClearCase®

IBM Tivoli Netcool/Impact CVS is a customized version of the Concurrent Revision System that has been prepared for use with IBM Tivoli Netcool/Impact. You can change the version control configuration at any time after you have installed IBM Tivoli Netcool/Impact by manually editing the version control properties file. This file is named *server_versioncontrol.props*, where server is the name of the instance of the IBM Tivoli Netcool/Impact server. This file is located in the \$\$NCHOME/impact/etc directory.

IBM Tivoli Netcool/Impact provides a script that you can use to access the version control system without using the native tools provided by the underlying system. This script is named nci_version_control and is located in the \$NCHOME/impact/bin directory. All file names must be specified relative to the \$NCHOME/impact directory. The generic syntax is:

nci_version_control <Impact_server_name> <command> <arguments>

The following list describes the specific functions that are available:

- Use the following syntax to check out a file from the version control system: \$NCHOME/impact/bin/nci_version_control server co filename username
- Use the following syntax to check in a file to the version control system: \$NCHOME/impact/bin/nci_version_control server ci filename comment username
- Use the following syntax to add a new file to the version control system: \$NCHOME/impact/bin/nci version control server add filename username
- Use the following syntax to remove a file to the version control system: \$NCHOME/impact/bin/nci version control server drop filename username
- Use the following syntax to remove the lock from a file that you have previously checked out without performing a new check in:

\$NCHOME/impact/bin/nci_version_control server unco filename username

- Use the following syntax to rename a file in the version control system: \$NCHOME/impact/bin/nci_version_control server mv old_filename new_filename username
- Use the following syntax to create a checkpoint:

\$NCHOME/impact/bin/nci_version_control server checkpoint checkpoint_id

Use the following syntax to check out the tip revision of all files in the version control system:

\$NCHOME/impact/bin/nci_version_control server update sandbox

**Note:** Although you can also use the version control script to check in and check out copies of policies files, data source files, data type files, and service properties files, you should not access the version control system for this level of management. Changes made on a per-server basis to these files are not propagated to other members of a server cluster.

The version control tool ensures that the IBM Tivoli Netcool/Impact environment is protected. When someone checks out a file, no modification is allowed by other users to the entity.

Some important files of the version control system are:

- The version control properties are stored in the \$NCHOME/impact/etc/<servername>_versioncontrol.props file.
- Locked files are defined in the \$NCHOME/impact/etc/<servername>_versioncontrol.locks file.
- Log files for the version control system are stored in the \$NCHOME/log/startcvs.log file.

# Α

### Sample test

This appendix contains a sample test and its answer key.

#### Sample test

This sample test is designed to give the candidate an idea of the content and format of the questions that will be on the certification exam. Performance on the sample test is *not* an indicator of performance on the certification exam and this should not be considered an assessment tool.

Answer the following questions:

- 1. A customer currently has several PERL scripts performing various tasks and would like to incorporate them into the IBM Tivoli Netcool/Impact policies without rewriting them. Which IBM Tivoli Netcool/Impact component can be used to achieve this goal?
  - a. nci_export
  - b. nci_jrexec
  - C. nci_trigger
  - d. nci_sendevent
- 2. The help desk is providing data for an event enrichment policy in the form of a comma delimited flat file that is written to the IBM Tivoli Netcool/Impact server system every night at midnight. The first field, the key field, contains an IP address. There are five other fields that contain the host name, the serial number, the physical location, a contact name, and a service level rating of 1 through 5, with 1 being the highest.

Currently, there is only interest in the service level rating data, but the other data may be needed in the future. The flat file contains a minimum of 1000 rows and a maximum of 5000 rows of data and the content changes daily.

What is the most appropriate and efficient way for IBM Tivoli Netcool/Impact to access this data in a production environment?

- a. Ask the help desk to supply the data in another format or from another source.
- b. Query the file directly using the IP address in the event to find the service level rating.
- c. Reformat the file so it only contains the necessary fields and query the newly formatted file.
- d. Write a policy that inserts the data into a custom PostgreSQL table on the IBM Tivoli Netcool/Impact server.

- 3. Which IBM Tivoli Netcool GUI component is not part of IBM Tivoli Netcool/Impact?
  - a. Policy Editor
  - b. Event Reader
  - c. Operator View
  - d. Active Event List
- 4. Which data source adapter (DSA) is read-only?
  - a. DB2 DSA
  - b. LDAP DSA
  - c. SNMP DSA
  - d. Oracle DSA
- 5. There is a default installation of IBM Tivoli Netcool/Impact on a server named appserver01. Which actions enable an administrator to verify which IBM Tivoli Netcool/Impact modules are installed and to view their current state?
  - a. Log in to the server at http://appserver01:8080/nci and click Reports.
  - b. Log in to the server at http://appserver01:8080/nci and click Services.
  - c. Log in to the server at http://appserver01:8080/console and click **Web App WARs**.
  - d. Log in to the server at http://appserver01:8080/console and click **Application EARs**.
- 6. In an IBM Tivoli Netcool/Impact server cluster, which property in the server properties file of each server specifies how often it should check the status of the other servers in the cluster?
  - a. impact.cluster.pinginterval
  - b. impact.cluster.pingtimeout
  - c. impact.cluster.statusinterval
  - d. impact.cluster.statuscheckinterval
- 7. When testing a data source that is configured with a backup data source in the Data Source list window, the following message is received: Connection could not be made. What does this message indicate?
  - a. Both the primary and backup data sources were tested and failed.
  - b. The primary data source failed and the backup should be tested separately.
  - c. The IBM Tivoli Netcool Security Manager instance could not authenticate the data source user.
  - d. One of the data sources failed, and they should be tested individually to determine which one failed.

- 8. Which three parameters are optional for the **nci_trigger** command? (Choose three.)
  - a. Password
  - b. User name
  - c. Serial field
  - d. Identifier field
  - e. Encrypted password
  - f. Name of the IBM Tivoli Netcool/Impact server
- 9. Policy A is to be executed any time a critical event occurs three or more times without being acknowledged. Policy B should *not* be executed if Policy A has already been run against the event.

Which statement is true about using a new ObjectServer field to prevent Policy B from running?

- a. Adding a field without refreshing the ObjectServer data source prevents the event from being picked up by the Event Reader.
- b. A new ObjectServer field should contain a value that only Policy A updates and include the new field as part of the event filter for Policy B.
- c. The probe rules file should be modified so that the new field is populated with a value of A or B so that the Event Reader knows which policy to execute.
- d. A new ObjectServer field cannot be used for this purpose. The Event Reader determines which policies are executed and it is an IBM Tivoli Netcool/Impact server service, not an ObjectServer service.

10. Why is a project created in IBM Tivoli Netcool/Impact?

- a. To group together all the policies for a particular type of event.
- b. To group together all the Event Readers, Event Processors, and policies.
- c. To group together all the components for a particular piece of development work.
- d. To group together all the data sources, data types, and data items for a database.

#### Sample answers

The following are the answers to the test questions:

B
 D
 D
 E
 D
 A
 A
 A
 C, D, and E
 B
 10.C



### **Related publications**

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

#### **IBM Redbooks**

For information about ordering these publications, see "How to get Redbooks" on page 178. Note that some of the documents referenced here may be available in softcopy only.

- Certification Guide Series: IBM Tivoli Business Service Manager V4.1.1 Implementation, SG24-7756
- Certification Guide Series: IBM Tivoli Netcool/OMNIbus V7.2 Implementation, SG24-7753
- Certification Guide Series: IBM Tivoli Netcool/Webtop 2.0 Implementation, SG24-7754

#### **Other publications**

These publications are also relevant as further information sources:

- IBM Tivoli Netcool/Impact Administration Guide, SC23-8853
- ► IBM Tivoli Netcool/Impact DSA Reference Guide, SC23-8856
- IBM Tivoli Netcool/Impact Policy Reference Guide, SC23-8855
- ▶ IBM Tivoli Netcool/Impact Release Notes, GI11-8132
- ► IBM Tivoli Netcool/Impact Solutions Guide, SC23-8857
- ► IBM Tivoli Netcool/Impact User Interface Guide, SC23-8854

#### **Online resources**

These Web sites are also relevant as further information sources:

► IBM Certification site:

http://www.ibm.com/certify/index.shtml

Test objectives:

http://www-03.ibm.com/certify/tests/obj938.shtml

#### How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks publications, at this Web site:

ibm.com/redbooks

#### **Help from IBM**

IBM Support and downloads

ibm.com/support

**IBM Global Services** 

ibm.com/services

### Index

#### Symbols

@DeleteEvent 140
\$NCHOME 64

#### Α

Accelerated Event Notification, see AEN Activate action function 138 ActivateHibernation 138 Active Event List, see AEL AddDataItem 138 AEL 43 AEN 32 AJP 35 Apache 35

#### В

basic IBM Tivoli Netcool/Impact cluster 36 BatchDelete 138 BatchUpdate 138

#### С

caching 39 CallDBFunction 138 CallStoredProcedure 138 certification benefits 3-4 checklist 5 objectives 8 recommended study resources 27 Tivoli 4 ClassOf 140 CLI 75 cluster installation 65 clustering 31 basic 36 failover 44 full 36 comma separated value, see CSV Command Execution Manager 36 command-line interface, see CLI Command-Line Manager 33, 36 CommandResponse 138

commands iostat 165 nc_db_postgres 80 ncgui_createear 151 nci_compilewsdl 40 nci_crypt 140 nci db 80 nci_deploy 150 nci_export 151 nci import 150 nci_jrexec 81 nci server 80 nci shutdown 81 nci trigger 97 nci_version_control 169 nco_config. 120 nco sql 124 ncsm_config 42 ncsm_db 70 netstat 53, 165 ps 166 SELECT 76, 78 setuparch.bin 63 top 164 **UPDATE** 75, 78 vmstat 165 Common Object Services, see COS communication 34 components 30 CORBA name service 33, 37 COS CSV 39 CurrentContext 140

#### D

data caching 39 data item 38 data models 31 data source 38 Data Source Adapter, see DSA data type 38 database 32 database listener 37 Decrypt parser function 140 default path 51 DeleteDataItem 138 Deploy 139 deployment types 35 Derby 35 DirectSQL 139 Distinct 140 DSA 38, 85

#### Е

e-mail readers 33 e-mail sender 33 Encrypt 141 Eval parser function 141 EvalArray parser function 141 Event 39 Event Listeners 32 Event Processor 33 Event Readers 32 event source 32 exception 39 Exit parser function 141 Extract parser function 141

#### F

failover configuration 44 Field 39 Flat File DSA 39 Float parser function 141 FormatDuration 141 full IBM Tivoli Netcool/Impact cluster 36

#### G

Generic Event Listener 37 Generic Event Reader 37 GetByFilter 139 GetByKey 139 GetByLinks 139 GetDate 141 GetHibernatingPolicies 139 GetScheduleMember 139 growth through skills 6 GUI server 31 gzipped 49

#### Н

Hibernate action function 139 hibernating policy activator 33

#### I

IBM Tivoli Netcool/Impact policy language, see IPL IMPACT_USER 43 impact.rc 81 installation 50 cluster 65 console 63 graphical 50 silent 64 instant messages 33 Int 141 iostat command 165 IPL 95, 114

#### J

JAAS port 35 Jabber Service 33 Java Messaging System, see JMS JMS 39 JMS DSA 39 JRExec 31, 64

#### K

Keys parser function 141

#### L

LDAP 31 LDAP DSA 40 Length parser function 141 Link 38 link dynamic 38 static 38 LocalTime 141 Log parser function 141 log.txt 64

#### Μ

Management Information Base, see MIB Mediator DSAs 40 Merge parser function 141 MIB 40

#### Ν

name server 31 nc_db_postgres command 80 ncgui_createear command 151 nci_compilewsdl command 40 nci_crypt command 140 nci db command 80 nci_deploy command 150 nci_export command 151 nci_import command 150 nci_jrexec command 81 nci_server command 80 nci_shutdown command 81 nci_trigger command 97 nci_version_control command 169 nco_config. command 120 nco_sql command 124 ncsm_config command 42 ncsm db command 70 netstat command 53, 165 NewEvent parser function 141 NewObject 141 NIS 31

#### 0

objectives 8 ObjectServer DSA 32

#### Ρ

ParseDate 141 policy logger 33 Precision DSA 40 Precision Event Listener 38 ps command 166

#### R

Random parser function 141 ReceiveJMSMessage 139 Redbooks Web site 178 Contact us xii RemoveHibernation 140 Replace parser function 141 return on investment, see ROI ReturnEvent 140 RExtract 141 RExtract All 142 ROI 5

#### S

Security Manager 31 SELECT command 76, 78 self-monitoring tool 73 SendEmail 140 SendInstantMessage 140 SendJMSMessage 140 server clustering 31 server startup 80 Service Status 79 setuparch.bin command 63 SME 8 SNMP DSA 40 Socket DSA 40 Software Value Incentive, see SVI Split 142 SQL database DSAs 40 SQL listener 37 startup 80 String parser function 142 Strip parser function 142 Subject Matter Expert, see SME Substring parser function 142 SVI 7-8

#### Т

TBSM 42 Tivoli Business Service Manager, see TBSM Tivoli Software Professional Certification 4 Tivoli Usage and Accounting Manager 23 ToLower 142 Tomcat 35 top command 164 ToUpper 142 Trim parser function 142

#### U

UPDATE command 75, 78 UpdateEventQueue 140

#### V

Value Advantage Plus, see VAP VAP 7–8 Virtual University Enterprises, see VUE vmstat command 165 VUE 5

#### W

Web Services Description Language, see WSDL Web services DSA 40 WebSphere Application Server Community Edition 31 WSDL 40

X XML DSA 41

(0.2"spine) 0.17"<->0.473" 90<->249 pages **Certification Guide Series: IBM Tivoli Netcool/Impact V4.0 Implementation** 

EEM 🛷 Redbooks

## **Certification Guide Series: IBM Tivoli Netcool/Impact V4.0 Implementation**



Detailed architecture and components discussion

Installation and configuration processing

Policy-based automation engine This IBM Redbooks publication is a study guide for the IBM Tivoli Netcool Impact V4.0 Implementation Certification test. It is aimed at the IT professional who wants to be an IBM Certified Professional for this product.

The IBM Tivoli Netcool/Impact V4.0 Implementation Certification test is offered through the IBM Professional Certification program. It is designed to validate the skills required of technical professionals who work in the implementation and deployment of IBM Tivoli Netcool/Impact V4.0.

This book provides the necessary information for understanding the subject matter. It includes sample questions. The sample questions help evaluate your personal progress. It familiarizes you with the types of questions that may be encountered in the exam.

This guide does not replace practical experience. It is not designed to be a stand-alone guide on the subject. Instead, this guide should be combined with educational activities and experiences and used as a very useful preparation guide for the exam.

For your convenience, the chapters are based on the certification objectives of the IBM Tivoli Netcool/Impact V4.0 Implementation Certification test. Those requirements are planning, prerequisites, installation, configuration, administration, and problem determination. Studying each chapter helps you prepare for the objectives of the exam.

#### INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

#### BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information: ibm.com/redbooks

SG24-7755-00

ISBN 0738433454

