

---

# NetIQ® AppManager®

## Application Programming Interface Guide

December 2018

## Legal Notice

For information about NetIQ legal notices, disclaimers, warranties, export and other use restrictions, U.S. Government restricted rights, patent policy, and FIPS compliance, see <https://www.netiq.com/company/legal/>.

© 2018 NetIQ Corporation. All Rights Reserved.

For information about NetIQ trademarks, see <https://www.netiq.com/company/legal/>. All third-party trademarks are the property of their respective owners.

# About this Book and the Library

The NetIQ AppManager product (AppManager) is a comprehensive solution for managing, diagnosing, and analyzing performance, availability, and health for a broad spectrum of operating environments, applications, services, and server hardware.

AppManager provides system administrators with a central, easy-to-use console to view critical server and application resources across the enterprise. With AppManager, administrative staff can monitor computer and application resources, check for potential problems, initiate responsive actions, automate routine tasks, and gather performance data for real-time and historical reporting and analysis.

## Intended Audience

This guide provides information about NetIQ AppManager API endpoints.

## Other Information in the Library

The library provides the following information resources:

### **Installation Guide for AppManager**

Provides complete information about AppManager pre-installation requirements and step-by-step installation procedures for all AppManager components.

### **User Guide for AppManager Control Center**

Provides complete information about managing groups of computers, including running jobs, responding to events, creating reports, and working with Control Center. A separate guide is available for the AppManager Operator Console.

### **Administrator Guide for AppManager**

Provides information about maintaining an AppManager management site, managing security, using scripts to handle AppManager tasks, and leveraging advanced configuration options.

### **Upgrade and Migration Guide for AppManager**

Provides complete information about how to upgrade from a previous version of AppManager.

### **Management guides**

Provide information about installing and monitoring specific applications with AppManager.

### **Help**

Provides context-sensitive information and step-by-step guidance for common tasks, as well as definitions for each field on each window.

The AppManager library is available in Adobe Acrobat (PDF) format from the [AppManager Documentation](#) page of the NetIQ website.



---

# Contents

<b>About this Book and the Library</b>	<b>3</b>
<b>About NetIQ Corporation</b>	<b>7</b>
<b>1 Introducing NetIQ AppManager Application Programming Interface</b>	<b>9</b>
<b>2 NetIQ AppManager API Endpoint Details</b>	<b>11</b>
2.1 Connect to Control Center Database (CCDB) . . . . .	11
2.1.1 Connect to CCDB . . . . .	11
2.1.2 Get list of all connected CCDBs . . . . .	12
2.2 Add QDB to CCDB . . . . .	13
2.2.1 Add QDB . . . . .	13
2.2.2 Get list of all added QDBs . . . . .	13
2.2.3 Check QDB sync status . . . . .	14
2.3 Management Group . . . . .	14
2.3.1 Create Management Group . . . . .	14
2.3.2 Update Management Group . . . . .	16
2.3.3 Get KS View under MG . . . . .	17
2.3.4 Get Servers View under MG . . . . .	18
2.3.5 Get Events View under MG . . . . .	19
2.3.6 Get Jobs View under MG . . . . .	21
2.3.7 Get Events View for a Server under Servers View . . . . .	23
2.3.8 Get Jobs View for a Server under Servers View . . . . .	25
2.4 Create Monitoring Policy . . . . .	27
2.4.1 Description . . . . .	27
2.4.2 Operation . . . . .	27
2.4.3 Path/Endpoint . . . . .	28
2.4.4 Resource . . . . .	28
2.5 Custom Property . . . . .	28
2.5.1 Create Custom Property . . . . .	28
2.5.2 Update Custom Property . . . . .	29
2.5.3 Delete Custom Property . . . . .	29
2.5.4 Get list of all Custom Properties . . . . .	30
2.6 Server Filters . . . . .	31
2.6.1 Set . . . . .	31
2.6.2 Get . . . . .	33
2.6.3 Delete . . . . .	35
2.7 Deployment . . . . .	37
2.7.1 Create deployment rule . . . . .	37
2.7.2 Get deployment rule . . . . .	42
2.7.3 Get Rule History (for AppManager 9.2 and later) . . . . .	43
2.7.4 Delete deployment Rule . . . . .	43
2.7.5 Enable deployment Rule . . . . .	44
2.7.6 Disable deployment Rule . . . . .	44
2.7.7 Get list of deployment rules . . . . .	45
2.7.8 Get single task for deployment rule . . . . .	46
2.7.9 Get all tasks for deployment rule . . . . .	46
2.7.10 Get Task status details . . . . .	48
2.7.11 Update Task status (i.e. Approve Task) . . . . .	48
2.7.12 Get Software Inventory . . . . .	49
2.8 Knowledge Script . . . . .	50

2.8.1	Copy KS/KSG	50
2.8.2	Delete KS/KSG	51
2.8.3	Propagate - All knowledge scripts to Derived Knowledge Scripts (for AppManager 9.2 and later)	51
2.8.4	Propagate - Selected Knowledge Scripts to Derived Knowledge Scripts (for AppManager 9.2 and later)	52
2.8.5	Propagate - All Knowledge Scripts to Ad Hoc Jobs (for AppManager 9.2 and later)	52
2.8.6	Propagate - Selected Knowledge Scripts to Ad Hoc Jobs (for AppManager 9.2 and later)	53
2.8.7	Propagate - Get KS list pending propagation to their Derived KSs (for AppManager 9.2 and later)	54
2.8.8	Propagate - Get KS list pending propagation to their Ad Hoc Jobs (for AppManager 9.2 and later)	54
2.9	Events	55
2.9.1	Update bulk events status to Acknowledge/Close/Delete (apply filter on Computer, Job ID, KS Name, Event ID)	55
2.9.2	Update Event status (Acknowledge/Close/Delete)	56
2.9.3	Get single event details	57
2.9.4	Get all events detail	57
2.10	Control Center Console Global option	58
2.10.1	To set default deployment rule credentials	58
2.11	Remote Deployment Package Mapping	59

# About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

## Our Viewpoint

### **Adapting to change and managing complexity and risk are nothing new**

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

### **Enabling critical business services, better and faster**

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

## Our Philosophy

### **Selling intelligent solutions, not just software**

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate — day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

### **Driving your success is our passion**

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with — for a change. Ultimately, when you succeed, we all succeed.

## Our Solutions

- ◆ Identity & Access Governance
- ◆ Access Management
- ◆ Security Management
- ◆ Systems & Application Management
- ◆ Workload Management
- ◆ Service Management

## Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

<b>Worldwide:</b>	<a href="http://www.netiq.com/about_netiq/officelocations.asp">www.netiq.com/about_netiq/officelocations.asp</a>
<b>United States and Canada:</b>	1-888-323-6768
<b>Email:</b>	<a href="mailto:info@netiq.com">info@netiq.com</a>
<b>Web Site:</b>	<a href="http://www.netiq.com">www.netiq.com</a>

## Contacting Technical Support

For specific product issues, contact our Technical Support team.

<b>Worldwide:</b>	<a href="http://www.netiq.com/support/contactinfo.asp">www.netiq.com/support/contactinfo.asp</a>
<b>North and South America:</b>	1-713-418-5555
<b>Europe, Middle East, and Africa:</b>	+353 (0) 91-782 677
<b>Email:</b>	<a href="mailto:support@netiq.com">support@netiq.com</a>
<b>Web Site:</b>	<a href="http://www.netiq.com/support">www.netiq.com/support</a>

## Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** at the bottom of any page in the HTML version of the documentation posted at [www.netiq.com/documentation](http://www.netiq.com/documentation). You can also email [Documentation-Feedback@netiq.com](mailto:Documentation-Feedback@netiq.com). We value your input and look forward to hearing from you.

## Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit <http://community.netiq.com>.



# 1 Introducing NetIQ AppManager Application Programming Interface

The NetIQ AppManager Application Programming Interface (NetIQ AppManager API) is a developer tool, which provides you with endpoints to perform common Control Center operations such as operations regarding Knowledge Script, event, management group, custom properties, remote deployment.

For other information regarding AppManager API, refer to the table below.

Topic	Link
Installation	<a href="#">Installing NetIQ AppManager Application Programming Interface</a>
System Requirements	<a href="#">NetIQ AppManager Application Programming Interface Requirements</a>
Reviewing Log Files	<a href="#">Reviewing AppManager Log Files</a>
Component Wise Requirement for TLS 1.2 Support	<a href="#">Component wise Prerequisites for TLS 1.2 Support</a>



# 2 NetIQ AppManager API Endpoint Details

The following format is used to understand the endpoints:

- ◆ Description: Overview of the operation
- ◆ Operation: The name of the operation
- ◆ Path/Endpoint: The name of the endpoint
- ◆ Resource: Resource obtained using the endpoint

---

**IMPORTANT:** JSON input tags are case sensitive.

---

Currently below Control Center operations can be performed using NetIQ API endpoints:

## 2.1 Connect to Control Center Database (CCDB)

### 2.1.1 Connect to CCDB

#### 2.1.1.1 Description

Connect to CCDB and return index of CCDB connected

---

**NOTE:** This allows user to perform operations on more than one CCDB using its index. This index is RepIndex, which is used in all endpoints. It is an integer value returned from POST operation on Repository endpoint.

---

#### 2.1.1.2 Operation

Name: **POST**

Format: **JSON**

Example:

```
{
  "server" : "server091",
  "instance" : "NQAATEST",
  "database" : "NQCCDB356",
  "user": "sa",
  "password" : "Control123"
}
```

#### 2.1.1.3 Path/Endpoint

**/Repository**

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository`

---

**NOTE:** "NQAppManagerAPI" is the site created in IIS

---

#### 2.1.1.4 Resource

Repository  
(i.e.,CCDB)

### 2.1.2 Get list of all connected CCDBs

#### 2.1.2.1 Description

List all connected CCDBs

#### 2.1.2.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{
  server: "server776"
  database: "NQCCDB"
  repositoryindex: 0
  version: "9.2.0.569"
}
```

#### 2.1.2.3 Path/Endpoint

/Repository

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository`

---

**NOTE:** "NQAppManagerAPI" is the site created in IIS. The "Service.svc" is a common attribute, which needs to be present in all endpoints.

---

#### 2.1.2.4 Resource

Repository  
(i.e., CCDB)

## 2.2 Add QDB to CCDB

### 2.2.1 Add QDB

#### 2.2.1.1 Description

Add QDB to CCDB

#### 2.2.1.2 Operation

Name: **POST**

Format: **JSON**

Example:

```
{
  "server" : "server075",
  "instance" : "SQLEXPRESS",
  "database": "QDB330",
  "user": "sa",
  "password": "Control123",
  "primary": "1"
}
```

#### 2.2.1.3 Path/Endpoint

/Repository/{repindex}/QDB

#### 2.2.1.4 Resource

QDB

## 2.2.2 Get list of all added QDBs

#### 2.2.2.1 Description

List all QDBs (added to CCDB)

#### 2.2.2.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{"database": "QDB", "datasourceid": "7fff985d-252d-4e3c-b913-36f519fc18a1", "modificationtime": "6/30/2017 8:29:05 AM", "primary": "1", "server": "server793", "status": 1, "version": "9.2.0.493" }
```

### 2.2.2.3 Path/Endpoint

/Repository/{repindex}/QDB

### 2.2.2.4 Resource

QDB

## 2.2.3 Check QDB sync status

### 2.2.3.1 Description

Return sync status of particular QDB

### 2.2.3.2 Operation

Name: **GET**

Format: **JSON**

Example:

"OK"

### 2.2.3.3 Path/Endpoint

/Repository/{repindex}/QDB/{datasourceid}/SyncStatus

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/QDB/ 0e838a16-f4a8-4101-95cc-82739ef82a33 /SyncStatus`

### 2.2.3.4 Resource

QDB

## 2.3 Management Group

### 2.3.1 Create Management Group

#### 2.3.1.1 Description

Create management group. Specify views/rules/server-groups as members. Specify KSGs to apply as a monitoring policy.

## 2.3.1.2 Operation

Name: **POST**

Format: **JSON**

Example 1: Create MG based on View

```
{
  "name": "SQL and NT Based MG",
  "description": "Test MG",
  "parentmgpath": "AppManager\\Master\\Management Group 3",
  "allowstatepropagation": 1,
  "membersdata": {"createstandardviews": 1, "members": [{"membertype": "view",
  "membername": "NT", "repository": ""}, {"membertype": "view", "membername": "SQL",
  "repository": "server776.qdb"}]},
}
```

Example 2: Create MG based on View and apply monitoring policy KSG\_TestSQL and KSG\_TestNT

```
{
  "name": "SQL and NT Based MG",
  "description": "Test MG",
  "parentmgpath": "AppManager\\Master\\Management Group 3",
  "allowstatepropagation": 1,
  "membersdata": {"createstandardviews": 1, "members": [{"membertype": "view",
  "membername": "NT", "repository": ""}, {"membertype": "view", "membername": "SQL",
  "repository": "server776.qdb"}]},
  "policiesdata": {"numberofrestartererrorjobs": 2, "ksglist": "KSG_TestSQL,
  KSG_TestNT"}
}
```

Example 3: Create MG based on Server Group and apply monitoring policy KSG\_TestSQL and KSG\_TestNT

```
{
  "name": "Server Group Based MG",
  "description": "Test MG",
  "parentmgpath": "AppManager\\Master\\Management Group 3",
  "allowstatepropagation": 1,
  "membersdata": {"createstandardviews": 1, "members": [{"membertype":
  "servergroup", "membername": "Master\\TestSG", "repository": ""}, {"membertype":
  "servergroup", "membername": "Master\\TestSG", "repository": "server776.qdb"}]},
  "policiesdata": {"numberofrestartererrorjobs": 1, "ksglist": "KSG_TestSQL,
  KSG_TestNT"}
}
```

Example 4: Create MG based on Rule and apply monitoring policy KSG\_TestSQL and KSG\_TestNT

```
{
  "name": "AmitMG3",
  "description": "Test MG",
  "parentmgpath": "AppManager\\Master",
  "allowstatepropagation": 1,
  "membersdata": {"createstandardviews": 1, "members": [{"membertype": "rule",
  "membername": "TestR", "repository": ""}, {"membertype": "view", "membername":
  "SQL", "repository": "server776.QDB"}]},
  "policiesdata": {"numberofrestartererrorjobs": 1, "ksglist": "KSG_TestSQL,
  KSG_TestNT"}
}
```

---

**NOTE:** If MG member is part of specific MG, then "repository" tag value has to be "<sqlserver>\<sqlinstance>.<qdb>". If MG member is part of "All Repositories" then "repository" tag can be empty or can be removed.

- ◆ For Rule based MG, "repository" tag can be empty or can be removed.
  - ◆ "create standard views" check box is available in "Members" tab while creating new MG.
  - ◆ "ksglist" is comma separated list of ksg
- 

### 2.3.1.3 Path/Endpoint

/Repository/{repindex}/ManagementGroup

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup`

### 2.3.1.4 Resource

Management Group

## 2.3.2 Update Management Group

### 2.3.2.1 Description

Modify Management Group. Update members as well as KSG list.

### 2.3.2.2 Operation

Name: **PUT**

Format: **JSON**

Example:

```
{
  "name": "SQL and NT Based MG",
  "description": "Test MG",
  "parentmgpath": "AppManager\\Master\\Management Group 3",
  "allowstatepropagation": 1,
  "membersdata": {"createstandardviews": 1, "members": [{"membertype": "view", "membername":
  "NT", "repository": ""}, {"membertype": "view", "membername": "SQL", "repository": "server776.qdb"}]}
}
```

---

**NOTE:** createstandardviews value is ignored.

---

### 2.3.2.3 Path/Endpoint

/Repository/{repindex}/ManagementGroup

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup`



## 2.3.2.4 Resource

Management Group

## 2.3.3 Get KS View under MG

### 2.3.3.1 Description

Get Knowledge Scripts View

### 2.3.3.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[2]
0:
{
  "Build Number": "7.1.36.0"
  "Category": "ACTION"
  "CreateTime": "3/13/2016 11:39:58 AM"
  "CreateUserID": "System.Byte[]"
  "Description": "Used with the CiscoCallMgr and CiscoUnity Discovery scripts to
create a server group and add the computer into that group."
  "Full Name": "Action_AddComputerToServerGroup"
  "IconID": "13"
  "IsKSBAGMEMBER": "False"
  "Is Knowledge Script Group": "False"
  "LangID": "1"
  "Modification Time": "3/13/2016 5:39:58 PM"
  "ParentVKSID": ""
  "Platform": "Windows"
  "SourceID": "65536"
  "Status": "Action"
  "VersID": "2"
  "Version": "1.0"
  "VirtualKSID": "1"
}
1:
{
  "Build Number": "7.8.60.0"
  "Category": "ACTION"
  "CreateTime": "3/13/2016 11:39:58 AM"
  "Description": "Trigger NetIQ Diagnostics Console to diagnose an NT, Exchange, or
Active Directory problem."
  "Full Name": "Action_Diagnose"
  "IconID": "13"
  "IsKSBAGMEMBER": "False"
  "Is Knowledge Script Group": "False"
  "LangID": "2"
  "Modification Time": "3/13/2016 5:39:58 PM"
  "ParentVKSID": ""
}
```

### 2.3.3.3 Path/Endpoint

Repository/{repIndex}/ManagementGroup/{mgname}/KnowledgeScripts/  
{ksviewname}?parentmgpath={parentmgpath}

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
KnowledgeScripts/S25vd2xlZGdlIFNjcmlwdHM?parentmgpath=QXBwTWFuYWdlcg
```

(where "TWFzdGVy" is base64encoded string of mgname "Master" and  
"S25vd2xlZGdlIFNjcmlwdHM" is base64encoded string of ksviewname "Knowledge Scripts" and  
"QXBwTWFuYWdlcg" is base64encoded string of parentmgpath "AppManager")

### 2.3.3.4 Resource

Knowledge Scripts View

## 2.3.4 Get Servers View under MG

### 2.3.4.1 Description

Get Servers View

### 2.3.4.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[1]  
0:  
{  
  "ActiveBias": "21600"  
  "Agent Name": "server776"  
  "Agent Status": "Online"  
  "Agent Version": "8.2.19.233"  
  "Bias": "25200"  
  "Component Name": "_AppManager_"  
  "Computer": "server776"  
  "DataSourceIntID": "3"  
  "FCObjID": "34"  
  "Header1": "Product Type"  
  "Header2": "OS Version"  
  "Header3": "CSD"  
  "Header4": "Build Number"  
  "Header5": "Install Time"  
  "Header6": "Current Type"  
  "Header7": "System Directory"  
  "Header8": "NT Domain (role)"  
  "Highest Event Severity Category": "1"  
  "IconID": "13"  
  "IsTopLevelObject": "True"  
  "LangID": "3"  
  "Last Discovered": "3/13/2016 12:04:26 PM"
```

```

"Maintenance Mode Status": "Online"
"ModificationTime": "6/6/2016 4:25:45 PM"
"ObjID": "34"
"Object Type Name": "NT_MachineFolder"
"ParentFCObjID": "34"
"Platform": null
"Primary Management Server": "server776"
"Relative Path": "server776"
"Repository": "server776.QDB"
"Repository Version": "9.2.0.569"
"RootMachineObjID": "34"
"Running Job Count": "7"
"Secondary Management Server": null
"Server Name": "server776"
"String1": "Windows Server 2008 R2 Enterprise Edition"
"String2": "6.1"
"String3": "Service Pack 1"
"String4": "7601"
"String5": "Tue Jul 29 04:07:52 2014"
"String6": "Multiprocessor Free:Virtual"
"String7":
"C:\Windows\system32"
"String8": "AMDOM009 (NT server)"
"SyncTime": "6/6/2016 4:25:53 PM"
"Time": "4/4/2016 11:14:27 AM"
"TypeID": "2"
"VersID": "2"
}

```

### 2.3.4.3 Path/Endpoint

Repository/{repIndex}/ManagementGroup/{mgname}/Servers  
 {serverviewname}?parentmgpath={parentmgpath}

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/Servers/U2VydmVycw?parentmgpath=QXBwTWFuYWdlcg>

(where "TWFzdGVy" is base64encoded string of mgname "Master" and "U2VydmVycw" is base64encoded string of serverviewname "Servers" and "QXBwTWFuYWdlcg" is base64encoded string of parentmgpath "AppManager")

### 2.3.4.4 Resource

Servers View

## 2.3.5 Get Events View under MG

### 2.3.5.1 Description

Get Events View

## 2.3.5.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[2]
0: {
  "Category Name": "Discovery"
  "ChildComment": "0"
  "Comment": null
  "DataSourceIntID": "3"
  "Dependent Job ID": null
  "ElementID": null
  "Event Count": "1"
  "Event ID": "1"
  "First Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "First Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Has Comment": "No"
  "Job ID": "14"
  "Knowledge Script": "Discovery_Networks-RT"
  "Knowledge Script Group Name": ""
}
[2]
0: {
  "Category Name": "Discovery"
  "ChildComment": "0"
  "Comment": null
  "DataSourceIntID": "3"
  "Dependent Job ID": null
  "ElementID": null
  "Event Count": "1"
  "Event ID": "1"
  "First Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "First Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Has Comment": "No"
  "Job ID": "14"
  "Knowledge Script": "Discovery_Networks-RT"
  "Knowledge Script Group Name": ""
  "Repository": "server776.QDB"
  "RootSrvObjID": "34"
  "Server Name": "server776"
  "Severity": "5"
  "Severity Category": "severe"
  "Status": "open"
  "SyncTime": "5/16/2016 11:18:12 AM"
  "UpdatingFlag": null
}-
1: {
  "Category Name": "Discovery"
  "ChildComment": "0"
  "Comment": null
  "DataSourceIntID": "3"
  "Dependent Job ID": null
  "ElementID": null
  "Event Count": "1"
  "Event ID": "2"
  "First Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "First Occurred (Console)": "3/13/2016 12:57:56 PM"
```

```

"Has Comment": "No"
"Job ID": "14"
"Knowledge Script": "Discovery_Networks-RT"
"Knowledge Script Group Name": ""
"Knowledge Script Name": "Networks-RT"
"Last Occurred (Agent)": "3/13/2016 12:57:56 PM"
"Last Occurred (Console)": "3/13/2016 12:57:56 PM"
"Logical Server Name": "server776"
"MachineObjID": "34"
"Message": "Networks ResponseTime Discovery Failed"
"Modification Time": "3/13/2016 6:57:57 PM"
"ObjID": null
"Parent Event ID": "1"
"Repository": "server776.QDB"
"RootSrvObjID": "34"
"Server Name": "server776"
"Severity": "5"
"Severity Category": "severe"
"Status": "open"
"SyncTime": "5/16/2016 11:18:12 AM"
"UpdatingFlag": null
}

```

### 2.3.5.3 Path/Endpoint

Repository/{repIndex}/ManagementGroup/{mgname}/Events/  
{eventviewname}?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/  
{servername}/Events?parentmgpath={parentmgpath}

Example

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
Events/RXZlbnRz?parentmgpath=QXBwTWFuYWdlcg

(where “TWFzdGVy” is base64encoded string of mgname “Master” and “RXZlbnRz” is  
base64encoded string of eventviewname “Events” and “QXBwTWFuYWdlcg” is base64encoded  
string of parentmgpath “AppManager”)

### 2.3.5.4 Resource

Event View

## 2.3.6 Get Jobs View under MG

### 2.3.6.1 Description

Get Jobs View

### 2.3.6.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[2]
0:
{
  "ActualKSName": "NT"
  "Category Name": "Discovery"
  "Child Job Count": null
  "Comment": ""
  "DataSourceIntID": "3"
  "Display As": ""
  "Drop KP Object ID": "34"
  "Enterprise Job ID": null
  "Is Master Job": "False"
  "Job ID": "2"
  "Job Type": "Ad Hoc (Knowledge Script)"
  "Knowledge Script": "Discovery_NT"
  "Knowledge Script Build Number": "7.8.60.0"
  "Knowledge Script Group
Name": ""
  "Knowledge Script Version": "1.0"
  "LangID": "1"
  "Last Run Time": "3/13/2016 12:04:17 PM"
  "MachineObjID": "34"
  "Master Job ID": null
  "Modification Time": "3/13/2016 6:04:32 PM"
  "NodeID": ""
  "NodeName": ""
  "ParentJobID": "1"
  "Repository": "server776.QDB"
  "Server Name": "server776"
  "SourceID": "65536"
  "Status": "Stopped"
  "Stop Time": "3/13/2016 12:04:32 PM"
  "Submit Time": "3/13/2016 12:04:10 PM"
  "SyncTime": "5/16/2016 11:18:23 AM"
  "User": "server776\Administrator"
  "VersID": "2"
  "Virtual Job ID": null
}
1:
{
  "ActualKSName": "AMHealth"
  "Category Name": "Discovery"
  "Child Job Count": null
  "Comment": ""
  "DataSourceIntID": "3"
  "Display As": ""
  "Drop KP Object ID": "34"
  "Enterprise Job ID": null
  "Is Master Job": "False"
  "Job ID": "4"
  "Job Type": "Ad Hoc (Knowledge Script)"
  "Knowledge Script": "Discovery_AMHealth"
  "Knowledge Script Build Number": "8.0.113.0"
  "Knowledge Script Group Name": ""
  "Knowledge Script Version": "1.0"
  "LangID": "2"
  "Last Run Time": "3/13/2016 1:15:08 PM"
  "MachineObjID": "34"
}
```

```
"Master Job ID": null
"Modification Time": "3/13/2016 7:15:13 PM"
"NodeID": ""
"NodeName": ""
"ParentJobID": "3"
"Repository": "server776.QDB"
"Server Name": "server776"
"SourceID": "65536"
"Status": "Stopped"
"Stop Time": "3/13/2016 1:15:13 PM"
"Submit Time": "3/13/2016 12:05:02 PM"
"SyncTime": "5/16/2016 11:18:23 AM"
"User": "server776\Administrator"
"VersID": "2"
"Virtual Job ID": null
}
```

### 2.3.6.3 Path/Endpoint

Repository/{repIndex}/ManagementGroup/{mgname}/Jobs/  
{jobviewname}?parentmgpath={parentmgpath}

Example

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
Jobs/Sm9icw?parentmgpath=QXBwTWFuYWdlcg

(where “TWFzdGVy” is base64encoded string of mgname “Master” and “Sm9icw” is  
base64encoded string of jobviewname “Jobs” and “QXBwTWFuYWdlcg” is base64encoded string of  
parentmgpath “AppManager”)

### 2.3.6.4 Resource

Jobs View

## 2.3.7 Get Events View for a Server under Servers View

### 2.3.7.1 Description

Get Events View

### 2.3.7.2 Operation

Name: **GET**

Format: **JSON**

Example:

```

[2]
0: {
  "Category Name": "Discovery"
  "ChildComment": "0"
  "Comment": null
  "DataSourceIntID": "3"
  "Dependent Job ID": null
  "ElementID": null
  "Event Count": "1"
  "Event ID": "1"
  "First Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "First Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Has Comment": "No"
  "Job ID": "14"
  "Knowledge Script": "Discovery_Networks-RT"
  "Knowledge Script Group Name": ""
  "Knowledge Script Name":
  "Networks-RT"
  "Last Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "Last Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Logical Server Name": "server776"
  "MachineObjID": "34"
  "Message": "Networks ResponseTime Discovery Failed"
  "Modification Time": "3/13/2016 6:57:57 PM"
  "ObjID": null
  "Parent Event ID": null
  "Repository": "server776.QDB"
  "RootSrvObjID": "34"
  "Server Name": "server776"
  "Severity": "5"
  "Severity Category": "severe"
  "Status": "open"
  "SyncTime": "5/16/2016 11:18:12 AM"
  "UpdatingFlag": null
}-
1: {
  "Category Name": "Discovery"
  "ChildComment": "0"
  "Comment": null
  "DataSourceIntID": "3"
  "Dependent Job ID": null
  "ElementID": null
  "Event Count": "1"
  "Event ID": "2"
  "First Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "First Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Has Comment": "No"
  "Job ID": "14"
  "Knowledge Script": "Discovery_Networks-RT"
  "Knowledge Script Group Name": ""
  "Knowledge Script Name": "Networks-RT"
  "Last Occurred (Agent)": "3/13/2016 12:57:56 PM"
  "Last Occurred (Console)": "3/13/2016 12:57:56 PM"
  "Logical Server Name": "server776"
  "MachineObjID": "34"

```



```
"Message": "Networks ResponseTime Discovery
Failed"
"Modification Time": "3/13/2016 6:57:57 PM"
"ObjID": null
"Parent Event ID": "1"
"Repository": "server776.QDB"
"RootSrvObjID": "34"
"Server Name": "server776"
"Severity": "5"
"Severity Category": "severe"
"Status": "open"
"SyncTime": "5/16/2016 11:18:12 AM"
"UpdatingFlag": null
}
```

### 2.3.7.3 Path/Endpoint

Repository/{replIndex}/ManagementGroup/{mgname}/Events/  
{eventviewname}?parentmgpath={parentmgpath}

Repository/{replIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/  
{servername}/Events?parentmgpath={parentmgpath}

Example

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
Servers/U2VydMvYcw==/Server/SURDRFZBTtc3Ng/Events?parentmgpath=QXBwTWFuYWdlcg==
```

(where "TWFzdGVy" is base64encoded string of mgname "Master" and "U2VydMvYcw==" is base64encoded string of serverviewname "Servers" and "SURDRFZBTtc3Ng" is base64encoded string of server "server776" and "QXBwTWFuYWdlcg==" is base64encoded string of parentmgpath "AppManager")

### 2.3.7.4 Resource

Events View

## 2.3.8 Get Jobs View for a Server under Servers View

### 2.3.8.1 Description

Get Jobs View

### 2.3.8.2 Operation

Name: **GET**

Format: **JSON**

Example:

```

Example:
[2]
0:
{
  "ActualKSName": "NT"
  "Category Name": "Discovery"
  "Child Job Count": null
  "Comment": ""
  "DataSourceIntID": "3"
  "Display As": ""
  "Drop KP Object ID": "34"
  "Enterprise Job ID": null
  "Is Master Job": "False"
  "Job ID": "2"
  "Job Type": "Ad Hoc (Knowledge Script)"
  "Knowledge Script": "Discovery_NT"
  "Knowledge Script Build Number": "7.8.60.0"
  "Knowledge Script Group
Name": ""
  "Knowledge Script Version": "1.0"
  "LangID": "1"
  "Last Run Time": "3/13/2016 12:04:17 PM"
  "MachineObjID": "34"
  "Master Job ID": null
  "Modification Time": "3/13/2016 6:04:32 PM"
  "NodeID": ""
  "NodeName": ""
  "ParentJobID": "1"
  "Repository": "server776.QDB"
  "Server Name": "server776"
  "SourceID": "65536"
  "Status": "Stopped"
  "Stop Time": "3/13/2016 12:04:32 PM"
  "Submit Time": "3/13/2016 12:04:10 PM"
  "SyncTime": "5/16/2016 11:18:23 AM"
  "User": "server776\Administrator"
  "VersID": "2"
  "Virtual Job ID": null
}
1:
{
  "ActualKSName": "AMHealth"
  "Category Name": "Discovery"
  "Child Job Count": null
  "Comment": ""
  "DataSourceIntID": "3"
  "Display As": ""
  "Drop KP Object ID": "34"
  "Enterprise Job ID": null
  "Is Master Job": "False"
  "Job ID": "4"
  "Job Type": "Ad Hoc (Knowledge Script)"
  "Knowledge Script": "Discovery_AMHealth"
  "Knowledge Script Build Number": "8.0.113.0"
  "Knowledge Script Group Name": ""
  "Knowledge Script Version": "1.0"
  "LangID": "2"
  "Last Run Time": "3/13/2016 1:15:08 PM"
  "MachineObjID": "34"
  "Master Job ID": null
}

```

```

"Modification Time": "3/13/2016 7:15:13 PM"
"NodeID": ""
"NodeName": ""
"ParentJobID": "3"
"Repository": "server776.QDB"
"Server Name": "server776"
"SourceID": "65536"
"Status": "Stopped"
"Stop Time": "3/13/2016 1:15:13 PM"
"Submit Time": "3/13/2016 12:05:02 PM"
"SyncTime": "5/16/2016 11:18:23 AM"
"User": "server776\Administrator"
"VersID": "2"
"Virtual Job ID": null
}

```

### 2.3.8.3 Path/Endpoint

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/{servername}/Jobs?parentmgpath={parentmgpath}

Example

```

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/Servers/U2VydmVycw/Server/SURDRFZBTtc3Ng/Jobs?parentmgpath=QXBwTWFuYWdlcg

```

(where “TWFzdGVy” is base64encoded string of mgname “Master” and “U2VydmVycw” is base64encoded string of serverviewname “Servers” and “SURDRFZBTtc3Ng” is base64encoded string of server “server776” and “QXBwTWFuYWdlcg” is base64encoded string of parentmgpath “AppManager”)

### 2.3.8.4 Resource

Jobs View

## 2.4 Create Monitoring Policy

### 2.4.1 Description

Create monitoring policy on existing management group

### 2.4.2 Operation

Name: **POST**

Format: **JSON**

Example:

```

{
  "mgksglist" : [{"managementgroup" :
    "Master", "ksglist" : "KSG1,KSG2"}, {"managementgroup" : "Master2", "ksglist" :
    "KSG1,KSG2"}]
}

```

## 2.4.3 Path/Endpoint

/Repository/{reindex}/MonitoringPolicy

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/MonitoringPolicy`

## 2.4.4 Resource

Monitoring Policy

# 2.5 Custom Property

## 2.5.1 Create Custom Property

### 2.5.1.1 Description

Create custom property

### 2.5.1.2 Operation

Name: **POST**

Format: **JSON**

Example:

```
{
  "name": "Location",
  "type": 0,
  "description": "Geographic Location",
  "values": [{"objectlist": "server776, SQL
Server:server790, server790", "value": "y"}, {"objectlist": "hostname, SQL
Server:server793, server793", "value": "n"}]
}
```

**Type: 0 - String, 1 - Integer, 2 - Float, 3 - DateTime**

### 2.5.1.3 Path/Endpoint

/Repository/{reindex}/CustomProperty

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/CustomProperty`

### 2.5.1.4 Resource

Custom Property

## 2.5.2 Update Custom Property

### 2.5.2.1 Description

Update Custom Property Value

### 2.5.2.2 Operation

Name: **PUT**

Format: **JSON**

Example:

```
{
  "name": "Location",
  "type": 0,
  "description": "Geographic Location",
  "values": [{"objectlist": "server776, SQL Server:server790, server790", "value":
  "n"}, {"objectlist": "hostname, SQL Server:server793, server793", "value": "y"}]
}
```

---

**NOTE:** Currently CCUI allows to update both custom property definition and custom property value on computers or objects. However, in case of updating custom property definition (keeping same name but changing type), it deletes existing custom property and creates new one (with new type). The current implementation only allows to update custom property value on computers or objects. Since update of custom property definition is not supported, so currently any change of description or type is not updated.

---

### 2.5.2.3 Path/Endpoint

/Repository/{repindex}/CustomProperty

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/CustomProperty>

### 2.5.2.4 Resource

Custom Property

## 2.5.3 Delete Custom Property

### 2.5.3.1 Description

Delete Custom Property

### 2.5.3.2 Operation

Name: **DELETE**

Format: **JSON**

Example 1: Delete custom property definition

```
{
  "name": "Location",
  "type": 0,
}
```

Example 2: Delete custom property values on servers/objects

```
{
  "name": "Location",
  "type": 0,
  "values": [{"objectlist": "server776, SQL Server:server790, server534", "value": "y"}, {"objectlist": "hostname, SQL Server:server793, server793", "value": "n"}]
}
```

**Type: 0 - String, 1 - Integer, 2 - Float, 3 – DateTime**

---

**NOTE:** The DELETE input Json is same as POST except Description is not required in DELETE.

---

### 2.5.3.3 Path/Endpoint

/Repository/{repindex}/CustomProperty

Example:

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/CustomProperty

### 2.5.3.4 Resource

Custom Property

## 2.5.4 Get list of all Custom Properties

### 2.5.4.1 Description

List all custom properties

### 2.5.4.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
0: {
  name: "Location"
  description: "Geographic Location"
  type: "String"
}-
1: {
  name: "Severity"
  description: "Event Severity"
  type: "Integer"
}
```

### 2.5.4.3 Path/Endpoint

/Repository/{repindex}/CustomProperty

Example:

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/CustomProperty

### 2.5.4.4 Resource

Custom Property

## 2.6 Server Filters

Server Filters for KS view, Servers view, Events view, Jobs view, Server Information Pane Jobs view, Server Information Pane Events view

### 2.6.1 Set

#### 2.6.1.1 Description

Create Server Filters on Events, Jobs, Knowledge Scripts, Servers, Servers Events pane, Servers Jobs pane

#### 2.6.1.2 Operation

Name: **POST**

Format: **JSON**

Example 1: Create server filters on events

```
{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Severity",
      "condition": "Between",
      "value": "5",
      "value2": "15"
    },
    {
      "fieldname": "Has Comment",
      "condition": "Equal To",
      "value": "Yes"
    }
  ]
}
```

Example 2: Create server filter on jobs

```

{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Knowledge Script",
      "condition": "Not Empty",
      "value": ""
    },
    {
      "fieldname": "Has Comment",
      "condition": "Equal To",
      "value": "Yes"
    }
  ]
}

```

#### Example 3: Create server filter on Knowledge Scripts

```

{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Full Name",
      "condition": "Equal To",
      "value": "NT_CpuLoaded"
    },
    {
      "fieldname": "Has Comment",
      "condition": "Equal To",
      "value": "Yes"
    }
  ]
}

```

#### Example 4: Create server filter on servers

```

{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Server Name",
      "condition": "Equal To",
      "value": "server930"
    },
    {
      "fieldname": "Has Comment",
      "condition": "Equal To",
      "value": "Yes"
    }
  ]
}

```

---

**NOTE:** fieldname as displayed in CCUI Server Filters for a view.

---



### 2.6.1.3 Path/Endpoint

---

**NOTE:** Below Endpoints (related to Server Filters, Events View, Jobs View, Knowledge Scripts View, Servers View) expect Base64Encoded string in dynamic parameters in URI. The Base64Encoded dynamic parameters are italicized.

---

Repository/{repIndex}/ManagementGroup/{mgname}/Events/{eventviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Jobs/{jobviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/KnowledgeScripts/{ksviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{servviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{servviewname}/Server/  
{servname}/Events/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{servviewname}/Server/  
{servname}/Jobs/ServerFilters

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
Events/RXZlbnRz/ServerFilters
```

(where “TWFzdGVy” is base64encoded string of mgname “Master” and “RXZlbnRz” is base64encoded string of eventviewname “Events”)

### 2.6.1.4 Resource

Server Filters

## 2.6.2 Get

### 2.6.2.1 Description

Get Server Filters on Events, Jobs, Knowledge Scripts, Servers, Servers Events pane, Servers Jobs pane

### 2.6.2.2 Operation

Name: **GET**

Format: **JSON**

Example 1: Get server filters on events

```
[
  {
    "condition": "Between",
    "fieldname": "Severity",
    "value": "5",
    "value2": "15"
  },
  {
    "condition": "Equal To",
    "fieldname": "Has Comment",
    "value": "Yes",
    "value2": null
  }
]
```

#### Example 2: Get server filters on jobs

```
[
  {
    "condition": "Not Empty",
    "fieldname": "Knowledge Script",
    "value": "",
    "value2": null
  },
  {
    "condition": "Equal To",
    "fieldname": "Has Comment",
    "value": "Yes",
    "value2": null
  }
]
```

#### Example 3: Get server filters on Knowledge Scripts

```
[
  {
    "condition": "Equal To",
    "fieldname": "Full Name",
    "value": "NT_CpuLoaded",
    "value2": null
  }
]
```

#### Example 4: Get server filters on servers

```
[
  {
    "condition": "Equal To",
    "fieldname": "Server Name",
    "value": "server930",
    "value2": null
  }
]
```

### 2.6.2.3 Path/Endpoint

---

**NOTE:** Below Endpoints (related to Server Filters, Events View, Jobs View, Knowledge Scripts View, Servers View) expect Base64Encoded string in dynamic parameters in URI. The Base64Encoded dynamic parameters are italicized.

---

Repository/{repIndex}/ManagementGroup/{mgname}/Events/{eventviewname}/  
ServerFilters?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/Jobs/{jobviewname}/  
ServerFilters?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/KnowledgeScripts/{ksviewname}/  
ServerFilters?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/  
ServerFilters?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/  
{servername}/Events/ServerFilters?parentmgpath={parentmgpath}

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/  
{servername}/Jobs/ServerFilters?parentmgpath={parentmgpath}

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/  
Events/RXZlbnRz/ServerFilters?parentmgpath=QXBwTWFuYWdlcg
```

(where "TWFzdGVy" is base64encoded string of mgname "Master" and "RXZlbnRz" is  
base64encoded string of eventviewname "Events" and "QXBwTWFuYWdlcg" is base64encoded  
string of parentmgpath "AppManager")

## 2.6.2.4 Resource

Server Filters

## 2.6.3 Delete

### 2.6.3.1 Description

Delete Server Filters on Events, Jobs, Knowledge Scripts, Servers, Servers Events pane, Servers  
Jobs pane

### 2.6.3.2 Operation

Name: **DELETE**

Format: **JSON**

Example 1: Json input (to remove all filters - serverfilters tag can be removed or passed null value):

```
{  
  "parentmgpath": "AppManager\\Master",  
  "serverfilters": null  
}
```

Example 2:

```

{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Severity",
      "condition": "Between",
      "value": "5",
      "value2": "15"
    },
    {
      "fieldname": "Has Comment",
      "condition": "Equal To",
      "value": "Yes"
    }
  ]
}

```

Example 3:

```

{
  "parentmgpath": "AppManager\\Master",
  "serverfilters": [
    {
      "fieldname": "Full Name",
      "condition": "Equal To",
      "value": "NT_CpuLoaded"
    }
  ]
}

```

### 2.6.3.3 Path/Endpoint

---

**NOTE:** Below Endpoints (related to Server Filters, Events View, Jobs View, Knowledge Scripts View, Servers View) expect Base64Encoded string in dynamic parameters in URI. The Base64Encoded dynamic parameters are italicized.

---

Repository/{repIndex}/ManagementGroup/{mgname}/Events/{eventviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Jobs/{jobviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/KnowledgeScripts/{ksviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/{servename}/Events/ServerFilters

Repository/{repIndex}/ManagementGroup/{mgname}/Servers/{serverviewname}/Server/{servename}/Jobs/ServerFilters

Example:

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ManagementGroup/TWFzdGVy/Events/RXZlbnRz/ServerFilters

(where "TWFzdGVy" is base64encoded string of mgname "Master" and "RXZlbnRz" is base64encoded string of eventviewname "Events")

## 2.6.3.4 Resource

Server Filters

# 2.7 Deployment

## 2.7.1 Create deployment rule

### 2.7.1.1 Description

Create Deployment Rule

### 2.7.1.2 Operation

Name: **POST**

Format: **JSON**

Example 1: Deploying agent

```
{
  "name": "InstallAgent",
  "description": "NewAgent",
  "packagesdata": {
    "uninstall": 0,
    "skipprecheck": 0,
    "packages": [
      {
        "packagename": "WINDOWSAGENT",
        "version": "9.2.0.569",
        "configinfo": {
          "mc_b_upgrade": "Install",
          "installdir": "C:/Program Files/NetIQ",
          "mc_b_reportagent": "y",
          "mc_seclevel": "Unencrypted",
          "ms_port": "9999",
          "mc_port": "9998",
          "mc_b_fqdn": "n",
          "mc_windomain": "amdom009",
          "mc_winuser": "Apullanthole",
          "mc_winpwd": "Controll123",
          "mc_msprimary": "server550",
          "mc_mssecondary": "",
          "prm_designatemss": "Allow anonymous MS until Primary/Secondary MS is set"
        }
      }
    ]
  },
  "credentials": {
    "username": "Apullanthole",
    "password": "Controll123",
    "domain": "amdom009"
  }
}
```

```

},
"targetcomputers": {
"targettype": "SpecificComputers",
"targetname": "server551",
"inclusionfilters": null,
"exclusionfilters": null
},
"deploymentservice": "server550",
"deploymentschedule": null,
"deploymentnotification": null
}

```

### Example 2: Upgrading agent

```

{
"name": "UpgradeAgent",
"description": "Upgrade the Agent",
"packagesdata": {
"uninstall": 0,
"skipprecheck": 0,
"packages": [
{
"packagename": "WINDOWSAGENT",
"version": "9.2.0.569",
"configinfo": {
"mc_b_upgrade": "Upgrade"
}
}
]
},
"credentials": {
"username": "Apullanthole",
"password": "Control123",
"domain": "amdom009"
},
"targetcomputers": {
"targettype": "SpecificComputers",
"targetname": "server551",
"inclusionfilters": null,
"exclusionfilters": null
},
"deploymentservice": "Server550",
"deploymentschedule": null,
"deploymentnotification": null
}

```

### Example 3: Deploying module

```

{
  "name": "SQL module",
  "description": "Install the SQL module",
  "packagesdata": {
    "uninstall": 0,
    "skipprecheck": 0,
    "packages": [
      {
        "packagename": "SQL",
        "version": "7.5.17.0",
        "configinfo": {}
      }
    ]
  },
  "credentials": {
    "username": "Apullanthole",
    "password": "Controll123",
    "domain": "amdom009"
  },
  "targetcomputers": {
    "targettype": "SpecificComputers",
    "targetname": "Server551",
    "inclusionfilters": null,
    "exclusionfilters": null
  },
  "deploymentservice": "Server550",
  "deploymentschedule": null,
  "deploymentnotification": null
}

```

#### Example 4: Deploying agent and module

```

{
  "name": "windows Agent and SQL module",
  "description": "Install the SQL module and agent module",
  "packagesdata": {
    "uninstall": 0,
    "skipprecheck": 0,
    "packages": [
      {
        "packagename": "SQL",
        "version": "7.5.17.0",
        "configinfo": { }
      },
      {
        "packagename": "WINDOWSAGENT",
        "version": "9.2.0.569",
        "configinfo": {
          "mc_b_upgrade": "Install",
          "installdir": "C:/Program Files/NetIQ",
          "mc_b_reportagent": "y",
          "mc_seclevel": "Unencrypted",
          "ms_port": "9999",
          "mc_port": "9998",
          "mc_b_fqdn": "n",
          "mc_windomain": "amdom009",
          "mc_winuser": "Apullanthole",
          "mc_winpwd": "Controll123",
          "mc_msprimary": "server550",
          "mc_mssecondary": "",
        }
      }
    ]
  }
}

```

```

MS is set"
        "prm_designatemss": "Allow anonymous MS until Primary/Secondary
    }
}
]
},
"credentials": {
    "username": "Apullanthole",
    "password": "Control123",
    "domain": "amdom009"
},
"targetcomputers": {
    "targettype": "SpecificComputers",
    "targetname": "server551",
    "inclusionfilters": null,
    "exclusionfilters": null
},
"deploymentsservice": "server550",
"deploymentsschedule": null,
"deploymentnotification": null
}

```

---

**NOTE:**

To know more about package mapping, see [Section 2.11, "Remote Deployment Package Mapping," on page 59](#)

- "uninstall": 0 or tag is not present => Install action.

"uninstall": 1 => packages to be uninstalled.

- "skipprecheck": 0 or tag is not present => do not skip precheck. "skipprecheck": 1 => skip pre check.

- "version" tag is not present or empty => It will find package of latest version and use that version.

- "configinfo" required for agent and couple of modules else we can remove tag (or pass null).

The agent configuration parameters to be considered for "configinfo":

"mc\_b\_upgrade" : "Install"

"installdir" : "C:\Program Files\NetIQ"

"mc\_b\_reportagent" : "y"

"mc\_seclevel" : "Authentication and encrypted communications"

"mc\_secpwd" : "compaq1-2"

"ms\_port" : "9999"

"mc\_port" : "9998"

"mc\_b\_fqdn" : "n"

"mc\_windomain" : "testdomain"

"mc\_winuser" : "testuser"



"mc\_winpwd" : "compaq1-2"

"mc\_msprimary" :

"testprimaryms"

"mc\_mssecondary" :

"testsecondaryms"

"prm\_designatemss" : "Allow anonymous MS until Primary/Secondary MS is set"

"mc\_ad\_upgrade" : "Existing Values on the Target Computer"

The CIM module configuration parameters to be considered for "configinfo": "community string" : ""

The NETFINITYDIR and SIEMENS modules configuration parameters to be considered for "configinfo": "community name" : ""

The EXCHANGE2000 module configuration parameters to be considered for "configinfo":

"p\_server0" :

"" "p\_mname0" :

"" "p\_profile\_name0" :

"" "p\_domain0" :

"" "p\_user0" :

"" "p\_pwde0" : ""

The WIN-RT7 module configuration parameters to be considered for "configinfo":

"domain" : ""

"user name" : ""

"password" : ""

- "inclusionfilters" tag is not present => no inclusion filters

- "exclusionfilters" tag is not present => no exclusion filters

- "deploymentsservice" tag is must

- Currently only two target type (SpecificComputers and Agent are supported). In case of targettype "Agent", targetname can be null or empty. e.g. "targettype": "Agent", "targetname": null

In case of Rule already exists, Rest will return code 2.

---

### 2.7.1.3 Path/Endpoint

/Repository/{repindex}/Deployment/Rule

Example:

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/  
Yw1bnRJbnN0

## 2.7.1.4 Resource

Remote Deployment

## 2.7.2 Get deployment rule

### 2.7.2.1 Description

Get Deployment Rule

### 2.7.2.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{ "credentials":null,"deploymentnotification":null,"deploymentschedule":null,"deploymentsservice":"server793","description":"","name":"testrule","packagesdata":{"packages":[{"configinfo":{"community name":null,"community string":null,"domain":null,"installdir":"C:\\Program Files\\NetIQ","mc_ad_upgrade":"Existing Values on the Target Computer","mc_b_fqdn":"n","mc_b_reportagent":"y","mc_b_upgrade":"Install","mc_b_wi nuser":"y","mc_msprimary":"server793","mc_mssecondary":"","mc_port":"9998","mc_sec level":"Unencrypted","mc_secpwd":"","mc_windomain":"server793","mc_winpwd":"","mc_w inuser":"Administrator","ms_port":"9999","p_domain0":null,"p_mname0":null,"p_pro file_name0":null,"p_pwde0":null,"p_server0":null,"p_user0":null,"password":null,"p rm_designatemss":"Never allow anonymous MS","user name":null},"packagename":"WINDOWSAGENT","packageunique name":"NetIQ_AppManager_Win dowsAgent__","version":"9.2.0.493"}]},"skipprecheck":0,"uninstall":0},"status":"Ena bled","statusdetails":"<Status>Processing Rule: testrule </Status><Status> Input Package(s):</Status><Status> - NetIQ AppManager Windows Agent</Status><Status> Input Machine(s):</Status><Status> - server793</Status><Stat Get Deployment Rule us>Processing Result:</Status> <Status> Machine: server793</Status><Status> Package: NetIQ AppManager Windows Agent</Status><Status> Result: Failed</ Status><Status> Comment: Install agent task not created (agent already installed).</Status><Status> -----</Status>","targetcomputers":null}
```

### 2.7.2.3 Path/Endpoint

/Repository/{replIndex}/Deployment/Rule/{ruleName}

Where ruleName is base64 encoded string

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/YWdlbnRJbnNO
```

### 2.7.2.4 Resource

Remote Deployment

## 2.7.3 Get Rule History (for AppManager 9.2 and later)

### 2.7.3.1 Description

Get Deployment Rule History

### 2.7.3.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{ "name": "testrule", "ruledetails": [ { "dateprocessed": "8/7/2017 9:04:24 PM", "machinedetails": [ { "agentpackage": "No", "machinename": "server793", "successful": "No", "taskcomment": "Install module for package AppManager for SQL Server Get Deployment Rule History task not created (software inventory detected the module for this package for computer with AuthoritativeID 57273A56-CF0D-45F9-91E7-B430C0BD5359 is the same as or higher than the version for the package)", "uninstall": "No" } ], "packagename": "AppManager for SQL Server", "packageversion": "7.5.17.0", "servicename": "server793", "username": "WORKGROU P\\Administrator" }, { "dateprocessed": "8/22/2017 9:01:28 AM", "machinedetails": [ { "agentpackage": "Yes", "machinename": "server793", "successful": "No", "taskcomment": "Install agent task not created (agent already installed).", "uninstall": "No" } ], "packagename": "NetIQ AppManager Windows Agent", "packageversion": "9.2.0.493", "servicename": "server793", "username": "WORKGROU P\\Administrator" } ] }
```

### 2.7.3.3 Event/Resource

/Repository/{repIndex}/Deployment/Rule/{ruleName}/History

Where ruleName is base64 encoded string

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/dGVzdHJlbGU=/History
```

### 2.7.3.4 Resource

Remote Deployment

## 2.7.4 Delete deployment Rule

### 2.7.4.1 Description

Delete Deployment Rule

### 2.7.4.2 Operation

Name: **DELETE**

### 2.7.4.3 Event/Endpoint

/Repository/{repIndex}/Deployment/Rule/{ruleName}

Where ruleName is base64 encoded string

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/dGVzdHJ1bGU=
```

### 2.7.4.4 Resource

Remote Deployment

## 2.7.5 Enable deployment Rule

### 2.7.5.1 Description

Enable Deployment Rule

### 2.7.5.2 Operation

Name: **PUT**

Format: **JSON** (pass empty tags like "{}" else it may throw error "the remote server returned an error (411) length required")

### 2.7.5.3 Path/Endpoint

/Repository/{repIndex}/Deployment/Rule/{ruleName}?command=enable

Where ruleName is base64 encoded string

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/dGVzdHJ1bGU=?command=enable
```

### 2.7.5.4 Resource

Remote Deployment

## 2.7.6 Disable deployment Rule

### 2.7.6.1 Description

Disable Deployment Rule

## 2.7.6.2 Operation

Name: **PUT**

Format: **JSON** (pass empty tags like "{}" else it may throw error "the remote server returned an error (411) length required")

## 2.7.6.3 Path/Endpoint

/Repository/{repIndex}/Deployment/Rule/{ruleName}}?command=disable

Where ruleName is base64 encoded string

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/dGVzdHJ1bGU=?command=disable
```

## 2.7.6.4 Resource

Remote Deployment

## 2.7.7 Get list of deployment rules

### 2.7.7.1 Description

Get all Deployment Rules

### 2.7.7.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{ "credentials": null, "deploymentnotification": null, "deploymentschedule": null, "deploymentsservice": "server793", "description": "", "name": "testrule", "packagesdata": { "packages": [ { "configinfo": { "community name": null, "community string": null, "domain": null, "installdir": "C:\\Program Files\\NetIQ", "mc_ad_upgrade": "Existing Values on the Target Computer", "mc_b_fgdn": "n", "mc_b_reportagent": "y", "mc_b_upgrade": "Install", "mc_b_w_inuser": "y", "mc_msprimary": "server793", "mc_mssecondary": "", "mc_port": "9998", "mc_seclevel": "Unencrypted", "mc_secpwd": "", "mc_windomain": "server793", "mc_winpwd": "", "mc_winuser": "Administrator", "ms_port": "9999", "p_domain0": null, "p_mbname0": null, "p_profile_name0": null, "p_pwd0": null, "p_server0": null, "p_user0": null, "password": null, "prm_designatemss": "Never allow anonymous MS", "user name": null }, "packagename": "WINDOWSAGENT", "packageunique name": "NetIQ_AppManager_WindowsAgent__", "version": "9.2.0.493" } ], "skipprecheck": 0, "uninstall": 0 }, "status": "Enabled", "statusdetails": "<Status>Processing Rule: testrule </Status><Status> Input Package(s):</Status><Status> - NetIQ AppManager Windows Agent</Status><Status> Input Machine(s):</Status><Status> - server793</Status><Status>Processing Result:</Status> <Status> Machine: server793</Status><Status> Package: NetIQ AppManager Windows Agent</Status><Status> Result: Failed</Status><Status> Comment: Install agent task not created (agent already installed).</Status><Status> -----</Status>", "targetcomputers": null }
```

### 2.7.7.3 Path/Endpoint

/Repository/{repIndex}/Deployment/Rule

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule`

### 2.7.7.4 Resource

Remote Deployment

## 2.7.8 Get single task for deployment rule

### 2.7.8.1 Description

Get single Task details

### 2.7.8.2 Operation

Name: **GET**

Format: **JSON**

Example:

The returned Json data is same like returned by endpoint

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Rule/{rulename}/Task`

### 2.7.8.3 Path/Endpoint

/Repository/{repindex}/Deployment/Task/{taskuniqueName}

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Task/EA7920D0-F596-4FB9-B01F-DFC2EECD01CA`

### 2.7.8.4 Resource

Remote Deployment

## 2.7.9 Get all tasks for deployment rule

### 2.7.9.1 Description

Get Software Inventory (Note: This will return s/w inventory for all machines with all applications details)

## 2.7.9.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[
4]
0:
{
  "applicationname": "AD-RT"
  "applicationversion": "7.1.195.0"
  "computer": "server534"
  "lastmodified": "12/16/2015 5:12:56 AM"
  "platform": "Windows"
}
-
1:
{
  "applicationname": "AD-RT"
  "applicationversion": "7.1.195.0"
  "computer": "server776"
  "lastmodified": "4/4/2016 7:17:27 AM"
  "platform": "Windows"
}
-
2:
{
  "applicationname": "AMWindowsOS.dll"
  "applicationversion": "8.0.80.0"
  "computer": "server937"
  "lastmodified": "3/9/2016 11:00:26 AM"
  "platform": "Windows"
}
-
3:
{
  "applicationname": "ASync"
  "applicationversion": "7.8.60.0"
  "computer": "server534"
  "lastmodified": "12/21/2015 6:24:59 AM"
  "platform": "Windows"
}
```

## 2.7.9.3 Event/Endpoint

/Repository/{repindex}/Deployment/Inventory

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Inventory>

## 2.7.9.4 Resource

Remote Deployment

## 2.7.10 Get Task status details

### 2.7.10.1 Description

Get a particular task status details

### 2.7.10.2 Operation

Name: **GET**

Format: **JSON**

```
{
  "status": "Completed"
  "statusdetails": "The task completed successfully."
}
```

### 2.7.10.3 Event/Endpoint

/Repository/{repindex}/Deployment/Task/{taskuniqueid}/Status

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/ Deployment/Task/
EA7920D0-F596-4FB9-B01F-DFC2EECD01CA/Status
```

### 2.7.10.4 Resource

Remote Deployment

## 2.7.11 Update Task status (i.e. Approve Task)

### 2.7.11.1 Description

Approve a particular task

### 2.7.11.2 Operation

Name: **PUT**

Format: **JSON** (pass empty tags like "{}" else it may throw error "the remote server returned an error (411) length required")

---

**NOTE:** only "approve" command is supported as of now.

---

### 2.7.11.3 Event/Endpoint

/Repository/{repindex}/Deployment/Task/{taskuniqueid}?command={commandname}

Example:



<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Task/EA7920D0-F596-4FB9-B01F-DFC2EECD01CA?command=approve>

## 2.7.11.4 Resource

Remote Deployment

## 2.7.12 Get Software Inventory

### 2.7.12.1 Description

Get Software Inventory

---

**NOTE:** This will return s/w inventory for all machines with all applications details

---

### 2.7.12.2 Operation

Name: **GET**

Format: **JSON**

```
[
  4]
  0:
  {
    "applicationname": "AD-RT"
    "applicationversion": "7.1.195.0"
    "computer": "server534"
    "lastmodified": "12/16/2015 5:12:56 AM"
    "platform": "Windows"
  }
  -
  1:
  {
    "applicationname": "AD-RT"
    "applicationversion": "7.1.195.0"
    "computer": "server776"
    "lastmodified": "4/4/2016 7:17:27 AM"
    "platform": "Windows"
  }
  -
  2:
  {
    "applicationname": "AMWindowsOS.dll"
```

```

"applicationversion": "8.0.80.0"
"computer": "server937"
"lastmodified": "3/9/2016 11:00:26 AM"
"platform": "Windows"
}
-
3:
{
"applicationname": "ASYNC"
"applicationversion": "7.8.60.0"
"computer": "server534"
"lastmodified": "12/21/2015 6:24:59 AM"
"platform": "Windows"
}

```

### 2.7.12.3 Event/Endpoint

/Repository/{repindex}/Deployment/Inventory

Example:

http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Deployment/Inventory

### 2.7.12.4 Resource

Remote Deployment

## 2.8 Knowledge Script

### 2.8.1 Copy KS/KSG

#### 2.8.1.1 Description

Create Copy KS/KSG

#### 2.8.1.2 Operation

Name: **POST**

Format: **JSON**

Example (KS):

```
{ "copyksname" : "NT_Copy123", "description" : "", "isksg": 0 }
```

:Example (KSG):

```
{ "copyksname" : "KSG_TestNT1", "description" : "", "isksg": 1 }
```

#### 2.8.1.3 Path/Endpoint

Repository/{repIndex}/KnowledgeScript/{ksname}?command=copy

Example (KS):

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
NT_CpuLoaded?command=copy`

Example (KSG):

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
KSG_TestNT?command=copy`

#### **2.8.1.4 Resource**

Knowledge Script

### **2.8.2 Delete KS/KSG**

#### **2.8.2.1 Description**

Delete KS/KSG

#### **2.8.2.2 Operation**

Name: **DELETE**

#### **2.8.2.3 Path/Endpoint**

`Repository/{repIndex}/KnowledgeScript/{ksname}`

Example (KS):

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
NT_CpuLoaded`

Example (KSG):

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
KSG_TestNT`

#### **2.8.2.4 Resource**

Knowledge Script

### **2.8.3 Propagate - All knowledge scripts to Derived Knowledge Scripts (for AppManager 9.2 and later)**

#### **2.8.3.1 Description**

Propagate all Knowledge Scripts to Derived KSs

#### **2.8.3.2 Operation**

Name: **POST**

Format: **JSON** (pass empty tags like "{}" else it may throw error "the remote server returned an error

(411) length required")

### 2.8.3.3 Path/Endpoint

Repository/{repIndex}/ KnowledgeScript/Propagation/DerivedKnowledgeScripts

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
Propagation/DerivedKnowledgeScripts
```

### 2.8.3.4 Resource

Knowledge Script

## 2.8.4 Propagate - Selected Knowledge Scripts to Derived Knowledge Scripts (for AppManager 9.2 and later)

### 2.8.4.1 Description

Propagate selected Knowledge Scripts to Derived KSs

### 2.8.4.2 Operation

Name: **POST**

### 2.8.4.3 Path/Endpoint

Repository/{repIndex}/KnowledgeScript/Propagation/DerivedKnowledgeScripts/  
{knowledgescriptsname}

where knowledgescriptsname is base64 encoding of comma separated ks list (like base 64 of "NT\_CpuLoaded, NT\_MemUtil, NT\_CpuResource")

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/  
Propagation/DerivedKnowledgeScripts/  
TlRfQ3B1TG9hZGVkLCBOVF9NZW1vdGlsLCBOVF9DcHVSZXNvdXJjZQ==
```

### 2.8.4.4 Resource

Knowledge Script

## 2.8.5 Propagate - All Knowledge Scripts to Ad Hoc Jobs (for AppManager 9.2 and later)

### 2.8.5.1 Description

Propagate KSs to their AdHoc Jobs

## 2.8.5.2 Operation

Name: **POST**

Format: **JSON** (pass empty tags like "{}" else it may throw error "the remote server returned an error (411) length required")

## 2.8.5.3 Path/Endpoint

Repository/{repIndex}/KnowledgeScript/Propagation/AdHocJobs

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/Propagation/AdHocJobs
```

## 2.8.5.4 Resource

Knowledge Script

# 2.8.6 Propagate - Selected Knowledge Scripts to Ad Hoc Jobs (for AppManager 9.2 and later)

## 2.8.6.1 Description

Propagate selected KSs to their AdHoc Jobs

## 2.8.6.2 Operation

Name: **POST**

Format: **JSON**

Example:

```
{
  "kslist": [
    {
      "datasourcename": "Server930.QDB92",
      "ksname": "NT_CpuLoaded"
    },
    {
      "datasourcename": "Server930\\instance.QDB",
      "ksname": "NT_CpuResource"
    }
  ]
}
```

## 2.8.6.3 Path/Endpoint

Repository/{repIndex}/KnowledgeScript/Propagation/AdHocJobs

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/Propagation/AdHocJobs>

## 2.8.6.4 Resource

Knowledge Script

## 2.8.7 Propagate - Get KS list pending propagation to their Derived KSs (for AppManager 9.2 and later)

### 2.8.7.1 Description

Get KS list pending propagation to their Derived KSs

### 2.8.7.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[{"baseksbuildid":"8.1.82.0","baseksname":"NT_CpuLoaded","basekssourceid":"1.4","derivedbuildid":"8.1.82.0","derivedksid":"2297","derivedksname":"KSG_TestKSG:NT_CpuLoaded","derivedsourceid":"1.3","policiesaffected":0,"policyjobsaffected":0,"scriptupdated":0}, {"baseksbuildid":"8.1.82.0","baseksname":"NT_CpuLoaded","basekssourceid":"1.4","derivedbuildid":"8.1.82.0","derivedksid":"2059","derivedksname":"NT_BestPractices:NT_CpuLoaded","derivedsourceid":"1.3","policiesaffected":0,"policyjobsaffected":0,"scriptupdated":0}]
```

### 2.8.7.3 Path/Endpoint

Repository/{replIndex}/KnowledgeScript/Propagation/DerivedKnowledgeScripts

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/Propagation/DerivedKnowledgeScripts>

## 2.8.7.4 Resource

Knowledge Script

## 2.8.8 Propagate - Get KS list pending propagation to their Ad Hoc Jobs (for AppManager 9.2 and later)

### 2.8.8.1 Description

Get KS list pending propagation to their AdHoc Jobs

## 2.8.8.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[{"childjobcount":1,"datasourcename":"server793.QDB","jobbuildid":"8.1.94.0","jobsourcename":"1.3","ksbuildid":"8.1.94.0","ksname":"NT_DiskSpace","kssourceid":"1.4","parentjobid":"1710"}]
```

## 2.8.8.3 Path/Endpoint

Repository/{repIndex}/KnowledgeScript/Propagation/AdHocJobs

Example:

```
http://hostname/NQAppManagerAPI/Service.svc/Repository/0/KnowledgeScript/Propagation/AdHocJobs
```

## 2.8.8.4 Resource

Knowledge Script

# 2.9 Events

## 2.9.1 Update bulk events status to Acknowledge/Close/Delete (apply filter on Computer, Job ID, KS Name, Event ID)

### 2.9.1.1 Description

Perform bulk Event Operation

### 2.9.1.2 Operation

Name: **PUT**

Format: **JSON**

Example:

```
{
  "filtertype" : "",
  "filtervalue" : "",
  "newstatus": ""
}
```

where filtertype can be among severity, computer, jobid, eventid and newstatus can be among acknowledge, close, delete.

Example 1:

```
{
  "filtertype" : "severity",
  "filtervalue" : "warning",
  "newstatus": "acknowledge"
}
```

Example 2:

```
{
  "filtertype" : "computer",
  "filtervalue" : "server795",
  "newstatus": "close"
}
```

Example 3:

```
{
  "filtertype" : "jobid",
  "filtervalue" : "21",
  "newstatus": "acknowledge"
}
```

Example 4:

```
{
  "filtertype" : "eventid",
  "filtervalue" : "102",
  "newstatus": "delete"
}
```

### 2.9.1.3 Path/Endpoint

/Repository/{repindex}/Event

### 2.9.1.4 Resource

Event

## 2.9.2 Update Event status (Acknowledge/Close/Delete)

### 2.9.2.1 Description

Perform operation on particular event

### 2.9.2.2 Operation

Name: PUT

### 2.9.2.3 Path/Endpoint

/Repository/{repindex}/Event/{eventId}?status=close

/Repository/{repindex}/Event/{eventId}?status=acknowledge

/Repository/{repindex}/Event/{eventId}?status=delete



Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Event?status=close`

## 2.9.2.4 Resource

Event

## 2.9.3 Get single event details

### 2.9.3.1 Description

Get particular event details

### 2.9.3.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
{ "comment": null, "computer": "server791", "count": 1, "datasourcename": "server793.QDB",  
  "eventid": 395, "eventmsg": "AppManager: 2 Jobs taking longer than normal to  
  execute", "jobid": 33, "ksname": "AMHealth_HeartbeatWin", "parenteventid": 143, "severity  
  ": "severe", "status": "open" }
```

### 2.9.3.3 Path/Endpoint

`/Repository/{repindex}/Event/{eventId}`

Example:

`http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Event/102`

### 2.9.3.4 Resource

Event

## 2.9.4 Get all events detail

### 2.9.4.1 Description

Get all events details

### 2.9.4.2 Operation

Name: **GET**

Format: **JSON**

Example:

```
[{"comment":null,"computer":"server793","count":2,"datasourcename":"server793.QDB",
"eventid":1,"eventmsg":"Discovery failed for Call Performance:server793
.", "jobid":14,"ksname":"Discovery_VoIPQuality_CallPerf", "parenteventid":0, "severity":
"severe", "status":"open"}, {"comment":null,"computer":"server793","count":1,"dat
asourcename":"server793.QDB", "eventid":2, "eventmsg":"Management Server: server793
failed to process discovery data for agent
", "jobid":14, "ksname":"Discovery_VoIPQuality_CallPerf", "parenteventid":1, "severity
":"severe", "status":"open"}]
```

### 2.9.4.3 Path/Endpoint

/Repository/{repindex}/Event

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Event>

### 2.9.4.4 Resource

Event

## 2.10 Control Center Console Global option

### 2.10.1 To set default deployment rule credentials

#### 2.10.1.1 Description

Set deployment default credentials

#### 2.10.1.2 Operation

Name: **POST**

Format: **JSON**

Example:

```
{
  "username" : "Administrator",
  "password" : "Controll123",
  "domain": "WORKGROUP"
}
```

#### 2.10.1.3 Path/Endpoint

/Repository/{repindex}/Options/Deployment/Credentials

Example:

<http://hostname/NQAppManagerAPI/Service.svc/Repository/0/Options/Deployment/Credentials>

## 2.10.1.4 Resource

Remote Deployment

## 2.11 Remote Deployment Package Mapping

To map packages for remote deployment, follow the table below:

Module Name	Package Name Mapping
NetIQ AppManager Windows Agent	WINDOWSAGENT
AppManager for Microsoft Windows	WINOS
AppManager for Active Directory	AD
AppManager ResponseTime for Active Directory	AD-RT
AppManager for Agentless Monitoring	AGENTLESS
NetIQ AppManager Self Monitoring	AMHEALTH
AppManager for Apache (UNIX)	APACHE
AppManager for Avaya Communication Manager	AVAYACM
AppManager for BlackBerry Enterprise Server	BES
AppManager for Call Data Analysis	CALLDATAANALYSIS
AppManager for HP Compaq Insight Manager	CIM
AppManager for Cisco CallManager	CISCOALLMGR
AppManager for Cisco Unified CallManager	CISCOCM
AppManager for Cisco Integrated Contact Distribution	CISCOICD
AppManager for Cisco Intelligent Contact Management	CISCOICM
AppManager for Cisco Unity Connection	CISCOUC
AppManager for Cisco Unified CallManager	CISCOUCM
AppManager for Cisco Unity Express	CISCOUE
AppManager for Cisco Unity	CISCOUNITY
AppManager for Dell OpenManage	DELL
AppManager for Lotus Domino	DOMINO
AppManager ResponseTime for Exchange and Exchange Online	EXCHANGE-RT
AppManager for Microsoft Exchange Server 2000/2003	EXCHANGE2000
AppManager for Microsoft Exchange Server and Exchange Online	EXCHANGE2007
AppManager for Hardware	HARDWARE

<b>Module Name</b>	<b>Package Name Mapping</b>
AppManager for Hyper-V Monitoring	HYPER-V
AppManager for Microsoft Internet Information Services	IIS
AppManager for JAVA Monitoring	JAVA
AppManager for Microsoft Lync	LYNC
AppManager for Citrix MetaFrame	MFXP
Module Builder	MODULEBUILDER
AppManager for MSCS	MSCS
AppManager for Symantec NetBackup	NETBACKUP
AppManager for NetBackup UNIX	NETBACKUPUNIX
AppManager for IBM Netfinity Director	NETFINITYDIR
AppManager for Network Device	NETWORKDEVICE
AppManager ResponseTime for Networks	NETWORKS-RT
AppManager for Nortel Contact Center	NORTELCC
AppManager for Nortel Communications Server	NORTE LCS
AppManager for Nortel Communication Server 2000/2100	NORTE LCS2X
AppManager for Office Communications Server	OCS
AppManager for Oracle Database	ORACLE
AppManager ResponseTime for Oracle	ORACLE-RT
AppManager for Oracle RDBMS (UNIX/Linux)	ORACLEUNIX
AppManager for IP Phone Quality	PHONEQUALITY
AppManager for IBM PowerVM	POWERVM
AppManager for Microsoft SharePoint Server	SHAREPOINT
AppManager for Siemens ServerView	SIEMENS
AppManager for Session Initiation Protocol Server	SIPSERVER
AppManager for SNMP	SNMP
AppManager for SNMP Traps	SNMPTRAPS
AppManager for Oracle Solaris Zones	SOLARISZONES
AppManager for SQL Server	SQL
AppManager ResponseTime for SQL	SQL-RT
AppManager for Microsoft SQL Server	SQLSERVER
AppManager for UNIX	UNIX
AppManager for VMware	VMWARE

<b>Module Name</b>	<b>Package Name Mapping</b>
AppManager for VoIP Quality	VOIPQUALITY
AppManager for WebLogic Server UNIX	WEBLOGICSVRUNIX
AppManager ResponseTime for Web	WEB-RT
AppManager for WebSphere Application Server UNIX	WEBSHEREAPPSRVUNIX
AppManager for IBM WebSphere MQ for UNIX	WEBSPHEREMQUNIX
AppManager ResponseTime 7 for Windows	WIN-RT7
AppManager for WMI	WMI
AppManager for Windows Terminal Server	WTS
AppManager for Citrix XenApp Server	XENAPP
AppManager for Citrix XenDesktop and XenApp	XENDESKTOP

