

Directory and Resource Administrator

Product Customization Reference

July 2023

Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see <https://www.microfocus.com/en-us/legal>.

© Copyright 2007-2025 Open Text or one of its affiliates.

The only warranties for products and services of Open Text and its affiliates and licensors ("Open Text") are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Open Text shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Contents

1	About this Guide	5
2	Product Customization Scenario Details.....	6
	Active Directory Configuration Script:	6
	User Personas	6
3	Customizing NRC Reports	8
	Create a User Defined (UD) Context for report filtering	9
	Scope	9
	Step-by-step Guide	9
	Creating a "Collections Status for the last seven days" Report.....	14
	Scope	14
	Step-by-step Guide	14
	Creating a "Temporary Group Assignment" Report.....	17
	Scope	17
	Step-by-step Guide	17
	Creating a "Virtual Attribute Associations" Report	19
	Scope	19
	Step-by-step Guide	19
	Modify an Existing Report, SQL column addition	22
	Scope	22
	Step-by-step Guide	22
	Modify an Existing Report, SQL where clause.....	23
	Scope	23
	Step-by-step Guide	23
	Process for Modifying the "General Report" in NRC	26
	Scope	26
	Step-by-step Guide	26
	Working with Custom Virtual Attributes.....	29
	Scope	29
	Step-by-Step Guide.....	29
	Database Schema.....	30
	DRA Reporting Database	30
4	Custom Triggers	71
	Collect Additional Virtual Attributes and Extend Object Property Page	71
	Scope	71
	Step-by-step Guide	71
	Enable or disable an Exchange Online user's Outlook web access when they are enabled or disabled in Active Directory.....	78
	Scope	78

Step-by-step Guide	78
Launch Workflow from Trigger	81
Scope	81
Step-by-step Guide	81
Move a user to a new OU when their Office property is changed	84
Scope	84
Step-by-step Guide	84
5 Web Client Customization.....	88
Custom Property Page Examples;	88
Add Virtual Attributes to Custom Forms.....	89
Step-by-step Guide	89
Indicate Field (A) is Required When Field (B) has a Value at Runtime	93
Step-by-step Guide	93
On Form Submit, Validate Field (A) has a Value When Field (B) has a Value.....	94
Step-by-step Guide	94
Populate selection values for a field based on previous field selection	96
Step-by-step Guide	96
Pre-populate Value (A) Based on Value (B) Using Built-in Attribute Setting	101
Step-by-step Guide	101
Pre-populate value automatically based on custom handler	103
Step-by-step Guide	103
Validate input using a regular expression	105
Built-in example regular expressions:	105
Step-by-step Guide	106
Validate Name Uniqueness for a Specified Field Value at Run-time	108
Step-by-step Guide	108
Validate Name Uniqueness for a Specified Field Value When a Form is Submitted	113
Step-by-step Guide	113
6 Workflow Customization	117
Custom Form Workflow to delete a user.....	119
Lab Data Setup	122
HumanResources Database	122
IQMail setup	122
Load HR DB to DRA	124
ServiceNow Event Trigger	126

About this Guide

The *Product Customization* reference provides several examples of the capability to customize reports, triggers, object property forms, and automated workflow processes using Directory and Resource Administrator (DRA) and its integrated components.

Intended Audience

This guide provides information for anyone using the customization features in the DRA Web Console.

Additional Documentation

This guide is part of the Directory and Resource Administrator documentation set. For the most recent version of this guide and other DRA documentation resources, visit the [DRA Documentation website](https://www.netiq.com/documentation/directory-and-resource-administrator/index.html) (<https://www.netiq.com/documentation/directory-and-resource-administrator/index.html>).

Contact Information

We want to hear your comments and suggestions about this book and the other documentation included with this product. You can use the [comment on this topic](#) link at the bottom of each page of the online documentation, or send an email to Documentation-Feedback@microfocus.com.

For specific product issues, contact Micro Focus Customer Care at <https://www.microfocus.com/support-and-services/>.

Product Customization Scenario Details

The examples in this document are all based on a simulated company, "Keystone Steel". This page contains information about the company, network environment, and users that will be referenced in the customizations to follow.

Active Directory Configuration Script:

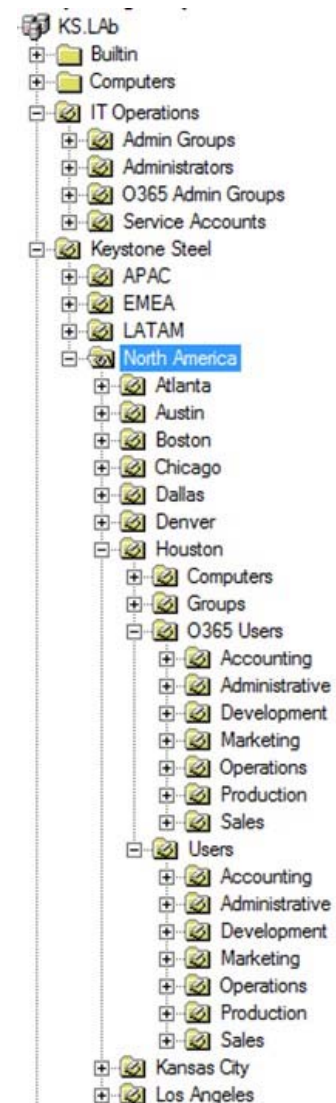
Use the files below from the resource kit to load the OUs. This script will load all of the OUs defined in the Keystone Steel domain. The Resource Kit is available with this document on the DRA Documentation page.

- [PowerShell Script](#)
- [OU CSV File](#)

Active Directory Structure Reference:

User Personas

Philip Jensen	CIO
Daniel Altaner the Architect	Daniel Altaner owns the "big picture" design of IAS for a large organization, providing the foundation for individual colleagues to focus on security, identity, and/or access requirements. In smaller organizations, Daniel might be the sole architect for the combined IAS solution.
Adam Mandari the Administrator	Adam Mandari takes requirements from Daniel Altaner, the IT architect, to set up and maintain the IAS solution and ensure that the infrastructure for IAS works. He helps configure the system so the enterprise knows who has access to what and the methods for ensuring security or analyzing events that might represent breaches in security. He installs and configures the backend agents and services.
Danita Olivarez in DevOps Director	Danita Olivarez is the Director of DevOps / DevSecOps
Devin in DevOps	Devin Merrill, DevOps engineer / operator
Arvanti Rana the Application/Resource Owner	Arvanti Rana is responsible for a specific application, service, or resource, such as a database or the SAP financial module. She is considered both a technical expert (SME) and business user. Arvanti administers her application, but might not understand the nuts and bolts of the organization's IAS solutions. Arvanti needs information from and provides information to the IAS solutions. She maintains and configures the applications, services, and resources that form a part of the iAS solution.



Arturo Perez the Auditor	Arturo Perez is responsible for verifying that the organization complies with established internal control procedures by examining records, reports, operating practices, and documentation. He might delegate certain tasks to his staff to ensure that the work is performed efficiently.
Ken Nagai in Compliance	Ken Nagai is responsible for architecting the IAS governance policies for an organization. He understands both the benefits and limitations of identity and access review solutions and owns the results of access certification reviews.
Isaac in IT	Isaac Torek is the IT Manager.
Mandy the Analyst	Mandy Rabani is responsible for the forensic analysis of alerts and incidents, based on information supplied by Oskar the Operator and the SIEM solution. She performs the situational assessment of what's going on based on the details that she gathers. She is also responsible for preventing data loss and service interruptions by researching new technologies that will effectively protect a network.
Oskar the Operator	Oskar Yegorov is responsible for identifying incoming alerts in the security software. He performs an initial analysis to determine whether the alerts should be escalated to a higher level of investigation. He might complete that investigation or forward it to a system or security analyst. He helps coordinate responses to computer security incidents and recommends actions for each incident to ensure that they are properly tracked, escalated, and reported.
Helen Winzen on the Help Desk	Helen Winzen provides the first line of support when an end user, such as Maria Belafonte, runs into identity, access, or software/hardware problems. She is technically astute and understand the nuts and bolts of identity and access review solutions. She also understands and works to ensure that users follow the organization's security policies.
Sarah the Supervisor / Business Line Manager	Sarah Gibson directly manages a team, and is sometimes responsible for requesting or granting access to resources for her team. Sarah needs information from and provides information to access certification reviews. She can supervise one of several departments, such as Marketing or Help Desk. Sarah ensures that her employees understand and comply with business and government regulations, and that they use the appropriate applications, resources, and services.
Maria in Marketing / End User	Maria Belafonte represents an end user on the corporate network. An end user could be any employee with a corporate identity who requires access to resources. She needs prompt access to resources with a minimum of thought or effort. Sometimes she is required to act on behalf of her Supervisor to approve or deny requests for access.

Customizing NRC Reports

This section contains customization examples and the database schema for NetIQ Reporting Center reports.

Definition of terms and components:

Term/Components	Definition
General Report	Contains the definition of all the reports available in NRC.
User Defined Context	Filter for adjusting the report results for a specific report execution.
Custom Virtual Attribute	Custom DRA extension attribute
Data Model	Available objects and attributes
Collectors	Component/Jobs for periodically collecting reporting data from DRA, Auditing, AD, and O365

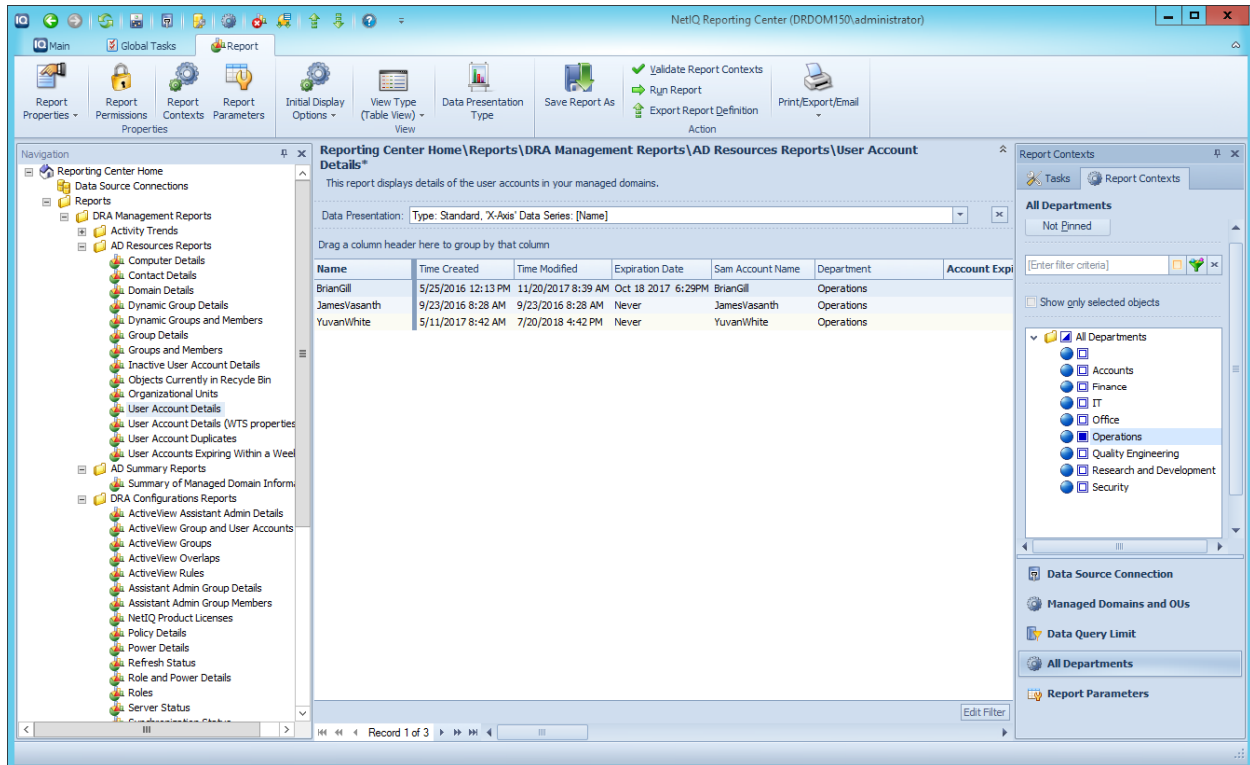
Examples:

Use Case	Components for Example
Extend an existing report with new filter (UDContext) to add additional filter to fetch the data. Example	• UD Contexts
Create a new report to show all collectors runs with job collection type, start/end time and status of the job within the last 7 days. Example	• New report templates
Create a new report to list all temporary group details and its associated members. Example	• New report templates
Create a new report to list all the VA attributes and their associated object types. Example	• New report templates
Extend a default report with an additional virtual attribute Example	• Virtual Attribute Configuration • Updating report template (Columns returned)
Modify the existing report that fetches user account data that expires within a week to instead fetch user accounts expiring within the next 30 days. Example	• Updating report template (SQL Where clause)
Put together a database schema reference Reference	

Create a User Defined (UD) Context for report filtering

Scope

The Reporting Center (NRC), allows for reports to be designed to accept user input at report execution time to dynamically filter and sort the information returned through the use of **"User Defined Contexts"**. In this example we will demonstrate the steps to create a new User Defined Context for filtering on the User object attribute "Department". Then we will extend the "User Accounts Details" report to use the new User Defined Context to allow filtering the returned results by department.



Step-by-step Guide

1. Follow the [Process for Modifying the "General Report" in NRC](#) up to the uploading section.
2. Create a SQL query to filter Users returned by Department.

SQL Statement

```
SELECT DISTINCT DEPARTMENT
FROM USERS
ORDER BY DEPARTMENT
```

NOTE:

You can use SQL Server Management Studio against the DRAReporting database to test the SQL statement to make sure it returns the desired results.

3. Create a `DRA.Custom.UDContext.XML` file if it does not exist. Update the Checkin batch file if needed.

DRA.Custom.UDContext.xml

```
<?xml version="1.0" encoding="utf-8"?>
<UDCONTEXTS>
</UDCONTEXTS>
```

Checkin Batch File

```
@echo off
REM Checkin DRA NRC Reports
if Exist "%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" goto Checkin
echo "%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe not found, check
path of NetIQ Reporting Center"
Goto End

:Checkin
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.DataSourceType.XML"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.UDContext.xml"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.ExtDataSource.XML"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Report.XML"

if not exist "%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Custom.UDContext.xml"
goto SkipToReport
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Custom.UDContext.xml"

:SkipToReport
if not exist "%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Custom.Reports.XML" goto
End
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Custom.Reports.XML"

:End
echo DRA NRC Reporting setup Done!
```

4. Next, in the DRA.Custom.UDContext.XML file and add a User Defined Context definition that specifies how this configuration will be exposed in the report and then save the file.

SQL Statement

```
<UDCONTEXT ID="DRADEPARTMENTCONTEXT" DataTarget="SQL" Type="DRAReporting">
  <SQLQUERY>
    <![CDATA[
      CREATE TABLE #tmpud
      (
        [udLevel] NVARCHAR(max) NULL,
        [udObject] NVARCHAR(max) NULL ,
        [udObjectParent] NVARCHAR(max) NULL,
        [udObjectName] NVARCHAR(max) NULL,
        [udObjectUIName] NVARCHAR(max) NULL,
        [udOutput] NVARCHAR(max) NULL,
        [udIcon] NVARCHAR(max) NULL,
        [udSortOrder] NVARCHAR(max) Null
      )

      INSERT #tmpud (udLevel, udObject, udObjectParent, udObjectName, udObjectUIName,
udOutput, udIcon, udSortOrder)

      SELECT DISTINCT
        [udLevel] = 0,
        [udObject] = 'AllDepartments',
        [udObjectParent]= NULL,
        [udObjectName] = 'All Departments',
        [udObjectUIName]= 'All Departments',
        [udOutput]= 'All Departments',
        [udIcon] = 0,
        [udSortOrder] = 0

      INSERT #tmpud (udLevel, udObject, udObjectParent, udObjectName, udObjectUIName,
udOutput, udIcon, udSortOrder)

      SELECT DISTINCT
        [udLevel] = 1,
        [udObject] = DEPARTMENT,
        [udObjectParent]= 'AllDepartments',
        [udObjectName] = DEPARTMENT,
        [udObjectUIName]= DEPARTMENT,
        [udOutput]= DEPARTMENT,
        [udIcon] = 0,
        [udSortOrder] = 0
      FROM [dbo].[Users]
      ORDER BY DEPARTMENT
    ]>
  </SQLQUERY>
</UDCONTEXT>
```

```

        SELECT * FROM #tmpud
        ORDER BY [udLevel],[udObjectName]

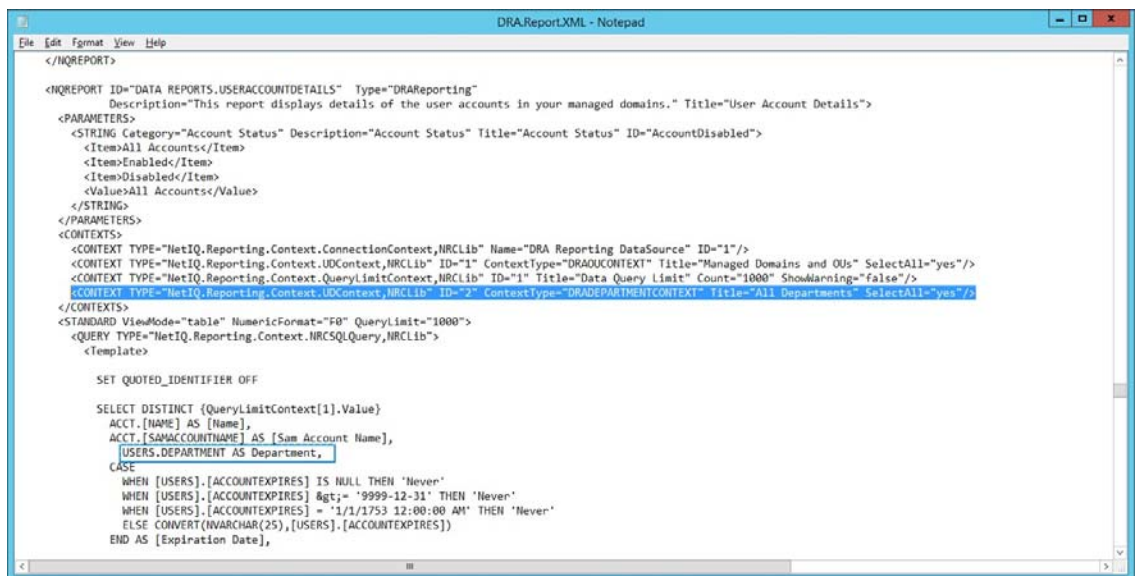
        DROP TABLE #tmpud

    ]]>
</SQLQUERY>

<RETURNDATA Type="SQL" Format="Set" ColumnName="" />
</UDCONTEXT>

```

- Now open the .\Program Files (x86)\NetIQ\DR\DR Reporting\DR.Report.XML file and extract the section for the "User Account Details" report to the custom report file. We will update the report definition to include the new User Defined Context that we created in the previous step.



Edit Report

Make the following changes to the report definition:

- Add the new context inside the <CONTEXTS> tags: **<CONTEXT TYPE="NetIQ.Reporting.Context.UDContext,N RCLib" ID="2" ContextType="DRADEPARTMENTCONTEXT"**
- Title="All Departments" SelectAll="yes"/>** Add the following line to the SQL statement as outlined in the blue box in the screenshot to the left: **USERS.DEPARTMENT AS Department,** just below the **ACCT.[SAMACCOUNTNAME] as [Sam Account Name],**
- Add the following line to the WHERE clause of the SQL statement as outlined in the blue box in the screenshot to the left: **AND ((1 = CASE when**
- {UDContext[2].FLDRID} = 'AllDomains' then 1 ELSE 0 END) OR (USERS.DEPARTMENT IN ({UDContext[2].NODEID})))**

```

File Edit Format View Help
DRAReportXML - Notepad

    WHEN [USERS].[RASPERMISSION] &gt; 7 THEN 1
    ELSE 0
END AS [User Dialin Privilege],
[ACCT].[DISTINGUISHEDNAME] AS [Distinguished Name],
[ACCT].[DESCRIPTION] AS [Description],
[USERS].[WHENCREATED] AS [Time Created],
[USERS].[WHENCHANGED] AS [Time Modified]
FROM
dbo.GET_GUID_TABLE ("(UDCONTEXT[1].NODEID),(UDCONTEXT[1].FLDRID)")
INNER JOIN (SELECT objectGuid, Parent_GUID, distinguishedName, samAccountName, name, description FROM ACCOUNTS WITH (NOLOCK) WHERE CLASS_ID = 5) AS ACCT
ON GET_GUID_TABLE.GUID = ACCT.PARENT_GUID
INNER JOIN CONTEXTTABLEFOROU WITH (NOLOCK) ON GET_GUID_TABLE.GUID = CONTEXTTABLEFOROU.UOOBJECT
INNER JOIN USERS WITH (NOLOCK) ON ACCT.OBJECTGUID = [USERS].OBJECTGUID
INNER JOIN DOMAINS WITH (NOLOCK) ON DOMAINS.DOMAIN_ID = [USERS].DOMAIN_ID
WHERE
(
((DOMAINS.MCSTSMANAGED = 1) OR (DOMAINS.MCSREQSUBSETMANAGED = 1 AND CONTEXTTABLEFOROU.ISMANAGED &gt; 2))
AND ((1 = CASE WHEN (Parameter[AccountDisabled]) = 'All Accounts' THEN 1 ELSE 0 END ) OR
(USERS.ACCOUNTDISABLED = (CASE WHEN (Parameter[AccountDisabled]) = 'Enabled' THEN 0 WHEN (Parameter[AccountDisabled]) = 'Disabled' THEN 1 END) ))
AND ((1 = CASE WHEN (UDCONTEXT[2].FLDRID) = 'AllDomains' THEN 1 ELSE 0 END) OR
(USERS.DEPARTMENT IN ((UDCONTEXT[2].NODEID))))
)
ORDER BY [Managed Domain Name], [Name]

SET QUOTED_IDENTIFIER ON
</Template>
<?QUERY>
<COLUMN Source="Time Created" Format="g"/>
<COLUMN Source="Time Modified" Format="g"/>
<COLUMN Source="Expiration Date" Format="g"/>
</STANDARD>
</NOREPORT>

<NOREPORT ID="DATA REPORTS.USERACCOUNTDETAILSWTSPROPERTIES" Type="DRAReporting">

```

NOTE:

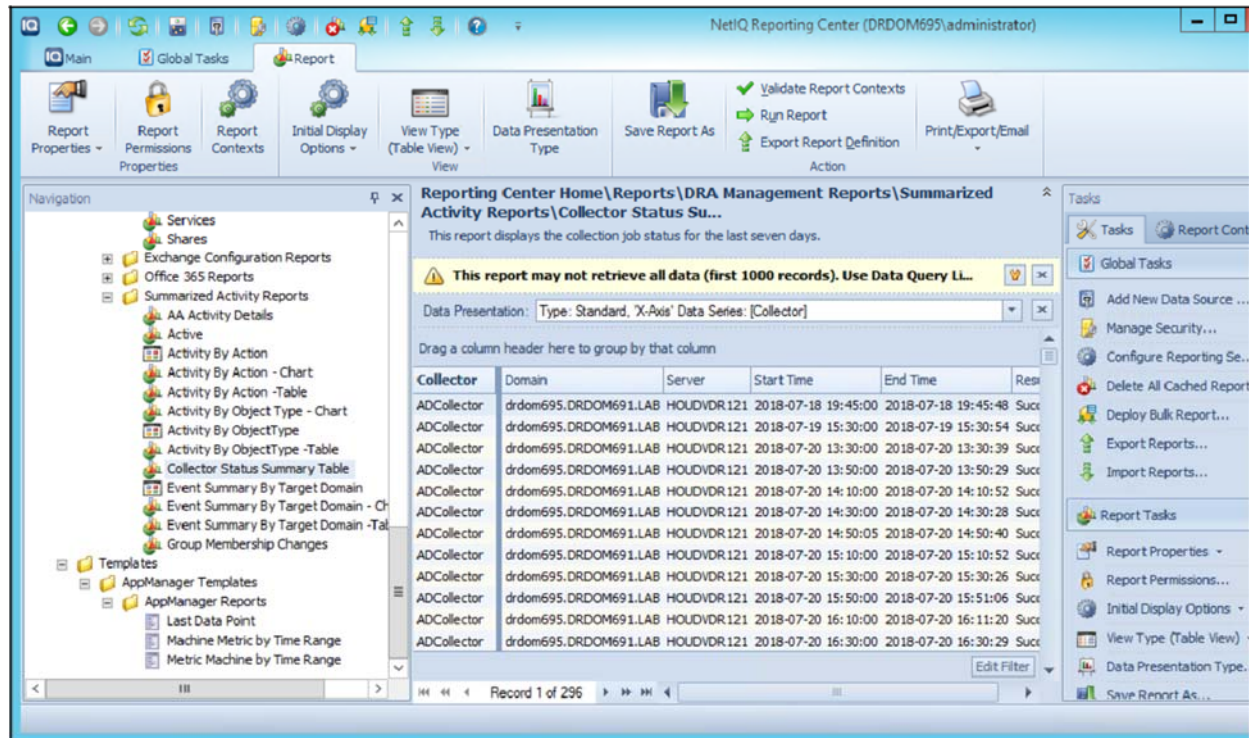
The UDContext we are adding in this step has an ID="2". Later, when the context is executed, it will be referenced as UDContext[2].

6. Save the master report file to the . \Program Files (x86) \NetIQ\DNA\DNA folder. Follow the final step in [Process for Modifying the "General Report" in NRC uploading section.](#)

Creating a "Collections Status for the last seven days" Report

Scope

The Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the SQL database. Additionally, NRC allows for the modification and addition of reports to provide a custom solution. In this example we will demonstrate the steps to create a new report that shows the status of all collector jobs executed within the last seven days.



Step-by-step Guide

1. Follow the [Process for Modifying the "General Report" in NRC](#) up to the uploading section.
2. You can use SQL Server Management Studio against the DRAReporting database to test the SQL statement we will use for this report to make sure it returns the desired results.

SQL Statement

```
SELECT  J1.Job_Name AS Collector,
        JH.Collector_Info AS Domain,
        JH.DRA_Server_Name AS Server,
        CONVERT(NVARCHAR(35), JH.Start_Time, 120) AS [Start Time],
        CONVERT(NVARCHAR(35), JH.End_Time, 120) AS [End Time],
        CASE WHEN JH.Result_Id = 1 THEN 'Success'
              WHEN JH.Result_Id = 2 THEN 'Failed'
              ELSE 'Aborted'
        END AS [Result]
```

```

FROM      JOB as J1 INNER JOIN
          JOBS_HISTORY AS JH ON J1.Job_Id = JH.Job_Id
WHERE (JH.Job_Id <> 3) AND (JH.Start_Time >= DATEADD(DD, - 7, GETDATE()))
UNION ALL
SELECT J2.Job_Name,
       RJ.Collector_Info,
       RJ.DRA_Server_Name,
       CONVERT(NVARCHAR(35), RJ.Start_Time, 120),
       NULL AS End_Time,
       'Running' AS [Result]
FROM      JOB AS J2 INNER JOIN
          RUNNING_JOBS AS RJ ON J2.Job_Id = RJ.Job_Id
WHERE (J2.Job_Id <> 3)
ORDER BY Domain, Server, Collector, [Start Time] DESC

```

3. Open the `.\Program Files (x86)\NetIQ\DNA\DNA Reporting\DNA.Report.XML` file and locate the section for the "Summarized Activity Reports" group and extract it to the custom report file. Add the definition for the new report inside this section of the XML:

<REPORTSET FOLDER="DNA Management Reports\Summarized Activity Reports">

Report Definition

```

<NQREPORT ID="SUMMARIZED ACTIVITY REPORTS.COLLECTOR STATUS SUMMARY TABLE"
  Type="DRAReporting"
  Description="This report displays the collection job status for the last seven days."
  Title="Collector Status Summary Table">
  <PARAMETERS/>
  <CONTEXTS>
    <CONTEXT TYPE="NetIQ.Reporting.Context.ConnectionContext,NRCLib" Name="DNA Reporting
DataSource" ID="1"/>

    <CONTEXT TYPE="NetIQ.Reporting.Context.QueryLimitContext,NRCLib"
ID="1" Title="Data Query Limit" Count="1000" ShowWarning="false"/>
  </CONTEXTS>
  <STANDARD ViewMode="table" NumericFormat="F0" QueryLimit="1000">
    <QUERY TYPE="NetIQ.Reporting.Context.NRCSQLQuery,NRCLib">
      <Template>
        SELECT  {QueryLimitContext[1].Value}
                J1.Job_Name AS Collector,
                JH.Collector_Info AS Domain,
                JH.DRA_Server_Name AS Server,

```

```

        CONVERT(NVARCHAR(35), JH.Start_Time, 120) AS [Start Time],
        CONVERT(NVARCHAR(35), JH.End_Time, 120) AS [End Time],
        CASE WHEN JH.Result_Id = 1 THEN 'Success'
              WHEN JH.Result_Id = 2 THEN 'Failed'
              ELSE 'Aborted'
        END AS [Result]
FROM      JOB as J1 INNER JOIN
          JOBS_HISTORY AS JH ON J1.Job_Id = JH.Job_Id
WHERE     (JH.Job_Id <> 3) AND (JH.Start_Time >= DATEADD(DD, - 7,
GETDATE()))
UNION ALL
SELECT    J2.Job_Name,
          RJ.Collector_Info,
          RJ.DRA_Server_Name,
          CONVERT(NVARCHAR(35), RJ.Start_Time, 120),
          NULL AS End_Time,
          'Running' AS [Result]
FROM      JOB AS J2 INNER JOIN
          RUNNING_JOBS AS RJ ON J2.Job_Id = RJ.Job_Id
WHERE     (J2.Job_Id <> 3)
ORDER BY Domain, Server, Collector, [Start Time] DESC
</Template>
</QUERY>
</STANDARD>
</MOREPORT>

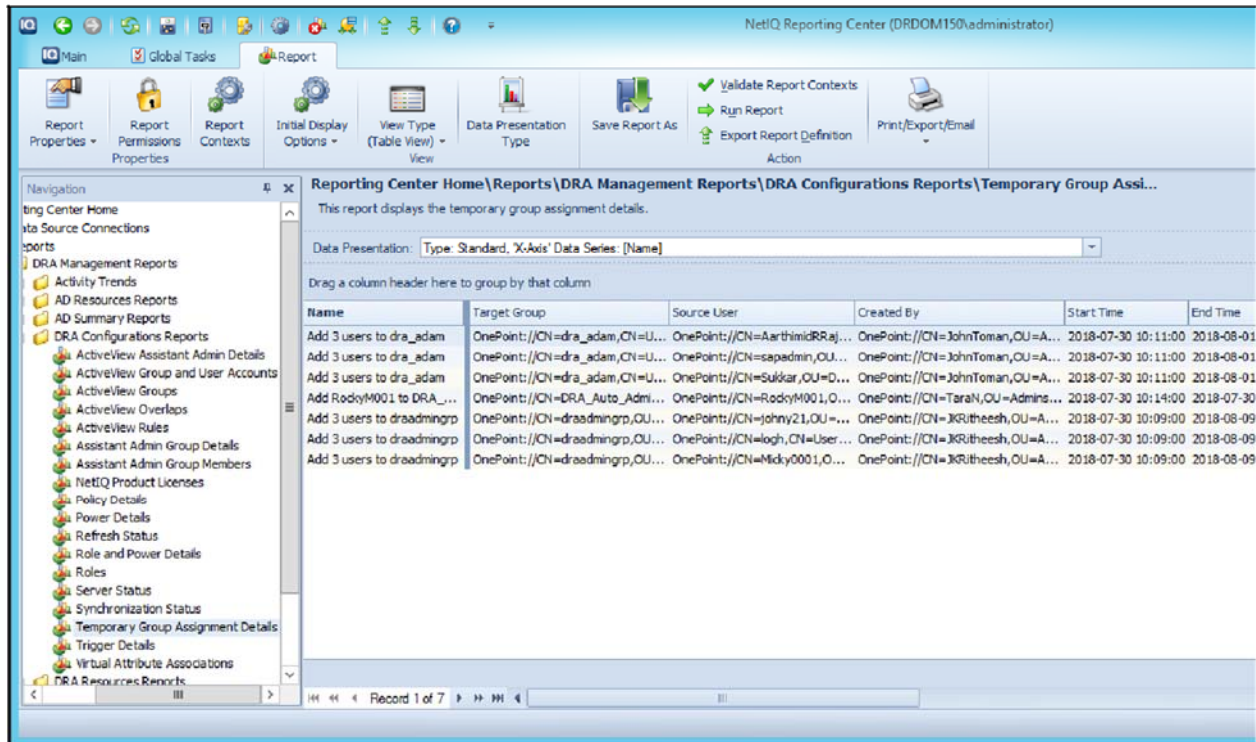
```

4. Save the report file to the .\Program Files (x86)\NetIQ\DRA\DRA folder. Follow the final step in [Process for Modifying the "General Report" in NRC uploading section](#). You will see the new report located under the folder "Summarized Activity Report".

Creating a "Temporary Group Assignment" Report

Scope

The Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the SQL database. Additionally, NRC allows for the modification and addition of reports to provide a custom solution. In this example we will demonstrate the steps to create a new report that shows the list of temporary group assignment details.



Step-by-step Guide

1. Follow the [Process for Modifying the "General Report" in NRC](#) up to the uploading section.
2. You can use SQL Server Management Studio against the DRAREporting database to test the SQL statement we will use for this report to make sure it returns the desired results.

SQL Statement

```
SELECT TGA.NAME AS Name,
       TGA.TARGET_GROUP AS [Target Group],
       TGM.USER_PATH AS [Source User],
       TGA.ASSISTANT_ADMIN AS [Created By],
       CONVERT(NVARCHAR(35), TGA.START_TIME, 120) AS [Start Time],
       CONVERT(NVARCHAR(35), TGA.END_TIME, 120) AS [End Time],
       TGA.STATUS_MESSAGE AS Status,
       TGA.DESCRPTION AS Description,
       TGA.CREATION_TIME as [Creation Time]
```

```

INNER JOIN TGA_GROUP_MEMBERSHIP TGM ON TGA.TGA_ID=TGM.TGA_ID
ORDER BY TGA.TARGET_GROUP, TGM.USER_PATH

```

3. Open the report file and locate the section for the "DRA Configurations Reports" group and extract the section to the custom report file. Place it under this section of the XML:

Report Definition

```

<NQREPORT ID="DRA CONFIGURATION REPORTS.TGADETAILS" Type="DRAReporting"
    Description="This report displays the temporary group assignment details."
    Title="Temporary Group Assignment Details">
    <PARAMETERS/>
    <CONTEXTS>
        <CONTEXT TYPE="NetIQ.Reporting.Context.ConnectionContext,NRCLib" Name="DRA
Reporting DataSource" ID="1"/>

        <CONTEXT
TYPE="NetIQ.Reporting.Context.QueryLimitContext,NRCLib" ID="1"
Title="Data Query Limit" Count="1000" ShowWarning="false"/>
    </CONTEXTS>
    <STANDARD ViewMode="table" NumericFormat="F0" QueryLimit="1000">
        <QUERY TYPE="NetIQ.Reporting.Context.NRCSQLQuery,NRCLib">
            <Template>
                SELECT {QueryLimitContext[1].Value}
                    TGA.NAME AS Name,
                    TGA.TARGET_GROUP AS [Target Group],
                    TGM.USER_PATH AS [Source User],
                    TGA.ASSISTANT_ADMIN AS [Created By],
                    CONVERT(NVARCHAR(35), TGA.START_TIME, 120) AS [Start Time],
                    CONVERT(NVARCHAR(35), TGA.END_TIME, 120) AS [End Time],
                    TGA.STATUS_MESSAGE AS Status,
                    TGA.DESCRPTION AS Description,
                    TGA.CREATION_TIME as [Creation Time]
                FROM TGA
                INNER JOIN TGA_GROUP_MEMBERSHIP TGM ON TGA.TGA_ID=TGM.TGA_ID
                ORDER BY TGA.TARGET_GROUP, TGM.USER_PATH
            </Template>
        </QUERY>
    </STANDARD>
</NQREPORT>

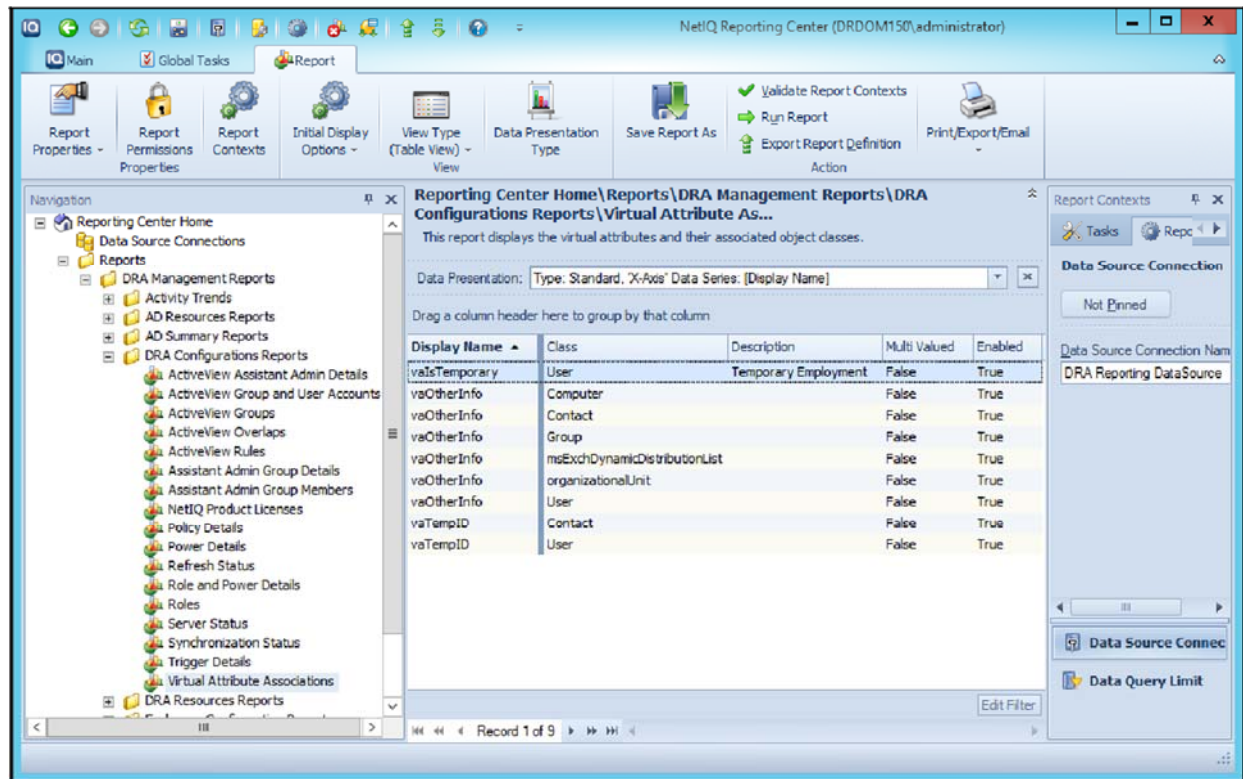
```

4. Save the report file to the .\Program Files (x86)\NetIQ\DRA\DRA folder. Follow the final step in [Process for Modifying the "General Report" in NRC](#) uploading section. You will see the new report located under the folder "DRA Configurations Reports".

Creating a "Virtual Attribute Associations" Report

Scope

The Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the SQL database. Once the report has been created, it can be sorted and/or filtered as needed. Additionally, NRC allows for the modification and addition of reports to provide a custom solution. This procedure will demonstrate the steps to create a new report that shows the list of virtual attributes and their association with classes.



Step-by-step Guide

1. Follow the [Process for Modifying the "General Report" in NRC](#) up to the uploading section.
2. You can use SQL Server Management Studio against the DRAReporting database to test the SQL statement we will use for this report to make sure it returns the desired results.

SQL Statement

```
SELECT
    VA.DisplayName as [Display Name],
    CLASS.CLASS AS [Class],
    VA.Description,
    VA.IsMultiValued AS [Multi Valued],
    VA.Enabled
```

```

FROM VIRTUAL_ATTRIBUTE VA
    INNER JOIN VA_CLASS VC ON VA.VIRTUAL_ATTRIBUTE_ID=VC.VIRTUAL_ATTRIBUTE_ID
    INNER JOIN CLASS ON CLASS.CLASS_ID=VC.CLASS_ID
ORDER BY VA.DisplayName, CLASS.CLASS

```

3. Open the custom report file and locate the section for the "DRA Configurations Reports" group and add the definition for the new report inside this section of the XML:

<REPORTSET FOLDER="DRA Management Reports\Summarized Activity Reports">

Report Definition

```

<NQREPORT ID="DRA CONFIGURATION REPORTS.VAASSOCIATIONDETAILS" Type="DRAReporting"

Description="This report displays the virtual attributes and their
associated object classes." Title="Virtual Attribute Associations">
    <PARAMETERS/>
    <CONTEXTS>
        <CONTEXT TYPE="NetIQ.Reporting.Context.ConnectionContext,NRCLib" Name="DRA Reporting
DataSource" ID="1"/>

        <CONTEXT TYPE="NetIQ.Reporting.Context.QueryLimitContext,NRCLib"
ID="1" Title="Data Query Limit" Count="1000" ShowWarning="false"/>
    </CONTEXTS>
    <STANDARD ViewMode="table" NumericFormat="F0" QueryLimit="1000">
        <QUERY TYPE="NetIQ.Reporting.Context.NRCSQLQuery,NRCLib">
            <Template>
                SELECT {QueryLimitContext[1].Value}
                    VA.DisplayName as [Display Name],
                    CLASS.CLASS AS [Class],
                    VA.Description,
                    VA.IsMultiValued AS [Multi Valued],
                    VA.Enabled
                FROM VIRTUAL_ATTRIBUTE VA
                    INNER JOIN VA_CLASS VC ON VA.VIRTUAL_ATTRIBUTE_ID=VC.VIRTUAL_ATTRIBUTE_ID
                    INNER JOIN CLASS ON CLASS.CLASS_ID=VC.CLASS_ID
                ORDER BY VA.DisplayName, CLASS.CLASS
            </Template>
        </QUERY>
    </STANDARD>
</NQREPORT>

```

4. Save the report file to the .\Program Files (x86)\NetIQ\DNA\DNA folder. Follow the final step in [Process for Modifying the "General Report" in NRC](#) uploading section. You will see the new report located under the folder "DNA Configurations Reports".

Modify an Existing Report, SQL column addition

Scope

The Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the DRA SQL reporting database. Additionally, NRC allows for the modification and addition of reports to provide a custom solution. In this example we will add an additional column to be returned in the report.

Step-by-step Guide

1. Follow the [Process for Modifying the "General Report" in NRC](#) up to the uploading section.
2. Open the `.\Program Files (x86)\NetIQ\DRA\DRA Reporting\DRA.Report.XML` file and locate the section for the "User Accounts Expiring Within a Week" report. Extract a copy to the custom report file.
3. Next, update the SQL database query return the extra column.

Edit SQL Query

Make the following changes to the SQL query:

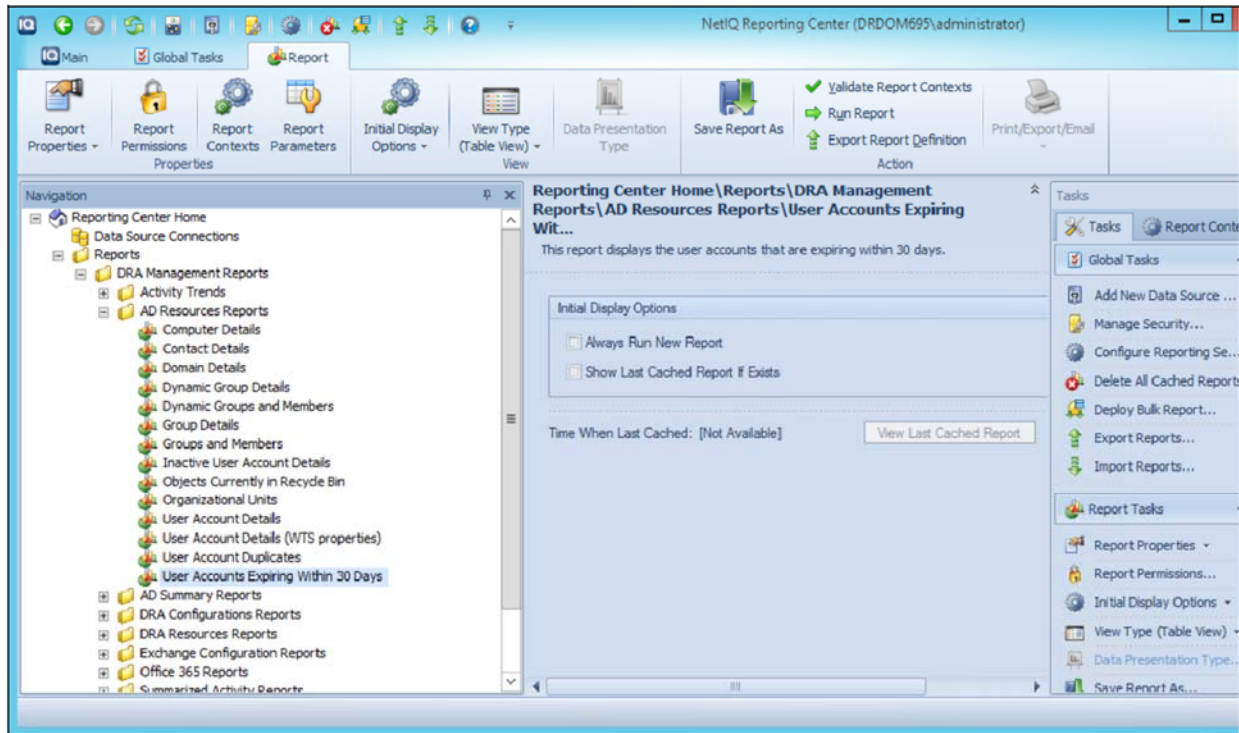
Add `[USERS].[DeskLocation] AS [Desk Location],`

4. Save the report file. Follow the final step in [Process for Modifying the "General Report" in NRC](#) uploading section.

Modify an Existing Report, SQL where clause

Scope

The Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the DRA SQL reporting database. Additionally, NRC allows for the modification and addition of reports to provide a custom solution. In this example we will modify the existing "User Accounts Expiring Within a Week" with a new custom date range of 30 days to convert the report to "User Accounts Expiring Within 30 Days".



Step-by-step Guide

1. Follow the Process for [Modifying the "General Report" in NRC](#) up to the uploading section.
2. Open report file and locate the section for the "User Accounts Expiring Within a Week" report and extract a copy to the custom report file. First we will update the contextual information for the report.


```

<NRREPORT ID="DATA REPORTS.USERACCOUNTSEXPIRINGWITHINAWEEK" Type="DRAReporting"
  Description="This report displays the user accounts that are expiring within a week." Title="User Accounts
Expiring Within a Week">
  <PARAMETERS>
    <STRING Category="Account Status" Description="Account Status" Title="Account Status" ID="AccountDisabled">
      <Item>All Accounts</Item>
      <Item>Enabled</Item>
      <Item>Disabled</Item>
      <Value>All Accounts</Value>
    </STRING>
  </PARAMETERS>
  <CONTEXTS>
    <CONTEXT TYPE="NetIQ.Reporting.Context.ConnectionContext,NRCLib" Name="DRA Reporting DataSource" ID="1"/>
    <CONTEXT TYPE="NetIQ.Reporting.Context.UContext,NRCLib" ID="1" ContextType="DRAOUCONTEXT" Title="Managed Domains and
OUs" SelectAll="yes"/>
    <CONTEXT TYPE="NetIQ.Reporting.Context.QueryLimitContext,NRCLib" ID="1" Title="Data Query Limit" Count="1000"
ShowWarning="false"/>
  </CONTEXTS>
  <STANDARD ViewMode="table" NumericFormat="F0" QueryLimit="1000">
    <QUERY TYPE="NetIQ.Reporting.Context.NRCSQLQuery,NRCLib">
      <Template>

        SET QUOTED_IDENTIFIER OFF

        SELECT DISTINCT {QueryLimitContext[1].Value}
        ACCOUNTS.[NAME]as [Name],
        ACCOUNTS.SAMACCOUNTNAME as [Sam AccountName],

```

Edit Report

Make the following changes to the report definition:

- Change the entry for "Description" to **Description="This report displays the user accounts that are expiring within 30 days"**
- Change the entry for "Title" to **Title="User Accounts Expiring Within 30 Days"**

3. Next, update the SQL database query to look for accounts expiring in 30 days instead of 7 days.

```

CASE
  WHEN [USERS].[RASPERMISSION] > 7 THEN 1
  ELSE 0
END AS [User Dialin Privilege],
[ACCOUNTS].[DISTINGUISHEDNAME] as [Distinguished Name],
[ACCOUNTS].[DESCRIPTION] as [Description],
[USERS].[WHENCREATED] AS [Time Created],
[USERS].[WHENCHANGED] AS [Time Modified]
FROM
  dbo.GET_GUID_TABLE ("UDCONTEXT[1].NODEID",{UDCONTEXT[1].FLDRID})
  INNER JOIN ACCOUNTS WITH (NOLOCK) ON GET_GUID_TABLE.GUID = ACCOUNTS.PARENT_GUID
  INNER JOIN CONTEXTTABLEFOROU on GET_GUID_TABLE.GUID = CONTEXTTABLEFOROU.UOBJECT
  INNER JOIN USERS ON ACCOUNTS.OBJECTGUID = [USERS].OBJECTGUID
  INNER JOIN DOMAINS ON DOMAINS.DOMAIN_ID = [USERS].DOMAIN_ID
WHERE (((DOMAINS.MCSISMANAGED = 1) OR (DOMAINS.MCSREQSUBSETMANAGED = 1 AND CONTEXTTABLEFOROU.ISMANAGED > 2))
AND [USERS].[ACCOUNTEXPIRES] >= GetDate() AND [USERS].[ACCOUNTEXPIRES] < DATEADD(day, 30, GetDate()))
AND ((1 = CASE when {Parameter[AccountDisabled]} = 'All Accounts' then 1 ELSE 0 END ) OR
(USERS.ACCOUNTDISABLED = (case when {Parameter[AccountDisabled]} = 'Enabled' then 0 when {Parameter
[AccountDisabled]} = 'Disabled' then 1 end )))
)

SET QUOTED_IDENTIFIER ON

</Template>
</QUERY>
<COLUMN Source="Time Created" Format="g"/>
<COLUMN Source="Time Modified" Format="g"/>
</STANDARD>

```


Edit SQL Query

Make the following changes to the SQL query:

Change ***DATEADD(day, 7, GetDate())*** to ***DATEADD(day, 30, GetDate())***

4. Save the file. Follow the final step in [Process for Modifying the "General Report"](#) in NRC uploading section.

Process for Modifying the "General Report" in NRC

Scope

Reporting Center (NRC), provides a set of standard reports that can be run to retrieve data from the SQL database. Once the report has been created, it can be sorted and/or filtered as needed. Additionally, NRC allows for the modification and addition of new reports to provide a custom solution. This procedure will cover the general mechanics of making custom reports.

Step-by-step Guide

1. Back up XML files.
2. Edit the Checkin batch file to appear like the example.

Checkin Batch File

```
@echo off
REM Checkin DRA NRC Reports
if Exist "%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" goto Checkin
echo "" "%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe not found, check
path of NetIQ Reporting Center"
Goto End

:Checkin
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DRA\DRA Reporting

\DRData.DataSourceType.XML"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DRA\DRA Reporting

\DRData.UDContext.xml"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DRA\DRA Reporting

\DRData.ExtDataSource.XML"
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DRA\DRA Reporting

\DRData.Report.XML"

if not exist "%~d0\Program Files (x86)\NetIQ\DRA\DRA Reporting\DRData.Custom.UDContext.xml"
goto SkipToReport
```

```

"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DNA\DNA Reporting

\DNA.Custom.UDContext.xml"

:SkipToReport
if not exist "%~d0\Program Files (x86)\NetIQ\DNA\DNA Reporting\DNA.Custom.Reports.XML" goto
End
"%~d0\Program Files\NetIQ\Reporting Center\Utilities\XmlCheckIn.exe" /S localhost
"%~d0\Program Files (x86)\NetIQ\DNA\DNA Reporting

\DNA.Custom.Reports.XML"

```

3. Create a `DNA.Custom.Report.xml` file in the directory if one does not exist.

```

<?xml version="1.0" encoding="utf-8"?>
<REPORTSETS>
  <REPORTSET FOLDER="DNA Management Reports\Summarized Activity Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\Activity Trends">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\Exchange Configuration Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\DNA Configurations Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\AD Resources Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\Office 365 Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\AD Summary Reports">
  </REPORTSET>
  <REPORTSET FOLDER="DNA Management Reports\DNA Resources Reports">
  </REPORTSET>
</REPORTSETS>

```

4. Edit the `DNA.Custom.Report.xml` file with the report to be customized.

{QueryLimitContext[x].Value} - Text like this may be in the query. It is a API (like TOP 1000) that will be filled in by NRC when the query is sent to SQL.

> or < - Html rules apply in this XML file. These will be converted to > (greater than) and < (less than) prior being sent to SQL.

UDContext - User description context used for working with data from the controls. Each UDContext used must have a unique id. Please don't modify the existing controls. See the section below for a list of existing contexts.

5. Upload report definitions by running Checkin batch file.

Working with Custom Virtual Attributes

Scope

DRA allows the extension of objects (User, Contact, Computer, Group, Organizational Unit, & Dynamic Distribution Group) through the use of custom virtual attributes. These attributes may be created and managed with custom interface pages, collected from AD LDS to a SQL database, and displayed as part of a report. This walkthrough gives end-to-end guidance on the process of how to setup, manage, collect, and report custom virtual attributes.

For this example, the Keystone Steel Company needs a new custom virtual attribute, DeskLocation. It will be added as a string attribute for the USER object. We will start by defining the new virtual attribute, creating a custom interface, and then adding data.

Step-by-Step Guide

1. Create new virtual attribute DeskLocation.
2. Assign DeskLocation to USER.
3. Add User page to collect data
4. Assign data to DeskLocation.
5. Setup AD Collector to collect data.
6. Follow the [Process for Modifying the "General Report" in NRC](#) up to the section titled "Uploading the Updated Master Report File".
7. Follow the [Modify an Existing Report, SQL column addition](#) to add new DeskLocation column in report.

Database Schema

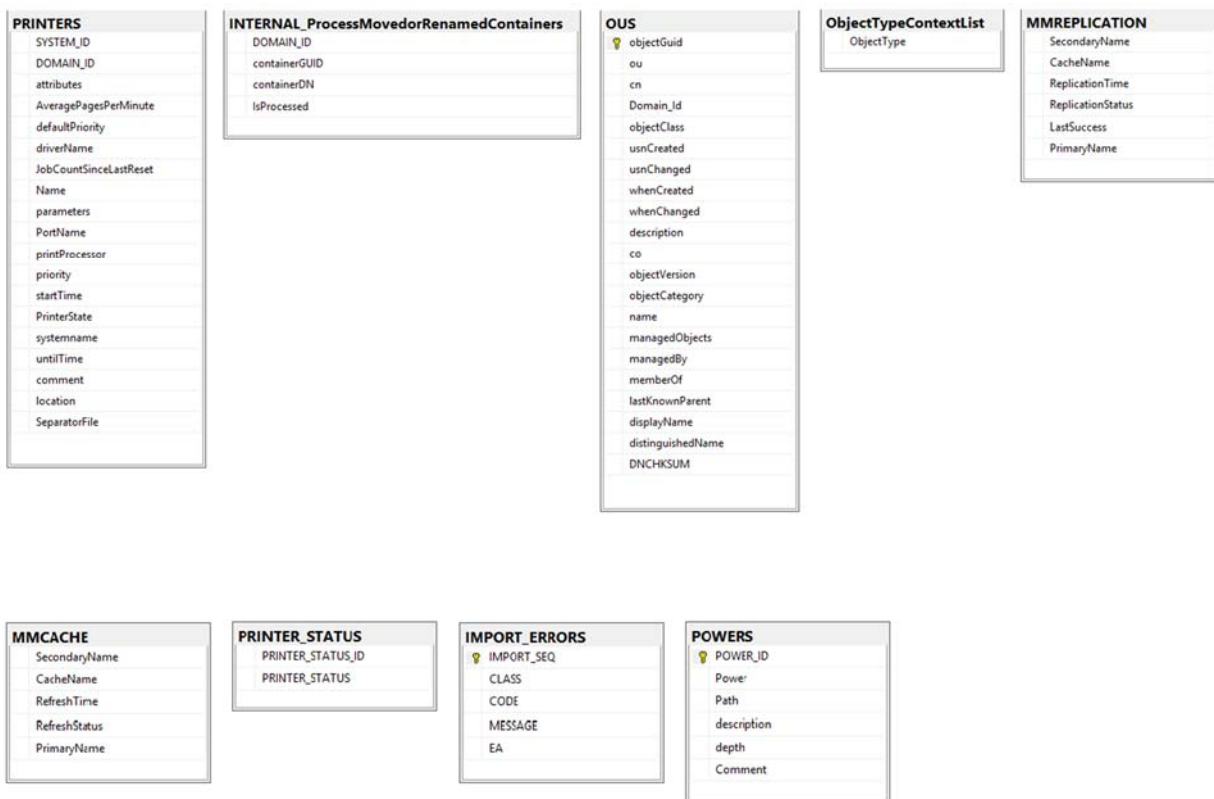
- **DRA Reporting Database**
 - Visual Layout
 - Tables
 - Stored Procedures
 - Functions

DRA Reporting Database

This section includes a visual layout, tables, stored procedures, and functions of the database.

Visual Layout

The DRA Reporting Database stores the data collected by the DRA collectors.



LICENSES
⚡ LICENSE_ID
DisplayName
FinalDay
Flags
GraceExpirationDate
GracePeriodLength
LicenseType
LicensedBy
NumberOfAgents
NumberOfComputers
NumberOfMailboxes
NumberOfOperations
NumberOfRecords
NumberOfServers
NumberOfUsers
NumberOfWorkstations
OverdraftValue
OverdraftPercentage
PackageComment
PackageId
PackageName
PackageVersion
SupportsAllOperations
TrialExpirationDate
TrialPeriodLength
TrialStartDate
CompanyName
CompanyId
SerialNumber
Comment
sNodeLockOn
sShared
sUpgradeOnly

IMPORT_CATEGORY_GROUPS
⚡ CATEGORY_GROUP_ID
CATEGORY_GROUP

PASSWORD_POLICIES
EnablePasswordPolicy
MaxChars
MaxConsecutiveAlpha
MaxDigits
MaxLowerChars
MaxPasswordLength
MaxSameCharRepeat
MaxSpecial
MaxUpperChars
MinChars
MinDigits
MinLowerChars
MinPasswordLength
MinSpecial
MinUpperChars

MP_UPDATES
recordNumber
ClientName
AccountExpires
AccountDisabled
Lock
NetWare
UserName
Comment
FullName
PasswordModify
PasswordReq
PasswordChange
PasswordExpire

LAST_COLLECTED_ATTRIBUTES
Name
DataType
ClassId
DomainId
IsVirtual

POLICIES
⚡ POLICIES_ID
McsClass
McsContainer
McsName
McsNameValue
McsParent
McsPath
McsSchema
Comment
Description
LastModified
Enabled

OPTIONSTABLE
DOMAINID
OPTIONNAME
VALUE

MODPROPS
samAccountName
RecordNumber
LoginScript
WstProfilePath
WstHomeDir
LoginWorkstations
LoginHours
RasPermissions
ModifiedProperties

INTERNAL_GuidNameMap
ObjectGUID
ObjectName
ObjectType

MMSSERVERS
⚡ ServerName
Mode
PrimaryName
AccessAccount
AccessTime
AccessStatus
LastSuccess

SERVICE_STATUS
STATE_TYFE_ID
STATE_TYFE

TGA
TGA_ID
ASSISTANT_ADMIN
CREATION_TIME
DESCRIPTION
END_TIME
START_TIME
STATE
STATUS_MESSAGE
TARGET_GROUP
NAME
ENABLED

SHARE_TYPE
SHARE_TYPE_ID
SHARE_TYPE

TRUSTED_DOMAINS
DOMAIN_NAME
TRUSTED_DOMAIN_NAME
TRUSTED_DOMAIN_DNS_NAME

TRIGGERS
TRIGGER_ID
McsNameValue
McsName
McsPath
McsClass
McsParent
McsSchema
Description
Action
CmdLineArgs
ErrorMsg_General
ErrorMsg_Undo
Fath
UndoCmdLineArgs
UndoPath
Type
Enabled

RULES
RULE_ID
McsNameValue
RuleSpecification
ACTIVEVIEW_ID
McsClass
McsContainer
McsName
McsParent
McsPath
McsSchema
description
includeFlag
nestedFlag
sourceFlag
targetFlag
RuleSpecification_MatchString
RuleSpecification_MatchConta...
RuleSpecification_Name
RuleSpecification_MatchPrinte...
RuleSpecification_Comment
RuleSpecification_ResourceType
RuleSpecification_AccountFilter
RuleSpecification_Recursive
RuleSpecification_SelectBase
RuleSpecification_SelectMemb...
RuleSpecification_Power
RuleSpecification_Description
RuleSpecification_OriginalNam...
ASSISTANT_ADMIN_ID
RuleSpecification_Rule1
RuleSpecification_Rule2
RuleSpecification_SelectSysObj
RuleSpecification_ResourceMsi...
RuleSpecification_ResourceMsi...
RuleSpecification_Version
RuleSpecification_containerBas...
RuleSpecification_containersf...
RuleSpecification_containerMa...
RuleSpecification_containerMa...
RuleSpecification_containerMa...
RuleSpecification_containerPro...
RuleSpecification_containerRe...
RuleSpecification_containerSe...
RuleSpecification_domainMat...
RuleSpecification_domainProp...
RuleSpecification_groupMemb...
RuleSpecification_groupRecurs...
RuleSpecification_groupTypeFi...
RuleSpecification_membersFo...
RuleSpecification_normalFilter
RuleSpecification_objectMatch...
RuleSpecification_objectMatch...
RuleSpecification_objectMatch...
RuleSpecification_objectMust8...
RuleSpecification_objectProper...
RuleSpecification_objectSelect...
RULE_ACC_BIND_ID

Resource_Computer	
objectGUID	
Name	
DOMAIN_D	

RESOURCE_TYPE	
RESOURCE	

SCOPES	
SCOPE_ID	
McsPolicy	
McsPolicyPath	
McsScopeType	
McsRule	

TRIGGER_SCOPES	
McsName/value	
OperationScope	
TRIGGER_ID	
TriggerType	

TGA_GROUP_MEMBERSHIP	
TGA_ID	
USER_PATH	

STATUS	
id	
Domain	
CollectionType	
CollectionStatus	
StartTime	
EndTime	

SERVICES	
SYSTEM_ID	
DOMAIN_ID	
Caption	
[Checkpoint]	
Description	
DesktopInteract	
DisplayName	
ErrorControl	
ExitCode	
PathName	
ServiceSpecificExitCode	
ServiceType	
StartMode	
StartName	
State	
SystemName	
WaitHint	

RECYCLEBIN	
objectGUID	
CLASS_ID	
DOMAIN_ID	
sAMAccountName	
description	
usNCchanged	
New_Name	
Orig_Name	
Orig_ParentDN	
Orig_ParentGUID	
Orig_CN	
whenDeleted	

SHARES	
SYSTEM_ID	
DOMAIN_ID	
AllowMaximum	
MaximumAllowed	
Name	
Path	
Type	
Computer	
Description	

SERVICE_STARTUP	
STARTMODE_TYPE_ID	
STARTMODE_TYPE	

ROLES	
ROLES_ID	
McsClass	
McsContainer	
McsName	
McsName/value	
McsParent	
McsPath	
McsSchema	
Comment	
Creator	
Description	
Flags	
LastModified	
NumAssocDTE	
NumLocalPowers	
NumVestedRoles	
NumOwnerRoles	

ROLE_BINDINGS	
ID	
ASSOC_ID	

USERS - objectGuid - cn - Domain_H - objectClass - userCreated - userChange - HasMailbox - MailboxVersion - homeMDI - accountExpires - name - memberOf - manager - managedObjects - mail - mailNickname - homeDrive - homeDirectory - givenName - displayName - whenCreated - description - canonicalName - AccountDisabled - AccountLockedOut - AccountNotDelegated - DoNotRequireAuth - EncryptTextPasswordAllowed - fpnEnabled - HomeDirectoryRequired - InterDomainTrustAccount - NormalAccount - PasswordDoesNotExpire - PasswordExpired - PasswordNotRequired - Script - ServerTrustAccount - SmartCardRequired - TempDuplicateAccount - TrustedForDelegation - TrustedToAuthenticateForDele... - UseDesKeyOnly - UserAccountControl - WorkstationTrustAccount - whenChanged - userPrincipalName - sAMAccountName - sAMAccountType - RasPermission - RasPhoneNumber - WbAllowLogon - WbBrokenTimeOutSettings - WbDeviceClientDefaultPrinter - WbDeviceClientDrives - WbDeviceClientPrinters - WbHomeDir - WbHomeDirDrive - WbInheritInitialProgram - WbInitialProgram - WbModemCallbackPhoneNumber... - WbModemCallbackSettings - WbProfilePath - WbReconnectSettings - WbRemoveHornDr - WbShadowingSettings - WbTimerSettingsConnecto... - WbTimerSettingsDisconnec... - WbTimerSettingsIdle - WbWorkingDirectory - ou - objectCategory - lastLogonTimestamp - MailboxType	RECYCLE_BIN - RecycleBinID - McDaysRecycleBin - McClass - samAccountName - Domain - McPath - McFriendlyName - McOrgParentCFRecycledObj	USER_MAILBOXES - IMPORT_TIME - MiniVB_DistinguishedName - MiniVB_FirstName - MiniVB_Initials - MiniVB_LastName - MiniVB_DisplayName - MiniVB_Alias - MiniVB_HomeServer - MiniVB_HomeSite - MiniVB_MsgOutgoingLimit - MiniVB_MsgIncomingLimit - MiniVB_UserDefaults - MiniVB_StoreLimitWarn - MiniVB_StoreLimitProhibit - MiniVB_ProhibitSendReceive - MiniVB_DeliveryMechanism - MiniVB_DeletedItemRetention - MiniVB_RID - McPath - Domain	USER_INTERFACE_EXTENSIONS - McName - McPath - Status	TRUSTING_DOMAINS - DCMAIN_NAME - TRUSTING_DOMAIN_NAME - TRUSTING_DOMAIN_DNS_VA...	ROLE_POWER_BINDINGS - ID - POWER_NAME
--	--	---	---	--	---

VA_CLASS

VIRTUAL_ATTRIBUTE_ID

CLASS_ID

VIRTUAL_ATTRIBUTE

VIRTUAL_ATTRIBUTE_ID

McNameValue

ValueType

McClass

IsMultiValued

DisplayName

Description

Enabled

DomainsIDToName

ID

Name

ADDynamicGroupMember

groupGUID

memberGUID

isWhitelst

CUSTOM_TOOLS

CUSTOM_TOOL_ID

Name

SupportedObjects

Status

Description

Comment

AppLocation

AppParameters

AppDir

AppWinInterface

AppCloseOnExit

MenuStructure

ADVANCE_QUERIES

ADVANCE_QUERY_ID

QueryName

QueryAuthor

QueryBaseContainer

QueryColumnList

QueryScope

VQueryFilter

QueryFilter

isPublic

QueryColumnType

COMPUTERS

objectGuid

cn

Domain_d

objectClass

whenCreated

whenChanged

whenCreated

whenChanged

manager

managerObject

managerBy

mail

givenName

displayName

comment

adminDisplayName

adminDescription

sAMAccountName

AccountDisabled

AccountLockedOut

AccountNotDelegated

DontRequireAuth

EncryptedTextPasswordAllowed

FpnmEnabled

HomeDirectoryRequired

InterDomainTrustAccount

NormalAccount

PasswordDoesNotExpire

PasswordExpires

PasswordNotRequired

Script

ServerTrustAccount

SmartCardRequired

TempDuplicateAccount

TrustedForDelegation

TrustedToAuthenticateForDele...

UseDesKeyOnly

UsesAccountControl

WorkstationTrustAccount

objectCategory

operatingSystemVersion

operatingSystemServicePack

operatingSystemHotfix

operatingSystem

objectVersion

name

memberOf

description

ACTIVEVIEWS

ACTIVEVIEW_ID

McNameValue

Comment

McParent

McPath

McScheme

Description

NumAssocDeputies

NumRules

McName

BuiltIn

Enabled

ACCOUNT_RULE_BINDING_STATUS

RULE_ACC_BIND_ID

HAS_MCRE_OBJECTS

ASSISTANT_ADMIN_ROLE_BINDINGS

ID

ASSOC_ID

AV_ID

ASSISTANT_ADMIN_POWER_BINDINGS

ID

ASSOC_ID

AV_ID

DEVICES
SYSTEM_ID
DOMAIN_ID
DisplayName
ExitCode
ServiceSpecificCode
StartMode
State
SystemName

DEVICE_STATUS
STATUS_TYPE_ID
STATUS_TYPE

CONTACTS
objectGuid
cn
Dcnair_Id
usrCreated
objectClass
usrChanged
whenCreated
whenChanged
description
comment
ou
name
manager
managedObjects
mailNickname
mail
displayName
objectCategory
distinguishedName
CONTACTDNCH-KSJM

CONTEXTTABLEFOROU
UDLEVEL
UDOBJECT
UDOBJECTPARENT
UDOBJECTNAME
UDOBJECTUNAME
UDOUTPUT
UDICON
UDSORTORDER
DOMAINID
PFCECESSFLAG
ISMNAGED

CLONE_EXCEPTION
CLONE_EXCEPTION_ID
Name
Description
DisplayName

ASSISTANT_ADMINS
ASSISTANT_ADMIN_ID
McsNameValue
Comment
McsName
McsParent
McsPath
McsScheme
Description
NumAssociativeViews
NumRules

ADDynamicGroup
groupGUID
memberFilterGUID
rootMemberGUID
ldapFilter
includeChildContainers

ACTIVEVIEW_ROLE_BINDINGS
IC
ASSOC_ID

ActivitySummary
Date
utcatetime
Hour
Action
ObjectType
EventType
ReturnCode
EventDescriptor
OperationName
dsaServerGUID
WhatDomain
WhoUserGUID
SourceObjectName
SourceObjectType
SourceObjectGUID
SourceObjectContainerName
SourceObjectContainerType
SourceObjectContainerGUID
TargetObjectName
TargetObjectType
TargetObjectGUID
TargetObjectContainerName
TargetObjectContainerType
TargetObjectContainerGUID
DestinationObjectName
DestinationObjectType
DestinationObjectGUID
DestinationObjectContainerName
DestinationObjectContainerType
DestinationObjectContainerGUID
OperationCategory
IsAutomatic
EventCategoryText
OperationType

ACCOUNT_RULE_BINDINGS
RULE_ACC_BIND_ID
objectGuid

ACTIVEVIEW_ACCOUNTS
ACTIVEVIEW_ID
objectGuid

AD_Attribute_size
Attribute_Name
Attribute_Size

DOMAIN_CONTROLLERS
DOMCTRL_ID
Domain_Id
McsNameValue
McsPath

ActionContextList
Action

CLASS
CLASS_ID
CLASS
SCHEMA_CLASS

ACCOUNTS
objectGuid
DOMAIN_ID
usrCreated
usrChanged
CLASS_ID
Famnt_GUID
Famnt_DN
Internal_ProcessNextIVCR
closingSheetName
srcAccountName
name
description
FID
Internal_VA_Process

DomainContextList
DomainName

DEVICE_STARTUP
STARTUP_TYPE_ID
STARTUP_TYPE

ExchangeMgtToolsInstalled	
ID	
Name	

EXCHANGE_PROXY	
Domain	
ProxyFernalString	

EVENTLOGS	
SYSTEM_ID	
DOMAIN_ID	
CreationDate	
LogfileName	
CSName	
FileSize	
MaxFileSize	
OverwriteOutDated	
OverWritePolicy	

EXCHANGE_CONFIG	
EXCHANGE_CONFIG_ID	
McsClass	
McsPath	
McsSchema	
AliasNamingConvention	
AliasNamingConventionFlags	
DisplayNameNamingConvention	
DisplayNameNamingConventionFL...	
DisplayExch2KSupportPolicy	
DisplayExch2007SupportPolicy	
EnforceAliasNamingConvention	
EnforceExch2KAliasNamingCo...	
Exch2KAliasNamingConvention	
Exch2KAliasNamingConventio...	
IsExch2KEnabled	
IsExch2KSupportInstalled	
IsExch2007Enabled	
IsExch2007SupportInstalled	
MailboxPolicyDuringCreate	
MailboxPolicyDuringDelete	
MailboxPolicyDuringUpdate	
ExchangeToolsInstalled	
IsExch2010Enabled	
IsExch2010SupportInstalled	
IsExch2013Enabled	
IsExch2013SupportInstalled	

EVENTLOG_WRAP_TYPE	
OverWritePolicy_ID	
OverWritePolicy	

EVENT_TYPE	
EVENT_TYPE_ID	
EVENT_TYPE	

DynamicDistributionGroups	
objectGuid	
cn	
displayName	
distinguishedName	
Domain_Id	
objectClass	
mail	
MailboxVersion	
mailNickname	
msExchDynamicDLBaseDN	
msExchDynamicDLFilter	
msExchQueryFilter	
msExchQueryFilterMetadata	
name	
objectCategory	
usnCreated	
usnChanged	
whenChanged	
whenCreated	

DOMCTRL_TYPE	
DOMCTRL_TYPE_ID	
DOMCTRL_TYPE	

ExchangeVersion_Name	
ID	
Name	

GROUPS	
objectGuid	
groupType	
Domain_Id	
objectClass	
usnCreated	
usnChanged	
whenCreated	
whenChanged	
description	
displayName	
name	
sAMAccountName	
objectCategory	
memberOf	
cn	
member	
managedObjects	
managedBy	
mail	
mailNickname	
lastKnownParent	
isDynamicGroup	
groupAttributes	

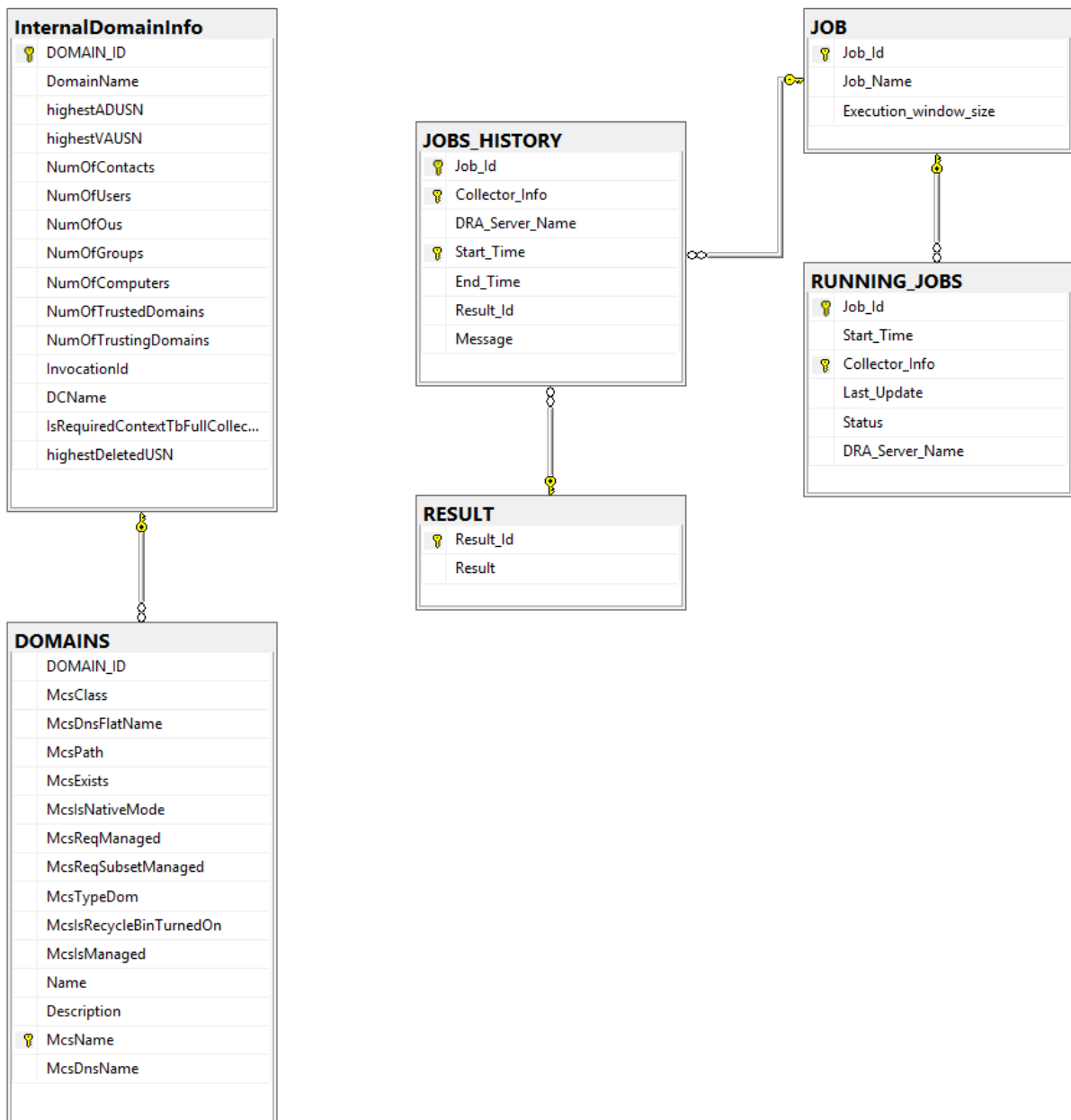
GuidNameMap	
ObjectGUID	
ObjectName	
ObjectType	

GroupMembershipByDN	
GroupDomainId	
GroupGUID	
MemberDN	
MemberDomainId	

EXPORT_ERRORS	
EXPORT_SEQ	
CLASS	
CODE	
MESSAGE	
EA	

Group_Membership	
Group_GUID	
Member_GUID	
DOMAIN_ID	

IMPORT_CATEGORIES	
CATEGORY_ID	
CATEGORY	
CATEGORY_GROUP_ID	
SEQUENCE	



Tables

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
ACCOUNT_RULE_BINDING_STATUS	HAS_MORE_OBJECTS	bit		Yes
	RULE_ACC_BIND_ID	int		No
ACCOUNT_RULE_BINDINGS	objectGuid	uniqueidentifier		Yes
	RULE_ACC_BIND_ID	int		No
ACCOUNTS	CLASS_ID	int		Yes
	description	nvarchar	1024	Yes
	distinguishedName	nvarchar	-1	Yes
	DOMAIN_ID	int		Yes
	Internal_ProcessNextINCR	bit		Yes
	Internal_VA_Process	bit		Yes
	name	nvarchar	255	Yes
	objectGuid	uniqueidentifier		No
	Parent_DN	nvarchar	-1	Yes
	Parent_GUID	uniqueidentifier		Yes
	RID	int		Yes
	samAccountName	nvarchar	255	Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
ActionContextList	Action	nvarchar	50	No
ACTIVEVIEW_ACCOUNTS	ACTIVEVIEW_ID	int		No
	objectGuid	uniqueidentifier		Yes
ACTIVEVIEW_ASSISTANT_ADMIN_BINDINGS	ASSOC_ID	int		Yes
	ID	int		No
ACTIVEVIEW_ROLE_BINDINGS	ASSOC_ID	int		Yes
	ID	int		No
ACTIVEVIEWS	ACTIVEVIEW_ID	int		No
	BuiltIn	int		Yes
	Comment	ntext	1073741823	Yes
	Description	ntext	1073741823	Yes
	Enabled	bit		Yes
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	Yes
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	NumAssocDeputies	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
ActivityCounts	NumRules	int		Yes
	Action	nvarchar	64	Yes
	Count	int		Yes
	Date	datetime		Yes
	DestinationObjectContainerName	nvarchar	255	Yes
	DestinationObjectContainerType	nvarchar	100	Yes
	draServerGUID	uniqueidentifier		Yes
	EventType	int		Yes
	Hour	int		Yes
	ObjectType	nvarchar	50	Yes
	OperationName	nvarchar	100	Yes
	TargetObjectContainerName	nvarchar	255	Yes
	TargetObjectContainerType	nvarchar	100	Yes
	WhatDomain	nvarchar	250	Yes
	WhoUserGUID	uniqueidentifier		Yes
ActivitySummary	Action	nvarchar	64	Yes
	Date	datetime		Yes
	DestinationObjectContainerGUID	uniqueidentifier		Yes
	DestinationObjectContainerName	nvarchar	255	Yes
	DestinationObjectContainerType	nvarchar	100	Yes
	DestinationObjectGUID	uniqueidentifier		Yes
	DestinationObjectName	nvarchar	255	Yes
	DestinationObjectType	nvarchar	255	Yes
	draServerGUID	uniqueidentifier		Yes
	EventCategoryText	nvarchar	50	Yes
	EventDescription	nvarchar	512	Yes
	EventType	int		Yes
	Hour	int		Yes
	IsAutomatic	bit		Yes
	ObjectType	nvarchar	50	Yes
	OperationCategory	nvarchar	50	Yes
	OperationName	nvarchar	100	Yes
	OperationType	int		Yes
	ReturnCode	int		Yes
	SourceObjectContainerGUID	uniqueidentifier		Yes
	SourceObjectContainerName	nvarchar	255	Yes
	SourceObjectContainerType	nvarchar	100	Yes
	SourceObjectGUID	uniqueidentifier		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	SourceObjectName	nvarchar	255	Yes
	SourceObjectType	nvarchar	255	Yes
	TargetObjectContainerGUID	uniqueidentifier		Yes
	TargetObjectContainerName	nvarchar	255	Yes
	TargetObjectContainerType	nvarchar	100	Yes
	TargetObjectGUID	uniqueidentifier		Yes
	TargetObjectName	nvarchar	255	Yes
	TargetObjectType	nvarchar	100	Yes
	utcdatetime	datetime		Yes
	WhatDomain	nvarchar	250	Yes
	WhoUserGUID	uniqueidentifier		Yes
AD_Attribute_size	Attribute_Name	nvarchar	255	Yes
	Attribute_Size	int		Yes
ADDynamicGroup	groupGUID	uniqueidentifier		No
	includeChildContainers	bit		No
	IdapFilter	ntext	1073741823	No
	memberFilterGUID	uniqueidentifier		No
	rootMemberGUID	uniqueidentifier		No
ADDynamicGroupMember	groupGUID	uniqueidentifier		No
	isWhiteList	bit		No
	memberGUID	uniqueidentifier		No
ADVANCE_QUERIES	ADVANCE_QUERIES_ID	int		No
	isPublic	nvarchar	20	Yes
	QueryAuthor	nvarchar	-1	Yes
	QueryBaseContainer	nvarchar	-1	Yes
	QueryColumnsList	nvarchar	-1	Yes
	QueryColumnsType	int		Yes
	QueryFilter	nvarchar	-1	Yes
	QueryName	nvarchar	-1	Yes
	QueryScope	int		Yes
	VAQueryFilter	nvarchar	-1	No
ASSISTANT_ADMIN_POWER_BINDINGS	ASSOC_ID	int		Yes
	AV_ID	int		Yes
	ID	int		No
ASSISTANT_ADMIN_ROLE_BINDINGS	ASSOC_ID	int		Yes
	AV_ID	int		Yes
	ID	int		No
ASSISTANT_ADMINS	ASSISTANT_ADMIN_ID	int		No

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	Comment	ntext	1073741823	Yes
	Description	ntext	1073741823	Yes
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	Yes
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	NumAssocActiveViews	int		Yes
	NumRules	int		Yes
CLASS	CLASS	nvarchar	-1	No
	CLASS_ID	int		No
	SCHEMA_CLASS	nvarchar	-1	No
CLONE_EXCEPTION	CLONE_EXCEPTION_ID	int		No
	Description	nvarchar	2048	Yes
	DisplayName	nvarchar	512	Yes
	Name	nvarchar	50	Yes
COMPUTERS	AccountDisabled	int		Yes
	AccountLockedOut	int		Yes
	AccountNotDelegated	int		Yes
	adminDescription	nvarchar	2048	Yes
	adminDisplayName	nvarchar	512	Yes
	cn	nvarchar	128	Yes
	comment	nvarchar	2048	Yes
	description	nvarchar	2048	Yes
	displayName	nvarchar	512	Yes
	Domain_Id	int		Yes
	DontRequirePreauth	int		Yes
	EncryptedTextPasswordAllowed	int		Yes
	FpnwEnabled	int		Yes
	givenName	nvarchar	128	Yes
	HomeDirectoryRequired	int		Yes
	InterDomainTrustAccount	int		Yes
	mail	nvarchar	512	Yes
	managedBy	nvarchar	-1	Yes
	managedObjects	nvarchar	-1	Yes
	manager	nvarchar	-1	Yes
	memberOf	nvarchar	-1	Yes
	name	nvarchar	255	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	NormalAccount	int		Yes
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	objectVersion	int		Yes
	operatingSystem	nvarchar	-1	Yes
	operatingSystemHotfix	nvarchar	-1	Yes
	operatingSystemServicePack	nvarchar	-1	Yes
	operatingSystemVersion	nvarchar	-1	Yes
	PasswordDoesNotExpire	int		Yes
	PasswordExpired	int		Yes
	PasswordNotRequired	int		Yes
	sAMAccountName	nvarchar	255	Yes
	Script	int		Yes
	ServerTrustAccount	int		Yes
	SmartCardRequired	int		Yes
	TempDuplicateAccount	int		Yes
	TrustedForDelegation	int		Yes
	TrustedToAuthenticateForDelegation	int		Yes
	UseDesKeyOnly	int		Yes
	UserAccountControl	int		Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
	WorkstationTrustAccount	int		Yes
CONTACTS	cn	nvarchar	128	Yes
	comment	nvarchar	2048	Yes
	CONTACTDNCHKSUM	int		Yes
	description	nvarchar	2048	Yes
	displayName	nvarchar	512	Yes
	distinguishedName	nvarchar	-1	Yes
	Domain_Id	int		Yes
	mail	nvarchar	512	Yes
	mailNickname	nvarchar	-1	Yes
	managedObjects	nvarchar	-1	Yes
	manager	nvarchar	-1	Yes
	name	nvarchar	255	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	ou	nvarchar	128	Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
CONTEXTTABLEFOROU	DOMAINID	int		Yes
	ISMANAGED	int		Yes
	PROCESSFLAG	bit		Yes
	UDICON	int		Yes
	UDLEVEL	int		Yes
	UDOBJECT	nvarchar	-1	Yes
	UDOBJECTNAME	nvarchar	-1	Yes
	UDOBJECTPARENT	nvarchar	-1	Yes
	UDOBJECTUINAME	nvarchar	-1	Yes
	UDOUTPUT	nvarchar	-1	Yes
	UDSORTORDER	int		Yes
CUSTOM_TOOLS	AppCloseOnExit	bit		Yes
	AppDir	nvarchar	-1	Yes
	AppLocation	nvarchar	-1	Yes
	AppParameters	nvarchar	-1	Yes
	AppWinInterface	bit		Yes
	Comment	ntext	1073741823	Yes
	CUSTOM_TOOL_ID	int		No
	Description	ntext	1073741823	Yes
	MenuStructure	nvarchar	-1	Yes
	Name	nvarchar	-1	Yes
	Status	nvarchar	10	Yes
	SupportedObjects	nvarchar	120	Yes
DEVICE_STARTUP	STARTUP_TYPE	nvarchar	10	Yes
	STARTUP_TYPE_ID	smallint		No
DEVICE_STATUS	STATUS_TYPE	nvarchar	20	Yes
	STATUS_TYPE_ID	int		No
DEVICES	DisplayName	nvarchar	255	Yes
	DOMAIN_ID	int		No
	ExitCode	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	ServiceSpecificExitCode	int		Yes
	StartMode	int		Yes
	State	int		Yes
	SYSTEM_ID	uniqueidentifier		No
	SystemName	nvarchar	255	Yes
DOMAIN_CONTROLLERS	Domain_Id	int		Yes
	DOMCTRL_ID	int		No
	McsNameValue	nvarchar	255	Yes
	McsPath	nvarchar	255	Yes
DomainContextList	DomainName	nvarchar	250	No
DOMAINS	Description	ntext	1073741823	Yes
	DOMAIN_ID	int		No
	McsClass	nvarchar	255	Yes
	McsDnsFlatName	nvarchar	255	Yes
	McsDnsName	nvarchar	255	Yes
	McsExists	bit		Yes
	McsIsManaged	bit		Yes
	McsIsNativeMode	bit		Yes
	McsIsRecycleBinTurnedOn	bit		Yes
	McsName	nvarchar	255	No
	McsPath	nvarchar	255	Yes
	McsReqManaged	bit		Yes
	McsReqSubsetManaged	bit		Yes
	McsTypeDom	int		Yes
	Name	nvarchar	255	Yes
DomainsIDToName	ID	int		No
	Name	nvarchar	50	Yes
DOMCTRL_TYPE	DOMCTRL_TYPE	nvarchar	50	Yes
	DOMCTRL_TYPE_ID	int		No
draview_AA_GuidNameMap	ObjectGUID	uniqueidentifier		No
	ObjectName	nvarchar	255	No
draview_Activeview_Accounts	ACCOUNT_DESCRIPTION	nvarchar	1024	Yes
	ACCOUNT_FULLNAME	nvarchar	255	Yes
	ACCOUNT_NAME	nvarchar	255	Yes
	ACCOUNT_TYPE	nvarchar	-1	No
	ACTIVEVIEW_NAME	nvarchar	255	Yes
	DOMAIN_NAME	nvarchar	255	Yes
	ISMANAGED	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	McsIsManaged	bit		Yes
	MCSREQSUBSETMANAGED	bit		Yes
draview_Server_GuidNameMap	ObjectGUID	uniqueidentifier		No
	ObjectName	nvarchar	255	No
DynamicDistributionGroups	cn	nvarchar	512	Yes
	displayName	nvarchar	512	Yes
	distinguishedName	nvarchar	-1	Yes
	Domain_Id	int		Yes
	mail	nvarchar	512	Yes
	MailboxVersion	int		Yes
	mailNickname	nvarchar	-1	Yes
	msExchDynamicDLBaseDN	nvarchar	-1	Yes
	msExchDynamicDLFilter	nvarchar	-1	Yes
	msExchQueryFilter	nvarchar	-1	Yes
	msExchQueryFilterMetadata	nvarchar	-1	Yes
	name	nvarchar	512	Yes
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
EVENT_TYPE	EVENT_TYPE	nvarchar	15	Yes
	EVENT_TYPE_ID	int		No
EVENTLOG_WRAP_TYPE	OverWritePolicy	nvarchar	50	Yes
	OverWritePolicy_ID	smallint		No
EVENTLOGS	CreationDate	datetime		Yes
	CSName	nvarchar	-1	Yes
	DOMAIN_ID	int		No
	FileSize	int		Yes
	LogfileName	nvarchar	-1	Yes
	MaxFileSize	int		Yes
	OverwriteOutDated	int		Yes
	OverWritePolicy	int		Yes
	SYSTEM_ID	uniqueidentifier		No
EXCHANGE_CONFIG	AliasNamingConvention	nvarchar	255	Yes
	AliasNamingConventionFlags	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	DirNameNamingConvention	nvarchar	255	Yes
	DirNameNamingConventionFlags	int		Yes
	DisplayNameNamingConvention	nvarchar	255	Yes
	DisplayNameNamingConventionFlags	int		Yes
	EnableExch2007SupportPolicy	int		Yes
	EnableExch2KSupportPolicy	int		Yes
	EnforceAliasNamingConvention	int		Yes
	EnforceExch2KAliasNamingConvention	int		Yes
	Exch2KAliasNamingConvention	nvarchar	255	Yes
	Exch2KAliasNamingConventionFlags	int		Yes
	EXCHANGE_CONFIG_ID	int		No
	ExchangeToolsInstalled	int		Yes
	IsExch2007Enabled	int		Yes
	IsExch2007SupportInstalled	int		Yes
	IsExch2010Enabled	int		Yes
	IsExch2010SupportInstalled	int		Yes
	IsExch2013Enabled	int		Yes
	IsExch2013SupportInstalled	int		Yes
	IsExch2KEnabled	int		Yes
	IsExch2KSupportInstalled	int		Yes
	MailboxPolicyDuringCreate	int		Yes
	MailboxPolicyDuringDelete	int		Yes
	MailboxPolicyDuringUpdate	int		Yes
	McsClass	nvarchar	255	Yes
	McsPath	nvarchar	255	Yes
	McsSchema	ntext	1073741823	Yes
EXCHANGE_PROXY	Domain	nvarchar	150	Yes
	ProxyFormatString	nvarchar	-1	Yes
ExchangeMgtToolsInstalled	ID	int		Yes
	Name	nvarchar	50	Yes
ExchangeVersion_Name	ID	int		Yes
	Name	nvarchar	50	Yes
EXPORT_ERRORS	CLASS	nvarchar	30	Yes
	CODE	int		Yes
	EA	nvarchar	255	Yes
	EXPORT_SEQ	int		No
	MESSAGE	ntext	1073741823	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
Group_Membership	DOMAIN_ID	int		Yes
	Group_GUID	uniqueidentifier		No
	Member_GUID	uniqueidentifier		Yes
GroupMembershipByDN	GroupDomainId	int		No
	GroupGUID	uniqueidentifier		No
	MemberDN	nvarchar	-1	No
	MemberDomainId	int		No
GROUPS	cn	nvarchar	128	Yes
	description	nvarchar	2048	Yes
	displayName	nvarchar	512	Yes
	Domain_Id	int		Yes
	groupAttributes	int		Yes
	grouptype	int		Yes
	isDynamicGroup	bit		Yes
	lastKnownParent	nvarchar	-1	Yes
	mail	nvarchar	512	Yes
	mailNickname	nvarchar	-1	Yes
	managedBy	nvarchar	-1	Yes
	managedObjects	nvarchar	-1	Yes
	member	nvarchar	-1	Yes
	memberOf	nvarchar	-1	Yes
	name	nvarchar	255	Yes
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	sAMAccountName	nvarchar	255	Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
GuidNameMap	ObjectGUID	uniqueidentifier		No
	ObjectName	nvarchar	255	No
	ObjectType	nvarchar	50	Yes
IMPORT_CATEGORIES	CATEGORY	nvarchar	30	Yes
	CATEGORY_GROUP_ID	smallint		Yes
	CATEGORY_ID	smallint		No
	SEQUENCE	smallint		Yes
IMPORT_CATEGORY_GROUPS	CATEGORY_GROUP	nvarchar	50	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
IMPORT_ERRORS	CATEGORY_GROUP_ID	smallint		No
	CLASS	nvarchar	200	Yes
	CODE	int		Yes
	EA	ntext	1073741823	Yes
	IMPORT_SEQ	int		No
INTERNAL_ActivitySummary	MESSAGE	ntext	1073741823	Yes
	Action	nvarchar	64	Yes
	Date	datetime		Yes
	DestinationObjectContainerGUID	uniqueidentifier		Yes
	DestinationObjectContainerName	nvarchar	255	Yes
	DestinationObjectContainerType	nvarchar	100	Yes
	DestinationObjectGUID	uniqueidentifier		Yes
	DestinationObjectName	nvarchar	255	Yes
	DestinationObjectType	nvarchar	255	Yes
	draServerGUID	uniqueidentifier		Yes
	EventCategoryText	nvarchar	50	Yes
	EventDescription	nvarchar	512	Yes
	EventType	int		Yes
	Hour	int		Yes
	IsAutomatic	bit		Yes
	ObjectType	nvarchar	50	Yes
	OperationCategory	nvarchar	50	Yes
	OperationName	nvarchar	100	Yes
	OperationType	int		Yes
	ReturnCode	int		Yes
	SourceObjectContainerGUID	uniqueidentifier		Yes
	SourceObjectContainerName	nvarchar	255	Yes
	SourceObjectContainerType	nvarchar	100	Yes
	SourceObjectGUID	uniqueidentifier		Yes
	SourceObjectName	nvarchar	255	Yes
	SourceObjectType	nvarchar	255	Yes
	TargetObjectContainerGUID	uniqueidentifier		Yes
	TargetObjectContainerName	nvarchar	255	Yes
	TargetObjectContainerType	nvarchar	100	Yes
	TargetObjectGUID	uniqueidentifier		Yes
	TargetObjectName	nvarchar	255	Yes
	TargetObjectType	nvarchar	100	Yes
	utcdatetime	datetime		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	WhatDomain	nvarchar	250	Yes
	where_drserver_name	nvarchar	255	Yes
	Who_Name	nvarchar	255	Yes
	WhoUserGUID	uniqueidentifier		Yes
INTERNAL_GuidNameMap	ObjectGUID	uniqueidentifier		No
	ObjectName	nvarchar	255	No
	ObjectType	nvarchar	50	Yes
INTERNAL_ProcessMovedorRenamedContainers	containerDN	nvarchar	-1	No
	containerGUID	uniqueidentifier		No
	DOMAIN_ID	int		Yes
	IsProcessed	bit		Yes
InternalDomainInfo	DCName	nvarchar	255	Yes
	DOMAIN_ID	int		No
	DomainName	nvarchar	-1	No
	highestADUSN	bigint		Yes
	highestDeletedUSN	bigint		Yes
	highestVAUSN	bigint		Yes
	InvocationId	binary	16	Yes
	IsRequiredContextTbFullCollection	bit		Yes
	NumOfComputers	bigint		Yes
	NumOfContacts	bigint		Yes
	NumOfGroups	bigint		Yes
	NumOfOus	bigint		Yes
	NumOfTrustedDomains	bigint		Yes
	NumOfTrustingDomains	bigint		Yes
	NumOfUsers	bigint		Yes
InternalIOUStructure	CONTAINER_GUID	uniqueidentifier		No
	DOMAIN_ID	int		No
JOB	Execution_window_size	int		Yes
	Job_Id	int		No
	Job_Name	nvarchar	50	Yes
JOBS_HISTORY	Collector_Info	nvarchar	400	No
	DRA_Server_Name	nvarchar	400	No
	End_Time	datetime		Yes
	Job_Id	int		No
	Message	ntext	1073741823	Yes
	Result_Id	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
LAST_COLLECTED_ATTRIBUTES	Start_Time	datetime		No
	ClassId	int		No
	DataType	int		No
	DomainId	int		No
	IsVirtual	bit		No
LICENSES	Name	nvarchar	50	No
	Comment	ntext	1073741823	Yes
	CompanyId	nvarchar	255	Yes
	CompanyName	nvarchar	255	Yes
	DisplayName	nvarchar	150	Yes
	FinalDay	datetime		Yes
	Flags	int		Yes
	GraceExpirationDate	datetime		Yes
	GracePeriodLength	int		Yes
	IsNodeLockOn	bit		Yes
	IsShared	bit		Yes
	IsUpgradeOnly	bit		Yes
	LICENSE_ID	int		No
	LicensedBy	int		Yes
	LicenseType	int		Yes
	NumberOfAgents	int		Yes
	NumberOfComputers	int		Yes
	NumberOfMailboxes	int		Yes
	NumberOfOperations	int		Yes
	NumberOfRecords	int		Yes
	NumberOfServers	int		Yes
	NumberOfUsers	int		Yes
	NumberOfWorkstations	int		Yes
	OverdraftPercentage	int		Yes
	OverdraftValue	int		Yes
	PackageComment	nvarchar	255	Yes
	Packageld	int		Yes
	PackageName	nvarchar	255	Yes
	PackageVersion	nvarchar	50	Yes
	SerialNumber	nvarchar	255	Yes
	SupportsAllOperations	bit		Yes
	TrialExpirationDate	datetime		Yes
	TrialPeriodLength	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	TrialStartDate	datetime		Yes
MANAGED_DOMAINS	ID	int		No
	McsDnsName	nvarchar	255	Yes
ManagedContainers	DOMAIN_ID	int		Yes
	objectGUID	uniqueidentifier		Yes
MEMBERS	GROUP_ID	int		Yes
	MEMBER_ID	int		Yes
MMCACHE	CacheName	nvarchar	255	No
	PrimaryName	nvarchar	255	Yes
	RefreshStatus	int		Yes
	RefreshTime	datetime		Yes
	SecondaryName	nvarchar	255	No
MMREPLICATION	CacheName	nvarchar	255	No
	LastSuccess	datetime		Yes
	PrimaryName	nvarchar	255	Yes
	ReplicationStatus	int		Yes
	ReplicationTime	datetime		Yes
	SecondaryName	nvarchar	255	No
MMSERVERS	AccessAccount	nvarchar	255	Yes
	AccessStatus	int		Yes
	AccessTime	datetime		Yes
	LastSuccess	datetime		Yes
	Mode	int		Yes
	PrimaryName	nvarchar	255	Yes
	ServerName	nvarchar	255	No
MODPROPS	LoginHours	int		Yes
	LoginScript	int		Yes
	LoginWorkstations	int		Yes
	ModifiedProperties	ntext	1073741823	Yes
	RasPermissions	int		Yes
	RecordNumber	int		No
	samAccountName	nvarchar	255	Yes
	WtsHomeDir	int		Yes
	WtsProfilePath	int		Yes
MP_UPDATES	AccountDisabled	bit		No
	AccountExpires	bit		No
	ClientName	nvarchar	255	No
	Comment	bit		No

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	FullName	bit		No
	Lock	bit		No
	NetWare	bit		No
	PasswordChange	bit		No
	PasswordExpire	bit		No
	PasswordModify	bit		No
	PasswordReq	bit		No
	recordNumber	int		Yes
	UserName	bit		No
ObjectTypeContextList	ObjectType	nvarchar	50	Yes
OPTIONSTABLE	DOMAINID	int		Yes
	OPTIONNAME	nvarchar	255	Yes
	VALUE	nvarchar	-1	Yes
OUS	cn	nvarchar	128	Yes
	co	nvarchar	256	Yes
	description	nvarchar	2048	Yes
	displayName	nvarchar	512	Yes
	distinguishedName	nvarchar	-1	Yes
	DNCHKSUM	int		Yes
	Domain_Id	int		Yes
	lastKnownParent	nvarchar	-1	Yes
	managedBy	nvarchar	-1	Yes
	managedObjects	nvarchar	-1	Yes
	memberOf	nvarchar	-1	Yes
	name	nvarchar	255	Yes
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	objectVersion	int		Yes
	ou	nvarchar	128	Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
PASSWORD_POLICIES	EnablePasswordPolicy	bit		Yes
	MaxChars	int		Yes
	MaxConsecutiveAlpha	int		Yes
	MaxDigits	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	MaxLowerChars	int		Yes
	MaxPasswordLength	int		Yes
	MaxSameCharRepeat	int		Yes
	MaxSpecial	int		Yes
	MaxUpperChars	int		Yes
	MinChars	int		Yes
	MinDigits	int		Yes
	MinLowerChars	int		Yes
	MinPasswordLength	int		Yes
	MinSpecial	int		Yes
	MinUpperChars	int		Yes
POLICIES	Comment	ntext	1073741823	Yes
	Description	ntext	1073741823	Yes
	Enabled	bit		No
	LastModified	nvarchar	50	Yes
	McsClass	nvarchar	255	Yes
	McsIsContainer	bit		No
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	No
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	POLICIES_ID	int		No
POWERS	Comment	ntext	1073741823	Yes
	depth	int		Yes
	description	ntext	1073741823	Yes
	Path	nvarchar	255	Yes
	Power	nvarchar	255	No
	POWER_ID	int		No
PRINTER_STATUS	PRINTER_STATUS	nvarchar	50	Yes
	PRINTER_STATUS_ID	int		Yes
PRINTERS	attributes	int		Yes
	AveragePagesPerMinute	int		Yes
	comment	ntext	1073741823	Yes
	defaultPriority	int		Yes
	DOMAIN_ID	int		No
	driverName	nvarchar	255	Yes
	JobCountSinceLastReset	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	location	nvarchar	255	Yes
	Name	nvarchar	-1	Yes
	parameters	nvarchar	255	Yes
	PortName	nvarchar	255	Yes
	PrinterState	int		Yes
	printProcessor	nvarchar	255	Yes
	priority	int		Yes
	SeparatorFile	nvarchar	255	Yes
	startTime	datetime		Yes
	SYSTEM_ID	uniqueidentifier		No
	systemname	nvarchar	-1	Yes
	untilTime	datetime		Yes
RECYCLE_BIN	Domain	nvarchar	255	Yes
	McsClass	nvarchar	255	Yes
	McsDaysInRecycleBin	int		Yes
	McsFriendlyName	nvarchar	255	Yes
	McsOrigParentOfRecycledObj	ntext	1073741823	Yes
	McsPath	nvarchar	255	Yes
	RecycleBinID	int		No
	samAccountName	nvarchar	255	Yes
RECYCLEBIN	CLASS_ID	int		No
	description	nvarchar	1024	Yes
	DOMAIN_ID	int		No
	New_Name	nvarchar	255	Yes
	objectGUID	uniqueidentifier		No
	Orig_DN	nvarchar	-1	Yes
	Orig_Name	nvarchar	255	Yes
	Orig_ParentDN	nvarchar	-1	Yes
	Orig_ParentGUID	uniqueidentifier		Yes
	sAMAccountName	nvarchar	255	Yes
	uSNChanged	bigint		No
	whenDeleted	datetime		Yes
Resource_Computer	DOMAIN_ID	int		No
	Name	nvarchar	-1	No
	objectGUID	uniqueidentifier		No
RESOURCE_TYPE	RESOURCE	nvarchar	10	Yes
RESULT	Result	nvarchar	50	Yes
	Result_Id	int		No

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
ROLE_BINDINGS	ASSOC_ID	int		Yes
	ID	int		Yes
ROLE_POWER_BINDINGS	ID	int		Yes
	POWER_NAME	nvarchar	255	Yes
ROLES	Comment	ntext	1073741823	Yes
	Creator	nvarchar	255	Yes
	Description	ntext	1073741823	Yes
	Flags	int		Yes
	LastModified	nvarchar	50	Yes
	McsClass	nvarchar	255	Yes
	McsIsContainer	bit		Yes
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	Yes
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	NumAssocDTB	int		Yes
	NumLocalPowers	int		Yes
	NumNestedRoles	int		Yes
	NumOwnerRoles	int		Yes
	ROLES_ID	int		No
RULES	ACTIVEVIEW_ID	nvarchar	150	Yes
	ASSISTANT_ADMIN_ID	nvarchar	150	Yes
	description	ntext	1073741823	Yes
	includeFlag	bit		Yes
	McsClass	nvarchar	255	Yes
	McsIsContainer	bit		Yes
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	Yes
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	nestedFlag	bit		Yes
	RULE_ACC_BIND_ID	int		Yes
	RULE_ID	int		No
	RuleSpecification	nvarchar	255	Yes
	RuleSpecification_AccountFilter	nvarchar	255	Yes
	RuleSpecification_Comment	nvarchar	255	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	RuleSpecification_containerBaseFilter	int		Yes
	RuleSpecification_containerIsFocus	bit		Yes
	RuleSpecification_containerMatchAny	bit		Yes
	RuleSpecification_containerMatchMethod	int		Yes
	RuleSpecification_containerMatchvalue	nvarchar	255	Yes
	RuleSpecification_containerProperty	nvarchar	255	Yes
	RuleSpecification_containerRecursive	bit		Yes
	RuleSpecification_containerSelectBase	bit		Yes
	RuleSpecification_Description	nvarchar	255	Yes
	RuleSpecification_domainMatchValue	nvarchar	255	Yes
	RuleSpecification_domainProperty	nvarchar	255	Yes
	RuleSpecification_groupMemberFilter	int		Yes
	RuleSpecification_groupRecursive	bit		Yes
	RuleSpecification_groupTypeFilter	int		Yes
	RuleSpecification_MatchContainerString	nvarchar	255	Yes
	RuleSpecification_MatchPrinterString	nvarchar	255	Yes
	RuleSpecification_MatchString	nvarchar	255	Yes
	RuleSpecification_memberIsFocus	bit		Yes
	RuleSpecification_Name	nvarchar	255	Yes
	RuleSpecification_normalFilter	int		Yes
	RuleSpecification_objectMatchAny	bit		Yes
	RuleSpecification_objectMatchMethod	int		Yes
	RuleSpecification_objectMatchvalue	nvarchar	255	Yes
	RuleSpecification_objectMustBelongContainer	bit		Yes
	RuleSpecification_objectProperty	nvarchar	255	Yes
	RuleSpecification_objectSelectBase	bit		Yes
	RuleSpecification_OriginalNameOfObject	nvarchar	255	Yes
	RuleSpecification_Power	nvarchar	255	Yes
	RuleSpecification_Recursive	nvarchar	255	Yes
	RuleSpecification_ResourceMailboxType	nvarchar	255	Yes
	RuleSpecification_ResourceMailboxSelect	int		Yes
	RuleSpecification_ResourceType	nvarchar	255	Yes
	RuleSpecification_Rule1	nvarchar	255	Yes
	RuleSpecification_Rule2	nvarchar	255	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	RuleSpecification_SelectBase	nvarchar	255	Yes
	RuleSpecification_SelectMembers	nvarchar	255	Yes
	RuleSpecification_SelectSysObj	bit		Yes
	RuleSpecification_Version	int		Yes
	sourceFlag	bit		Yes
	targetFlag	bit		Yes
RUNNING_JOBS	Collector_Info	nvarchar	400	No
	DRA_Server_Name	nvarchar	400	Yes
	Job_Id	int		No
	Last_Update	datetime		Yes
	Start_Time	datetime		Yes
	Status	nvarchar	20	Yes
SCOPES	McsPolicy	nvarchar	255	No
	McsPolicyPath	nvarchar	255	No
	McsRule	nvarchar	255	No
	McsScopeType	nvarchar	255	No
	SCOPE_ID	int		No
SERVICE_STARTUP	STARTMODE_TYPE	nvarchar	20	Yes
	STARTMODE_TYPE_ID	int		Yes
SERVICE_STATUS	STATE_TYPE	nvarchar	20	No
	STATE_TYPE_ID	int		No
SERVICES	Caption	nvarchar	512	Yes
	Checkpoint	int		Yes
	Description	ntext	1073741823	Yes
	DesktopInteract	bit		Yes
	DisplayName	nvarchar	255	Yes
	DOMAIN_ID	int		No
	ErrorControl	nvarchar	255	Yes
	ExitCode	int		Yes
	PathName	nvarchar	255	Yes
	ServiceSpecificExitCode	bigint		Yes
	ServiceType	nvarchar	150	Yes
	StartMode	int		Yes
	StartName	nvarchar	255	Yes
	State	int		Yes
	SYSTEM_ID	uniqueidentifier		No
	SystemName	nvarchar	255	Yes
	WaitHint	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
SHARE_TYPE	SHARE_TYPE	nvarchar	50	Yes
	SHARE_TYPE_ID	bigint		Yes
SHARES	AllowMaximum	bit		Yes
	Computer	nvarchar	255	Yes
	Description	nvarchar	-1	Yes
	DOMAIN_ID	int		No
	MaximumAllowed	int		Yes
	Name	nvarchar	-1	Yes
	Path	nvarchar	-1	Yes
	SYSTEM_ID	uniqueidentifier		No
	Type	bigint		Yes
STATUS	CollectionStatus	nvarchar	255	Yes
	CollectionType	nvarchar	255	Yes
	Domain	nvarchar	255	No
	EndTime	datetime		Yes
	id	int		No
	StartTime	datetime		Yes
sysdiagrams	definition	varbinary	-1	Yes
	diagram_id	int		No
	name	nvarchar	128	No
	principal_id	int		No
	version	int		Yes
TGA	ASSISTANT_ADMIN	nvarchar	255	Yes
	CREATION_TIME	datetime		Yes
	DESCRIPTION	ntext	1073741823	Yes
	ENABLED	int		Yes
	END_TIME	datetime		Yes
	NAME	nvarchar	255	Yes
	START_TIME	datetime		Yes
	STATE	int		Yes
	STATUS_MESSAGE	ntext	1073741823	Yes
	TARGET_GROUP	nvarchar	-1	Yes
	TGA_ID	int		No
TGA_GROUP_MEMBERSHIP	TGA_ID	int		No
	USER_PATH	nvarchar	-1	Yes
TRIGGER_SCOPES	McsNameValue	nvarchar	255	Yes
	OperationScope	nvarchar	255	Yes
	TRIGGER_ID	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
TRIGGERS	TriggerType	nvarchar	255	Yes
	Action	nvarchar	255	Yes
	CmdLineArgs	nvarchar	255	Yes
	Description	ntext	1073741823	Yes
	Enabled	int		Yes
	ErrorMsg_General	ntext	1073741823	Yes
	ErrorMsg_Undo	ntext	1073741823	Yes
	McsClass	nvarchar	255	Yes
	McsName	nvarchar	255	Yes
	McsNameValue	nvarchar	255	Yes
	McsParent	nvarchar	255	Yes
	McsPath	ntext	1073741823	Yes
	McsSchema	ntext	1073741823	Yes
	Path	nvarchar	255	Yes
	TRIGGER_ID	int		No
	Type	int		Yes
	UndoCmdLineArgs	nvarchar	255	Yes
	UndoPath	nvarchar	255	Yes
TRUSTED_DOMAINS	DOMAIN_NAME	nvarchar	255	Yes
	TRUSTED_DOMAIN_DNS_NAME	nvarchar	255	Yes
	TRUSTED_DOMAIN_NAME	nvarchar	255	Yes
TRUSTING_DOMAINS	DOMAIN_NAME	nvarchar	255	Yes
	TRUSTING_DOMAIN_DNS_NAME	nvarchar	255	Yes
	TRUSTING_DOMAIN_NAME	nvarchar	255	Yes
USER_INTERFACE_EXTENTIONS	McsName	nvarchar	100	No
	McsPath	nvarchar	-1	Yes
	Status	nvarchar	50	Yes
USER_MAILBOXES	Domain	nvarchar	255	Yes
	IMPORT_TIME	real		Yes
	McsPath	ntext	1073741823	Yes
	MiniMB_Alias	nvarchar	68	Yes
	MiniMB_DeletedItemRetention	bit		No
	MiniMB_DeliveryMechanism	nvarchar	50	Yes
	MiniMB_DisplayName	nvarchar	255	Yes
	MiniMB_DistinguishedName	nvarchar	255	Yes
	MiniMB_FirstName	nvarchar	68	Yes
	MiniMB_HomeServer	nvarchar	50	Yes
	MiniMB_HomeSite	nvarchar	50	Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	MiniMB_Initials	nvarchar	8	Yes
	MiniMB_LastName	nvarchar	68	Yes
	MiniMB_MsgIncomingLimit	int		Yes
	MiniMB_MsgOutgoingLimit	int		Yes
	MiniMB_ProhibitSendReceive	int		Yes
	MiniMB_RID	int		Yes
	MiniMB_StoreLimitProhibit	int		Yes
	MiniMB_StoreLimitWarn	int		Yes
	MiniMB_UseISDefaults	bit		No
USERS	AccountDisabled	int		Yes
	accountExpires	datetime		Yes
	AccountLockedOut	int		Yes
	AccountNotDelegated	int		Yes
	canonicalName	nvarchar	-1	Yes
	cn	nvarchar	128	Yes
	description	nvarchar	2048	Yes
	displayName	nvarchar	512	Yes
	Domain_Id	int		Yes
	DontRequirePreauth	int		Yes
	EncryptedTextPasswordAllowed	int		Yes
	FpnwEnabled	int		Yes
	givenName	nvarchar	128	Yes
	HasMailbox	int		Yes
	homeDirectory	nvarchar	-1	Yes
	HomeDirectoryRequired	int		Yes
	homeDrive	nvarchar	-1	Yes
	homeMDB	nvarchar	-1	Yes
	InterDomainTrustAccount	int		Yes
	lastLogonTimestamp	datetime		Yes
	mail	nvarchar	512	Yes
	MailBoxType	int		Yes
	MailboxVersion	int		Yes
	mailNickname	nvarchar	-1	Yes
	managedObjects	nvarchar	-1	Yes
	manager	nvarchar	-1	Yes
	memberOf	nvarchar	-1	Yes
	name	nvarchar	255	Yes
	NormalAccount	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	objectCategory	nvarchar	-1	Yes
	objectClass	nvarchar	-1	Yes
	objectGuid	uniqueidentifier		No
	ou	nvarchar	128	Yes
	PasswordDoesNotExpire	int		Yes
	PasswordExpired	int		Yes
	PasswordNotRequired	int		Yes
	RasPermission	int		Yes
	RasPhonenumber	nvarchar	-1	Yes
	sAMAccountName	nvarchar	255	Yes
	sAMAccountType	int		Yes
	Script	int		Yes
	ServerTrustAccount	int		Yes
	SmartCardRequired	int		Yes
	TempDuplicateAccount	int		Yes
	TrustedForDelegation	int		Yes
	TrustedToAuthenticateForDelegation	int		Yes
	UseDesKeyOnly	int		Yes
	UserAccountControl	int		Yes
	userPrincipalName	nvarchar	-1	Yes
	usnChanged	bigint		Yes
	usnCreated	bigint		Yes
	whenChanged	datetime		Yes
	whenCreated	datetime		Yes
	WorkstationTrustAccount	int		Yes
	WtsAllowLogon	int		Yes
	WtsBrokenTimeoutSettings	int		Yes
	WtsDeviceClientDefaultPrinter	int		Yes
	WtsDeviceClientDrives	int		Yes
	WtsDeviceClientPrinters	int		Yes
	WtsHomeDir	nvarchar	-1	Yes
	WtsHomeDirDrive	nvarchar	-1	Yes
	WtsInheritInitialProgram	int		Yes
	WtsInitialProgram	nvarchar	-1	Yes
	WtsModemCallbackPhoneNumber	nvarchar	-1	Yes
	WtsModemCallbackSettings	int		Yes
	WtsProfilePath	nvarchar	-1	Yes
	WtsReconnectSettings	int		Yes

Table Name	Column Name	DATA_TYPE	Max Character Length	Is Nullable
	WtsRemoteHomeDir	int		Yes
	WtsShadowingSettings	int		Yes
	WtsTimeoutSettingsConnections	int		Yes
	WtsTimeoutSettingsDisconnections	int		Yes
	WtsTimeoutSettingsIdle	int		Yes
	WtsWorkingDirectory	nvarchar	-1	Yes
VA_CLASS	CLASS_ID	int		No
	VIRTUAL_ATTRIBUTE_ID	int		No
VIRTUAL_ATTRIBUTE	Description	ntext	1073741823	Yes
	DisplayName	nvarchar	50	Yes
	Enabled	nchar	10	Yes
	IsMultiValued	nvarchar	20	Yes
	McsClass	nvarchar	-1	Yes
	McsNameValue	nvarchar	50	Yes
	VarType	nchar	10	Yes
	VIRTUAL_ATTRIBUTE_ID	int		No

Stored Procedures

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
CONSTRUCTOUTREE	@OUOBJECTGUID	nvarchar	-1	No	Yes
	@START	int		No	Yes
	@UDLEVEL	int		No	Yes
	@VDOMAINID	int		No	Yes
CONSTRUCTOUTREEINCR	@DomainID	int		No	Yes
DRA_Accounts_HasGuid	@vObjectGUID	uniqueidentifier		No	Yes
DRA_AccountsAndOUs_MergeDomain	@vClassTypeId	int		No	Yes
	@vDomainDN	nvarchar	-1	No	Yes
	@vDomainGUID	uniqueidentifier		No	Yes
	@vDomainId	int		No	Yes
DRA_ADD_DomainToAccountsAndOUs	@vClassType	int		No	Yes
	@vDomainDN	nvarchar	-1	No	Yes
	@vDomainGUID	uniqueidentifier		No	Yes
	@vDomainId	int		No	Yes
DRA_ADDJOB	@vCollector_Info	nvarchar	800	No	Yes
	@vDRA_Server_Name	nvarchar	800	No	Yes
	@vJob_Id	int		No	Yes
	@vLast_Update	datetime		No	Yes
	@vStart_Time	datetime		No	Yes
	@vStatus	nvarchar	40	No	Yes
DRA_ADDMovedorRenamedContainer	@vcontainerDN	nvarchar	-1	No	Yes
	@vcontainerGUID	uniqueidentifier		No	Yes
	@vDomainID	int		No	Yes
DRA_AddToRecycleBin	@vClassType	int		No	Yes
	@vdescription	nvarchar	510	No	Yes
	@vDomainId	int		No	Yes
	@vName	nvarchar	510	No	Yes
	@vObjectGuid	uniqueidentifier		No	Yes
	@vsamAccountName	nvarchar	510	No	Yes
	@vUSNChanged	bigint		No	Yes
DRA_CLEARALLADTABLES	@vDomainId	int		No	Yes
DRA_ClearAllResourceCollectorTables	@vDomainId	int		No	Yes
DRA_ClearGroupMembershipForAGroup	@vGroupGUID	uniqueidentifier		No	Yes

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
DRA_ClearRunningJobs	@DRAServerName	char		No	Yes
	@JobId	int		No	Yes
DRA_ContactGetGUID	@vContactDN	nvarchar	-1	No	Yes
	@vContactName	nvarchar	256	No	Yes
	@vDomainId	int		No	Yes
	@vObjectGUID	uniqueidentifier		Yes	Yes
DRA_DELETE_LAST_COLLECTION_VA_ATTRIBUTES	@vClassId	int		No	Yes
	@vDomainId	int		No	Yes
DRA_DeleteContainersFromInternalTable	@vdomainId	int		No	Yes
DRA_DeleteDynamicGroup	@vGroupGUID	uniqueidentifier		No	Yes
DRA_DeleteDynamicGroupMember	@vGroupGUID	uniqueidentifier		No	Yes
	@vMemberGUID	uniqueidentifier		No	Yes
DRA_DeleteFromGroupMembership	@vmemberGUID	uniqueidentifier		No	Yes
DRA_DeleteFromRecycleBin	@vObjectGuid	uniqueidentifier		No	Yes
DRA_DeleteLastCollectedAttributes	@vClassId	int		No	Yes
	@vDomainId	int		No	Yes
DRA_DeleteManagedContainers	@vDomainId	int		No	Yes
DRA_DOMAINS_HasDomainId	@vDomainId	int		No	Yes
DRA_GetandUpdateParentGUID	@vObjectGUID	uniqueidentifier		No	Yes
DRA_GetClassInfo	@vId	int		Yes	Yes
	@vName	nvarchar	-1	No	Yes
DRA_GetContainerChildren	@vParentGUID	uniqueidentifier		No	Yes
DRA_GetContainersToProcess	@vDomainId	int		No	Yes
DRA_GetDomainId	@vDomainId	int		Yes	Yes
	@vDomainName	nvarchar	-1	No	Yes
DRA_GetDomainIdFromDomains	@vDomainId	int		Yes	Yes
	@vDomainName	nvarchar	-1	No	Yes
DRA_GetDomainInfo	@vId	int		Yes	Yes
	@vName	nvarchar	-1	No	Yes
DRA_GetDuplicateAVRuleInfo	@RuleAccountBindId	int		Yes	Yes
	@RuleId	int		No	Yes
	@RuleSpecification	varchar		No	Yes

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
DRA_GetGroups	@vClassType	int		No	Yes
	@vDomainID	int		No	Yes
	@vLastADUSN	bigint		No	Yes
DRA_GetINCRObjects	@vDomainID	int		No	Yes
	@vLastUSN	bigint		No	Yes
DRA_GetInvocationIdDCName	@vDomainID	int		No	Yes
	@vprevDCName	nvarchar	2048	Yes	Yes
	@vprevInvocationId	binary		Yes	Yes
DRA_GetJobExecutionWindowSize	@vExecutionWindowSize	int		Yes	Yes
	@vJob_Id	int		No	Yes
DRA_GetJobID	@vJob_Id	int		Yes	Yes
	@vJob_Name	nvarchar	100	No	Yes
DRA_GetLastUSN	@ADUSN	bigint		Yes	Yes
	@DeletedUSN	bigint		Yes	Yes
	@vAUSN	bigint		Yes	Yes
	@vDomainName	nvarchar	-1	No	Yes
DRA_GetManagedContainersInfo	@vDomainId	int		No	Yes
DRA_GetObjectsOldDN	@vContainerDN	nvarchar	-1	Yes	Yes
	@vObjectGUID	uniqueidentifier		No	Yes
DRA_GetObjectsToProcessForVA	@vClassTypes	nvarchar	800	No	Yes
	@vDomainID	int		No	Yes
DRA_GetOptionValue	@vDomainId	int		No	Yes
	@vOptionName	nvarchar	-1	No	Yes
	@vValue	nvarchar	-1	Yes	Yes
DRA_GetPreviousINCRGroups	@vDomainID	int		No	Yes
DRA_GetRequiredContextFullCollectionInfo	@vDomainId	int		No	Yes
	@vIsRequiredContextTbFullCollection	bit		Yes	Yes
DRA_GetRunningJobsDetails	@inJobName	nvarchar	100	No	Yes
DRA_GetStartTimeOfJob	@inCollectorInfo	nvarchar	800	No	Yes
	@inJob_Id	int		No	Yes
	@inServerName	nvarchar	800	No	Yes
	@outStartTime	datetime		Yes	Yes
DRA_InsertDynamicGroup	@vGroupGUID	uniqueidentifier		No	Yes

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
	@vIncludeChildContainers	bit		No	Yes
	@vLdapFilter	nvarchar	-1	No	Yes
	@vMemberFilterGUID	uniqueidentifier		No	Yes
	@vRootMemberGUID	uniqueidentifier		No	Yes
DRA_InsertDynamicGroupMember	@vGroupGUID	uniqueidentifier		No	Yes
	@vIsWhiteList	bit		No	Yes
	@vMemberGUID	uniqueidentifier		No	Yes
DRA_InsertInInternalOUStructure	@vDomainId	int		No	Yes
	@vObjectGUID	uniqueidentifier		No	Yes
DRA_InsertMemberByDN	@vGroupDomainID	int		No	Yes
	@vGroupGUID	uniqueidentifier		No	Yes
	@vMemberDN	nvarchar	-1	No	Yes
	@vMemberDomainId	int		No	Yes
DRA_InsertMembers	@vDomainID	int		No	Yes
	@vGroupGUID	uniqueidentifier		No	Yes
	@vMemberGUIDS	nvarchar	-1	No	Yes
DRA_InserttoDomainContextList	@DomainName	nvarchar	500	No	Yes
DRA_INTERNAL_ProcessMovedOrRenamedContainers_Merge	@vContainerDN	nvarchar	-1	No	Yes
	@vContainerGuid	uniqueidentifier		No	Yes
	@vDomainId	int		No	Yes
DRA_InternalOUStructure_Merge	@vDomainId	int		No	Yes
	@vObjectGUID	uniqueidentifier		No	Yes
DRA_IsMemberPresent	@vObjectGUID	uniqueidentifier		No	Yes
	@vOUTGUID	uniqueidentifier		Yes	Yes
DRA_LastCollectedAttributesInfo	@vClassType	int		No	Yes
	@vDomainId	int		No	Yes
DRA_MarkGroupforNextINCR	@vGroupGUID	uniqueidentifier		No	Yes
DRA_MarkGroupProcessed	@vGroupGUID	uniqueidentifier		No	Yes
DRA_MoveToJobHistory	@vCollector_Info	nvarchar	800	No	Yes
	@vDomain_Name	nvarchar	800	No	Yes
	@vEnd_Time	datetime		No	Yes
	@vJob_Id	int		No	Yes
	@vMessage	ntext		No	Yes

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
	@vResult_Id	int		No	Yes
DRA_PopulateDomainSummaryInformation	@vDomainId	int		No	Yes
DRA_PopulateTrustDomainSummaryInformation	@vDomainId	int		No	Yes
DRA_ReadRecycleBinObjects	@vDomainId	int		No	Yes
	@vUSNChanged	bigint		No	Yes
DRA_RecycleBin_Merge	@vClassTypeId	int		No	Yes
	@vdescription	nvarchar	-1	No	Yes
	@vDomainId	int		No	Yes
	@vName	nvarchar	-1	No	Yes
	@vObjectGuid	uniqueidentifier		No	Yes
	@vsamAccountName	nvarchar	-1	No	Yes
	@vUSNChanged	bigint		No	Yes
DRA_Set_Internal_VA_Process	@vObjectGUID	uniqueidentifier		No	Yes
	@vProcess	bit		No	Yes
DRA_SetInvocationIdDCName	@vDCName	nvarchar	2048	No	Yes
	@vDomainID	int		No	Yes
	@vInvocationId	binary		No	Yes
DRA_SetLastUSN	@ADUSN	bigint		No	Yes
	@DELETEDUSN	bigint		No	Yes
	@VAUSN	bigint		No	Yes
	@vDomainName	nvarchar	-1	No	Yes
DRA_SetOptionValue	@vDomainId	int		No	Yes
	@vOptionName	nvarchar	-1	No	Yes
	@vValue	nvarchar	-1	No	Yes
DRA_Table_HasDomainId	@vDomainId	int		No	Yes
	@vTableName	nvarchar	100	No	Yes
DRA_Table_UpdateTypeAndStoredProcedure	@vCreateSP	nvarchar	-1	No	Yes
	@vCreateType	nvarchar	-1	No	Yes
	@vTableName	nvarchar	100	No	Yes
DRA_Update_IsRequiredContextTbFullCollectionValue	@vDomainId	int		No	Yes
	@vIsRequiredContextTbFullCollection	bit		No	Yes
DRA_UpdateContainerAsProcessed	@vcontainerGUID	uniqueidentifier		No	Yes

Stored Procedure name	Parameter name	Parameter Data Type	Parameter Max Bytes	Is Output Parameter	Is Nullable
DRA_UpdateContainerChildrenForNewDN	@vobjectDN	nvarchar	-1	No	Yes
	@vobjectGUID	uniqueidentifier		No	Yes
	@vparentDN	nvarchar	-1	No	Yes
DRA_UpdateContainerDN	@vobjectDN	nvarchar	-1	No	Yes
	@vobjectGUID	uniqueidentifier		No	Yes
DRA_UpdateIsDynamicGroupFlag	@vGroupGUID	uniqueidentifier		No	Yes
	@vIsDynamicGroup	bit		No	Yes
DRA_UpdateJob	@vCollector_Info	nvarchar	100	No	Yes
	@vJob_Id	int		No	Yes
	@vLast_Update	datetime		No	Yes
	@vStatus	nvarchar	40	No	Yes
DRA_UpdateRecycleBin	@vObjectGuid	uniqueidentifier		No	Yes
	@vorigDN	nvarchar	-1	No	Yes
	@vorigName	nvarchar	510	No	Yes
	@vorigOUDN	nvarchar	-1	No	Yes
	@vorigOUGUID	uniqueidentifier		No	Yes
	@vwhenDeleted	datetime		No	Yes
DRA_UpdateToRecycleBin	@vClassType	int		No	Yes
	@vdescription	nvarchar	510	No	Yes
	@vDomainId	int		No	Yes
	@vName	nvarchar	510	No	Yes
	@vObjectGuid	uniqueidentifier		No	Yes
	@vsamAccountName	nvarchar	510	No	Yes
	@vUSNChanged	bigint		No	Yes
SetChildNodes	@OUOBJECTGUID	nvarchar	-1	No	Yes
sp_InsertIntoInternalDomainInfo	@DomainName	nvarchar	-1	No	Yes
	@highestADUSN	bigint		No	Yes
	@highestDeleted	bigint		No	Yes
	@highestVAUSN	bigint		No	Yes
SPLIT_CONTEXT	@COLUMNNAME	nvarchar	-1	No	Yes
	@CONTEXT	nvarchar	-1	No	Yes
	@RESULTSTRING	nvarchar	-1	Yes	Yes
SPLIT_CONTEXT_CHANGES	@CONTEXT1	nvarchar	-1	No	Yes
	@RESULTDRACHANGE	int		Yes	Yes
	@RESULTSTRING	nvarchar	-1	Yes	Yes

Functions

Function Name	Parameter Name	Parameter Data Type	Parameter Max Bytes	Returns Parameter
BIGINTTODATETIME		datetime		True
	@INPUT	bigint		False
fn_diagramobjects		int		True
FN_PARSETEXT2TABLE	@P_SOURCETEXT	nvarchar	-1	False
	@P_DELIMETER	nvarchar	200	False
GET_GUID_TABLE	@CONTEXT	nvarchar	-1	False

Custom Triggers

This section contains examples for custom triggers.

Examples:

Use Case	Context
Collect Additional Virtual Attributes and Extend Object Property Page	• Virtual Attribute • PowerShell Trigger • Post Task Trigger • Exchange Online • Custom Web Client • Property Page
Enable or disable an Exchange Online user's Outlook web access when they are enabled or disabled in Active Directory	• PowerShell Trigger • Post Task Trigger • Exchange Online
Move a user to a new OU when their Office property is changed and change their 'City' attribute to match their new office location	• PowerShell Trigger • PreTask Trigger • Post Task Trigger
Launch Workflow from Trigger	• PreTask Trigger • Workflow

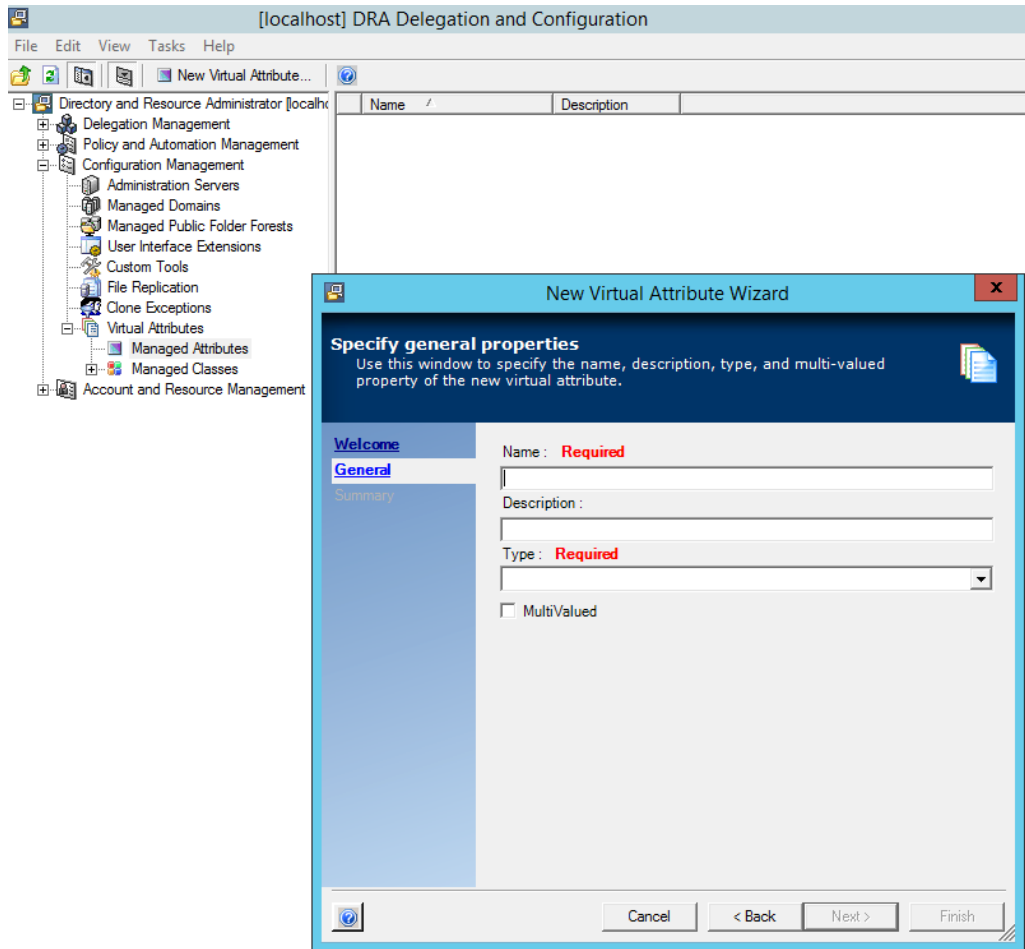
Collect Additional Virtual Attributes and Extend Object Property Page

Scope

This example demonstrates how to define virtual attributes in DRA, collect the virtual attribute values through a post task trigger and add extend an objects property page.

Step-by-step Guide

1. Start by creating a virtual attribute in the Delegation and Configuration Console:
 - On the "Configuration Management" node, navigate to the "Virtual Attributes" section
 - Select the option "New Virtual Attribute" from the "Managed Attributes" subheading

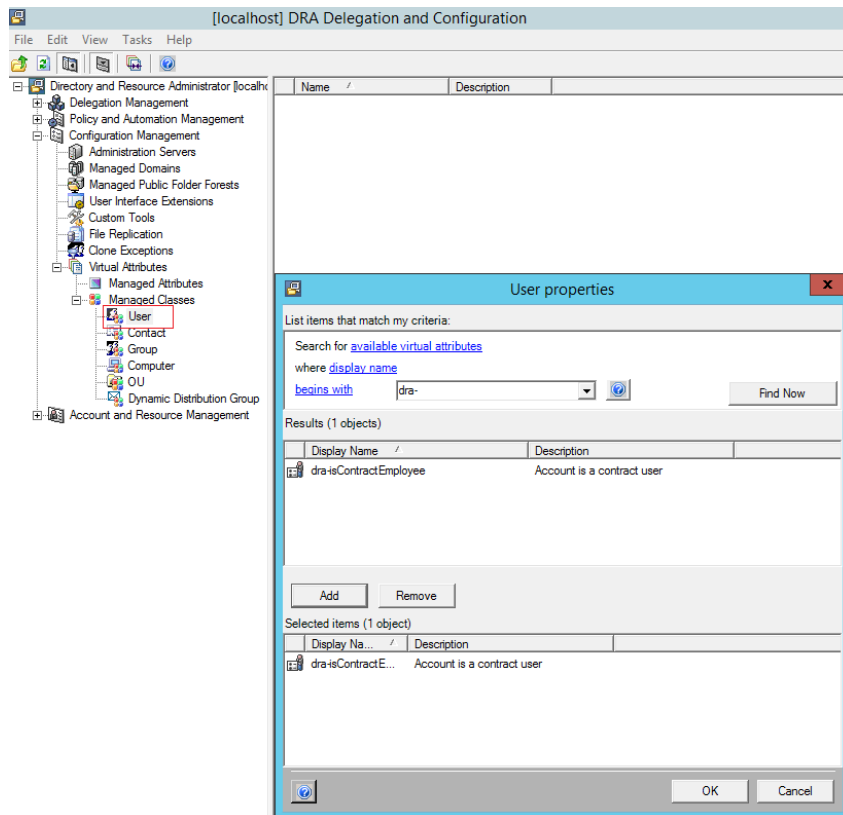


Define Attributes

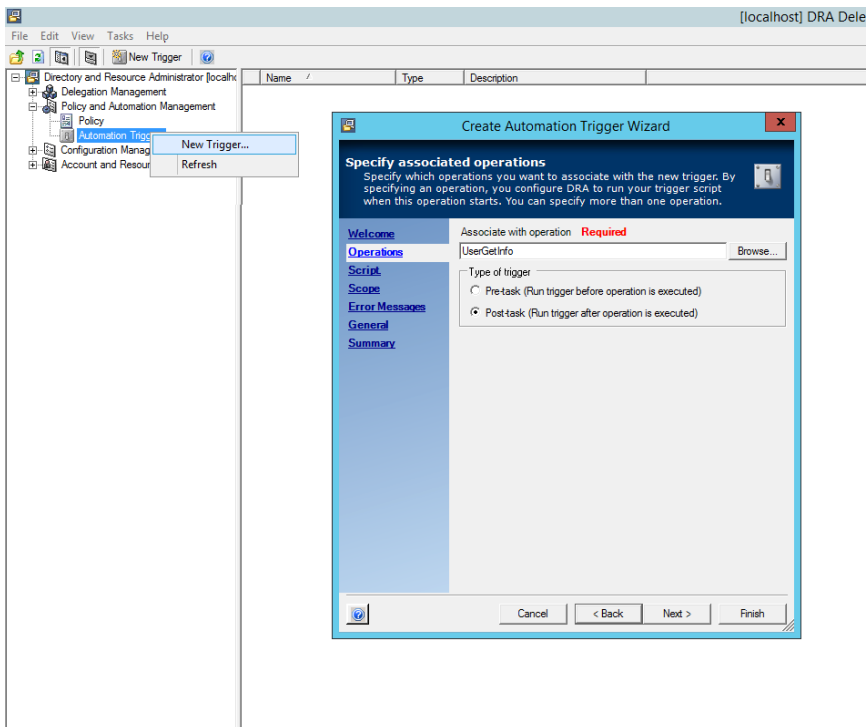
Create the following four attributes with the wizard:

- 'Name' = "va-ActiveSyncEnabled", 'Type' = Boolean
- 'Name' = "va-OWAEnabled", 'Type' = Boolean
- 'Name' = "va-MAPIEnabled", 'Type' = Boolean
- 'Name' = "va-EwsEnabled", 'Type' = Boolean

2. Associate the four new Virtual Attributes with the User object class.



3. Add a new post task trigger to the UserGetInfo() task that will execute a PowerShell command to collect the values of the new virtual attributes.



```
#####
#
# This is a post-task trigger that will collect additional Exchange Online Mailbox settings
# and expose them as additionalAttributes
#
#####

#####
#
# Function to convert DN (CN=Fred Smith,OU=North America,DC=ks,DC=com) to domain FQDN
# (keystone.com)
#
#####
function Convert-DN-To-FQDN([string] $dn)
{
    [array]$ADObjectDNArray = $dn -Split(",DC=")
    [int]$DomainNameFECCount = 0
    ForEach ($ADObjectDNArrayItem in $ADObjectDNArray)
    {
        if ($DomainNameFECCount -gt 0)
        {
            [string]$ADObjectDNArrayItemDomainName += $ADObjectDNArrayItem + "."
        }

        $DomainNameFECCount++
    }
    return $ADObjectDNArrayItemDomainName.Substring(0,$ADObjectDNArrayItemDomainName.Length-
1)
}

#####
#
# Verify the operation name is UserGetInfo, in other cases return $true
#
#####

$operation = $InVarSet.get("In.OperationName")

if ($operation -ne "UserGetInfo")
{
    return $true
}

#####
#
# Execute the script only if varset contains any of the keys ("In.ActiveSyncEnabled" ,
# "In.OWAEnabled" , "In.MAPIEnabled" , "In.EwsEnabled")
#
#####

$ActiveSyncEnabled = $InVarSet.get("In.Properties.va-ActiveSyncEnabled")
$OWAEnabled = $InVarSet.get("In.Properties.va-OWAEnabled")
$MAPIEnabled = $InVarSet.get("In.Properties.va-MAPIEnabled")
$EwsEnabled = $InVarSet.get("In.Properties.va-EwsEnabled")

if (!(($ActiveSyncEnabled-eq "") -and !($OWAEnabled-eq "") -and !($MAPIEnabled-eq "") `
-and !($EwsEnabled-eq ""))
{
    return $true
}
```

```

$userDN = $InVarSet.get("In.User").Replace("OnePoint://", "")
$domainFQDN = Convert-DN-To-FQDN($userDN)

Import-Module -Name NetIQ.DRA.PowershellExtensions
$user = Get-DRAUser -DRARestServer localhost -DRAHostServer localhost `
              -Domain $domainFQDN -Identifier $userDN `
              -Attribute "displayName","mailboxType" `
              -IgnoreCertificateErrors `
              -Force

if ($error.Count -gt 0)
{
    $errorMsg = "Failed to get $userDN displayname, error: $error[0]"
    $InVarSet.Put("Out.ErrorMessage.Script", $errorMsg)
    return $false
}

if (!$user.MailboxType -band 64)
{
    # User doesn't have an online mailbox
    return $true
}

$userDisplayName = $user.DisplayName

#####
#
# For this example we are using a plain text username and password
#
#####
$securePassword = ConvertTo-SecureString <Password> -AsPlainText -Force
$credentials = New-Object System.Management.Automation.PSCredential(<O365-Account>,
$securePassword)

#####
#
# Create the Exchange Online session
#
#####
$session = New-PSSession -ConfigurationName Microsoft.Exchange `
              -ConnectionUri https://outlook.office365.com/powershell-liveid/ `
              -Credential $credentials -Authentication Basic -AllowRedirection
Import-PSSession $Session -DisableNameChecking

#####
#
# Assume it will succeed
#
#####
$succeeded = $true

#####
#
# Execute command-let related O365 Mailbox
#
#####

$data = Get-CasMailbox -Identity $userDisplayName

#####
#
# Populate the out varset based on requested attribute
#
#####

```

```

if ($ActiveSyncEnabled -eq "")
{
    $InVarSet.Put("Out.Properties.va-ActiveSyncEnabled", $data.ActiveSyncEnabled)
}

if ($OWAEnabled -eq "")
{
    $InVarSet.Put("Out.Properties.va-OWAEnabled", $data.OWAEnabled)
}
if ($MAPIEnabled -eq "")
{
    $InVarSet.Put("Out.Properties.va-MAPIEnabled", $data.MAPIEnabled)
}

if ($EwsEnabled -eq "")
{
    $InVarSet.Put("Out.Properties.va-EwsEnabled", $data.EwsEnabled)
}

#####
#
# Cleanup
#
#####
Remove-PSSession -Session (Get-PSSession)

#####
#
# If the trigger failed send back an error message
#
#####
if ($error.Count -gt 0)
{
    $errorMsg = "Failed to get additional mailbox values for user: $userDN error: $error[0]"
    $InVarSet.Put("Out.ErrorMsg.Script", $errorMsg)
    $succeeded = $false
}

return $succeeded

```

4. To add the properties to the User Property Page, start by customizing the User Create Page and add a new 'Checkbox' field for each of the four properties.

The screenshot shows the DRA Administration console with the 'ADMINISTRATION' tab selected. Under 'CUSTOMIZATION', the 'USER' object type is chosen. The 'General' section is expanded, displaying various user creation fields. The 'User Logon Name' field includes a dropdown for 'Select domain suffix'. At the bottom, there are buttons for 'OK', 'APPLY', 'FORM PROPERTIES', and 'CANCEL'.

5. Configure each of the new fields for the new Virtual Attributes that we extended on the User object class.

Configure Form Fields

Configure the fields that were added to the form in the previous step as follows:

- Checkbox Field, Label: "Active Sync Enabled", 'Attribute' = "additionalAttributes.va-ActiveSyncEnabled"
- Checkbox Field, Label: "OWA Enabled", 'Attribute' = "additionalAttributes.va-OWAEnabled"
- Checkbox Field, Label: "MAPI Enabled", 'Attribute' = "additionalAttributes.va-MAPIEnabled"
- Checkbox Field, Label: "va-EWS Enabled", 'Attribute' = "additionalAttributes.va-EwsEnabled"

The screenshot shows a user configuration form with the following fields:

- User Logon Name (pre-Windows 2000) ***: A text input field with a placeholder text: "The pre-Windows 2000 account logon ID or user name and domain for the user (sAMAccountName)".
- Description**: A text input field.
- Employee ID**: A text input field with a placeholder text: "The employee ID of the user (employeeID). The maximum character length for this field is 16."
- Virtual Attributes**: A section containing four checkboxes, all of which are unchecked and highlighted by a red rectangular box:
 - ☐ Active Sync Enabled
 - ☐ OWA Enabled
 - ☐ MAPI Enabled
 - ☐ va-EWS Enabled

Enable or disable an Exchange Online user's Outlook web access when they are enabled or disabled in Active Directory

Scope

This example demonstrates how to define add a custom post task trigger to Enable or disable an Exchange Online user's Outlook web access when they are enabled or disabled in Active Directory.

Step-by-step Guide

Add a new post task trigger to the UserSetInfo() task that will execute a PowerShell command to update the users Outlook web access when they are enabled or disabled in Active Directory.

Post Task Trigger

```
#####  
#  
# This is a post-task trigger for the UserEnable operation. The UserEnable  
# operation is used to enable and disable users.  
#  
# We will enable or disable the user's Outlook web access when they are  
# enabled or disabled.  
#  
#####  
  
#####  
#  
# Function to convert DN (CN=Fred Smith,OU=North America,DC=ks,DC=com) to domain FQDN (keystone.com)  
#  
#####  
function Convert-DN-To-FQDN([string] $dn)  
{  
    [array]$ADObjectDNArray = $dn -Split(",DC=")  
  
    [int]$DomainNameFECCount = 0  
  
    ForEach ($ADObjectDNArrayItem in $ADObjectDNArray)  
    {  
        if ($DomainNameFECCount -gt 0)  
        {  
            [string]$ADObjectDNArrayItemDomainName += $ADObjectDNArrayItem + "."  
        }  
  
        $DomainNameFECCount++  
    }  
  
    return $ADObjectDNArrayItemDomainName.Substring(0,$ADObjectDNArrayItemDomainName.Length-1)  
}  
  
#####  
#  
# This is a quirk of DRA. To get a pre/post task trigger to fire on the  
# UserEnable operation you actually have to specify the UserSetInfo operation  
# when you configure your trigger. That means we need to make sure the operation  
# is actually UserEnable  
#  
#####  
$operation = $InVarSet.get("In.OperationName")
```

```

if ($operation -ne "UserEnable")
{
    return $true
}

#####
#
# Get the user's mailboxType and displayName. If they don't have an online
# mailbox just return.
#
# We need to the user's display name to use as the -Identity value when we
# call Exchange Online.
#
#####
$userDN = $InVarSet.get("In.User").Replace("OnePoint://", "")
$domainFQDN = Convert-DN-To-FQDN($userDN)

Import-Module -Name NetIQ.DRA.PowershellExtensions

$user = Get-DRAUser -DRARestServer localhost -DRAHostServer localhost `
    -Domain $domainFQDN -Identifier $userDN `
    -Attribute "displayName","mailboxType" `
    -IgnoreCertificateErrors `
    -Force

if ($error.Count -gt 0)
{
    $errorMsg = "Failed to move $userDN to OU ${ouToMoveTo}: $error[0]"
    $InVarSet.Put("Out.ErrorMessage.Script", $errorMsg)
    return $false
}

if (!(($user.MailboxType -band 64))
{
    # User doesn't have an online mailbox
    return $true
}

$userDisplayName = $user.DisplayName

#####
#
# See if we are enabling or disabling the user
#
#####
$enableOWA = $InVarSet.get("In.Enable")

#####
#
# For this example we are using a plain text username and password
#
#####
$securePassword = ConvertTo-SecureString <Password> -AsPlainText -Force
$credentials = New-Object System.Management.Automation.PSCredential(<user@domain>, $securePassword)

#####
#
# Create the Exchange Online session
#
#####
$session = New-PSSession -ConfigurationName Microsoft.Exchange `
    -ConnectionUri https://outlook.office365.com/powershell-liveid/ `
    -Credential $credentials -Authentication Basic -AllowRedirection
Import-PSSession $Session -DisableNameChecking

#####
#

```

```

# Assume it will succeed
#
#####
$succeeded = $true

#####
#
# See if OWA is enabled.
#
#####
$enabled = Get-CasMailbox -Identity $userDisplayName | Select-Object -ExpandProperty OWAEnabled

#####
#
# Only enable or disable it if we need to
#
#####
if ($enabled -ne $enableOWA)
{
    #####
    #
    # Enable or disable the user's Outlook web access
    #
    #####
    Set-CasMailbox -Identity $userDisplayName -OWAEnabled $enableOWA

    #####
    #
    # Validate it was changed
    #
    #####
    $enabled = Get-CasMailbox -Identity $userDisplayName | Select-Object -ExpandProperty OWAEnabled

    if ($enabled -ne $enableOWA)
    {
        $succeeded = $false
    }
}

#####
#
# Cleanup
#
#####
Remove-PSSession -Session (Get-PSSession)

#####
#
# If the trigger failed send back an error message
#
#####
if ($succeeded -eq $false)
{
    $enableOrDisable = if ($enableOWA) { "enable" } else { "disable" }

    if ($error.Count -gt 0)
    {
        $errorMsg = "Failed to $enableOrDisable Outlook web access for ${userDisplayName}: $error[0]"
    }
    else
    {
        $errorMsg = "Failed to $enableOrDisable Outlook web access for $userDisplayName"
    }

    $InVarset.Put("Out.ErrorMsg.Script", $errorMsg)
}

return $succeeded

```


Launch Workflow from Trigger

Scope

This example demonstrates how to define add a custom pre task trigger to launch a workflow.

Step-by-step Guide

1. Define a workflow that will update the manager when object is moved from one OU to another OU.
2. Add a new pre task trigger to launch the custom work.

Pre-Task Trigger

```
$operation = $InVarSet.get("In.OperationName")

if ($operation -ne "OUMoveHere")
{
    return $true
}

#####
#
# Find the Target Container
#
#####
$targetOU = $InVarSet.get("In.Container")

#####
#
# Find the object which is moving
#
#####

$movingObject = $InVarSet.get("In.Object")

#####
#
# Create server and varset Object
#
#####
$ea1 = New-Object -comobject EaServer.EaServe.1
$NewVarSet = New-Object -comobject NetIQDraVarSet.VarSet.1
```

```

#
# Populate the Varset based on workItem attribute defined in Aegis
# here UserDN is the workitem attribute to represent User which is moving to different
container
#
#####
$NewVarSet.put("Properties.UserDN", $movingObject)

#####
#
# Populate the Varset based on workItem attribute defined in Aegis
# here managerDN is the workitem attribute to represent manager the value has
# to be DN format
#
#####
$NewVarSet.put("Properties.managerDN", "cn=admin400_vino,ou=vino_400,dc=drdom400,dc=lab")

#####
#
# Populate the Varset for RequestID which is unique ID generated by randomly
#
#####
$NewVarSet.put("RequestID", "upot66f5wfaxktvga4vn43kq")

#####
#
# Populate the Varset for WorkflowID , it is unique path of the Aegis workflow
#
#####
$NewVarSet.put("WorkflowID", "/PG_TopLevelContainer=PG/PG_Process=7d6fb8d9-b2c1-4478-b263-
61468e121d79")

$succeeded = $true

#Submit the varset
$vsOut1 = $ea1.ScriptSubmit($NewVarSet)

```

```
$error = $vsOut1.Get("Errors.NumErrors")
```

```
If ($error -gt 0){
```

```
    $succeeded = $false
```

```
}
```

```
Else{
```

```
    $succeeded = $true
```

```
}
```

```
return $succeeded
```

Move a user to a new OU when their Office property is changed

Scope

This example demonstrates how to define add a custom pre & post task trigger to Move a user to a new OU when their Office property is changed and change their 'City' attribute to match their new office location.

Step-by-step Guide

1. Add a new pre task trigger to validate there is an OU matching the 'Office' attribute and update the 'City' attribute in the input VARSET.

Pre Task Trigger

```
#####
#
# This is a pre-task trigger that will make sure there is a matching OU when a user's Office
# property is changed.
# It will also make sure the user's City property is changed to match the Office property.
#
# There is a corresponding post-task trigger that will move the user to the new OU.
#
# 1. Make sure we have an OU with the same name as the Office
(In.Properties.physicalDeliveryOfficeName).
# -- If we don't find a matching OU return $false.
# 2. Change the City property ("l") to match the Office property
#
#####

#####
#
# See if the office value was changed
#
#####
$office = $InVarSet.get("In.Properties.physicalDeliveryOfficeName")

if (!$office)
{
    return $true
}

#####
#
# Get the user's domain
```

```
#####
$userDN = $InVarSet.get("In.User").Replace("OnePoint://", "")
$domain = $userDN.Substring($userDN.indexOf("DC="))

#####
#
# Make a call to DRA to see if we can find an OU matching the office
#
#####
Import-Module -Name NetIQ.DRA.PowershellExtensions

$ous = Find-DRAObjects -DRARestServer localhost -DRAHostServer localhost `
    -ContainerDN "OnePoint://$domain" `
    -OUAndFilter @{FriendlyName="$office"} `
    -OUOrFilter @{ShowInAdvancedViewOnly=$false} `
    -Attributes "Path" `
    -IncludeChildContainers `
    -IgnoreCertificateErrors `
    -Force

if ($ous.TotalObjectCount -ne 1)
{
    $InVarSet.Put("Out.ErrorMessage.Script", "Could not find an OU named $office")
    return $false
}

#####
#
# Store the DN of the OU so we can use it in the post-task trigger
#
#####
$InVarSet.put("WorkArea.OUToMoveTo", $ous.Items[0].Path.Replace("OnePoint://", ""))

#####
#
# Update the City property
#
#####
$InVarSet.put("In.Properties.1", $office)

return $true
```

2. Add a new post task trigger to move the user to the new OU if their 'Office' attribute changed

Post Task Trigger

```
#####  
#  
# This is a post-task trigger that will move a user to the OU specified in the Office  
# property  
#  
# There is a corresponding pre-task trigger that will set WorkArea.OUToMoveTo if the user  
# needs to be moved.  
#  
#####  
  
#####  
#  
# Function to convert DN (CN=Fred Smith,OU=North America,DC=ks,DC=com) to domain FQDN  
# (keystone.com)  
#  
#####  
function Convert-DN-To-FQDN([string] $dn)  
{  
    [array]$ADObjectDNArray = $dn -Split(",DC=")  
  
    [int]$DomainNameFECCount = 0  
  
    ForEach ($ADObjectDNArrayItem in $ADObjectDNArray)  
    {  
        if ($DomainNameFECCount -gt 0)  
        {  
            [string]$ADObjectDNArrayItemDomainName += $ADObjectDNArrayItem + "."  
        }  
  
        $DomainNameFECCount++  
    }  
  
    return $ADObjectDNArrayItemDomainName.Substring(0,$ADObjectDNArrayItemDomainName.Length-  
1)  
}  
  
#####  
#  
# See if the pre-task trigger set an OU for us to move the user to
```

```

#
#####
$ouToMoveTo = $InVarSet.get("WorkArea.OUToMoveTo")

if (!$ouToMoveTo)
{
    return $true
}

#####
#
# Get the domain FQDN to use in the Move-DRAUser cmdlet
#
#####
$userDN = $InVarSet.get("In.User").Replace("OnePoint://", "")
$domainFQDN = Convert-DN-To-FQDN($userDN)

#####
#
# Make a call to DRA to move the user to the office OU
#
#####
Import-Module -Name NetIQ.DRA.PowershellExtensions

Move-DRAUser -DRARestServer localhost -DRAHostServer localhost `
    -Domain $domainFQDN -Identifier $userDN -TargetContainer $ouToMoveTo `
    -IgnoreCertificateErrors `
    -Force

if ($error.Count -gt 0)
{
    $errorMsg = "Failed to move $userDN to OU ${ouToMoveTo}: $error[0]"
    $InVarSet.Put("Out.ErrorMessage.Script", $errorMsg)
    return $false
}

return $true

```

Web Client Customization

This section contains examples for customizing object property pages in the Web Console

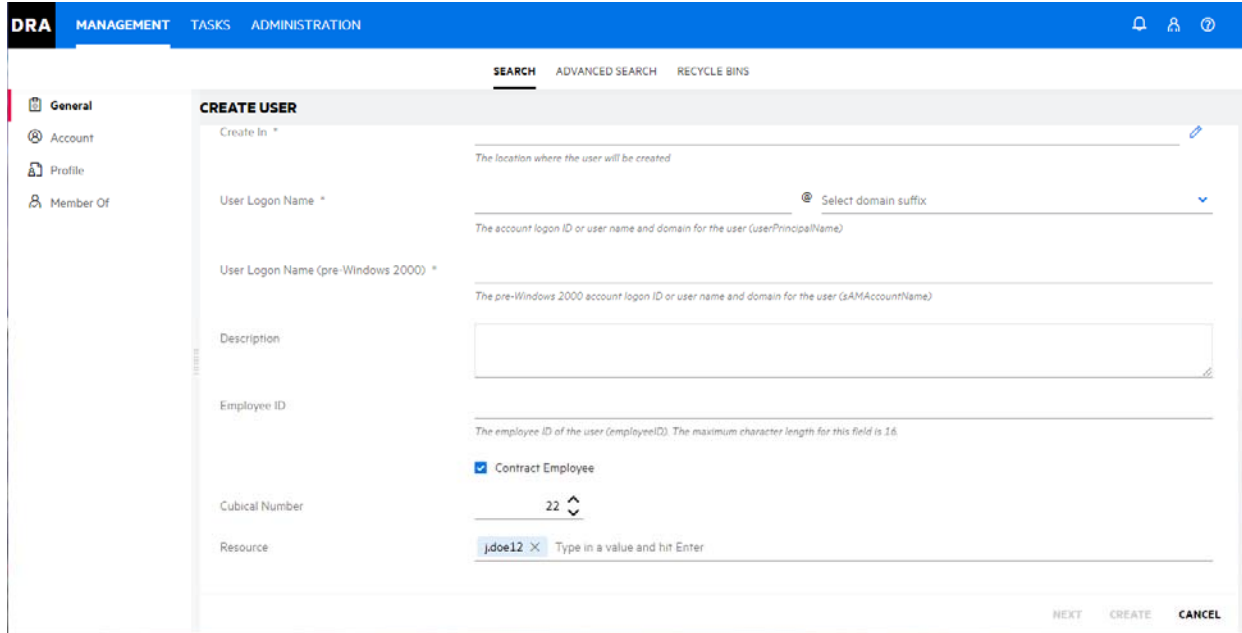
Custom Property Page Examples;

Use Case	Context
Pre-populate property values in form: Pre-populate Value (A) Based on Value (B) Using Built-in Attribute Setting Pre-populate value automatically based on custom handler	• On Attribute Change • On Load • Script Custom Handler • Built-in attribute function
Performing custom validation on form submit: On Form Submit, Validate Field (A) has a Value When Field (B) has a Value Validate Name Uniqueness for a Specified Field Value When a Form is Submitted	• On Submit • LDAP Query Custom Handler
Performing custom validation on property value change: Indicate Field (A) is Required When Field (B) has a Value at Runtime Validate Name Uniqueness for a Specified Field Value at Run-time	• On Attribute • Change Messagebox
Additional form customization scenarios: Populate selection values for a field based on previous field selection Add Virtual Attributes to Custom Forms	• On Attribute Change • LDAP Query custom handler • Custom Script Handler • Virtual attributes
Working with regular expressions in form properties Validate input using a regular expression	• Built-in attribute function

Add Virtual Attributes to Custom Forms

This example demonstrates how to define virtual attributes in DRA and add them to the user create page. We will define three new virtual attributes of different type ("dra-isContractEmployee", "dra-cubicleNumber" and "dra-orgResourceIDList") and associate them with a user property page.

Step-by-step Guide

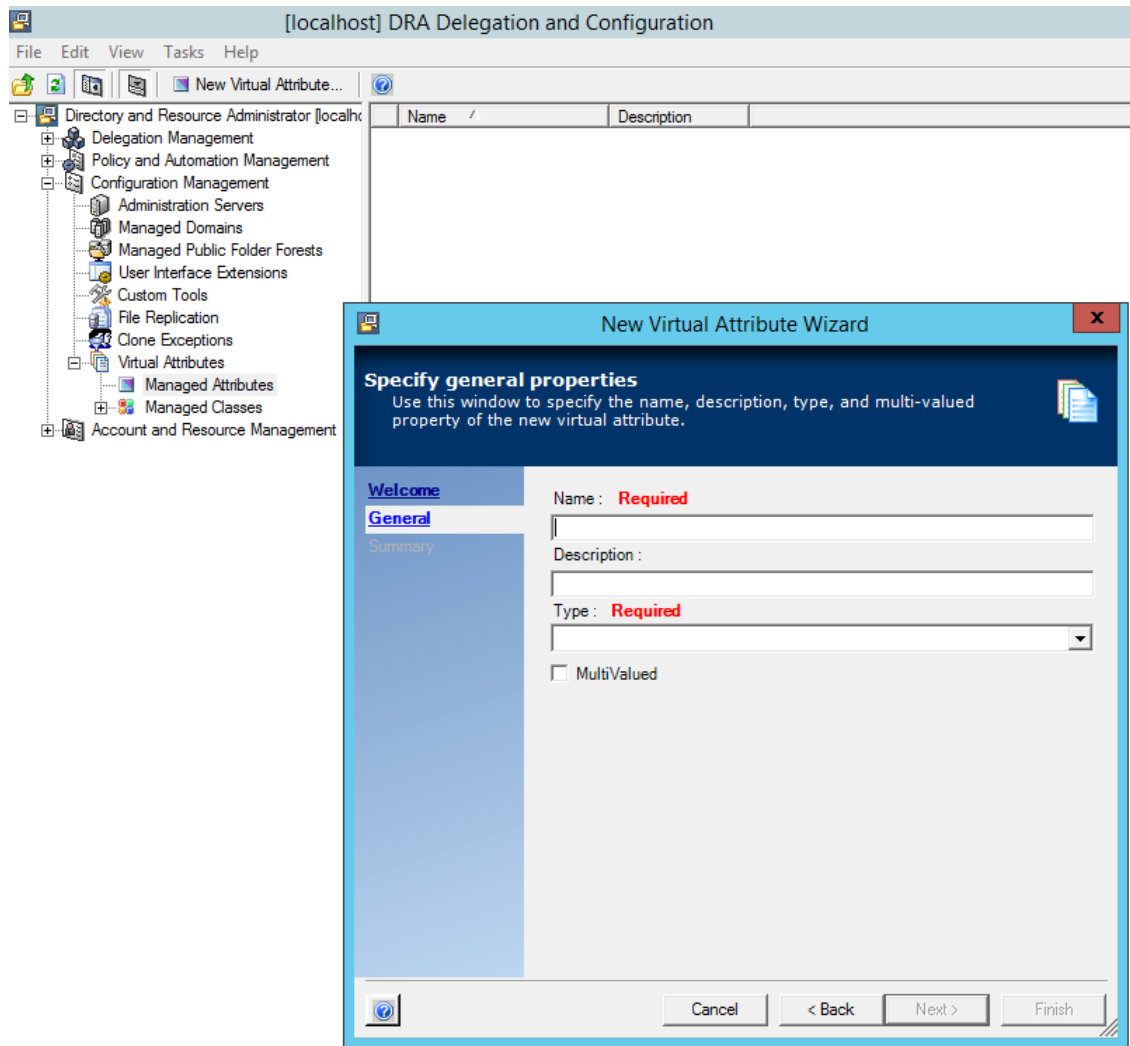


The screenshot shows the 'CREATE USER' form in the DRA (Delegation and Configuration Administration) console. The form is titled 'CREATE USER' and has a sidebar with navigation options: General, Account, Profile, and Member Of. The main form area contains several fields and options:

- Create In ***: A dropdown menu with a blue pencil icon for editing.
- User Logon Name ***: A text input field with a blue dropdown arrow for 'Select domain suffix'. Below it is a hint: 'The account logon ID or user name and domain for the user (userPrincipalName)'.
- User Logon Name (pre-Windows 2000) ***: A text input field. Below it is a hint: 'The pre-Windows 2000 account logon ID or user name and domain for the user (sAMAccountName)'.
- Description**: A large text area.
- Employee ID**: A text input field. Below it is a hint: 'The employee ID of the user (employeeID). The maximum character length for this field is 16'.
- Contract Employee**: A checkbox that is checked.
- Cubical Number**: A text input field with the value '22' and a blue dropdown arrow.
- Resource**: A text input field with the value 'jdoe12' and a blue 'X' icon. Below it is a hint: 'Type in a value and hit Enter'.

At the bottom right of the form are three buttons: 'NEXT', 'CREATE', and 'CANCEL'.

1. Start by creating a virtual attribute in the Delegation and Configuration Console:
 - On the "Configuration Management" node, navigate to the "Virtual Attributes" section
 - Select the option "New Virtual Attribute" from the "Managed Attributes" subheading

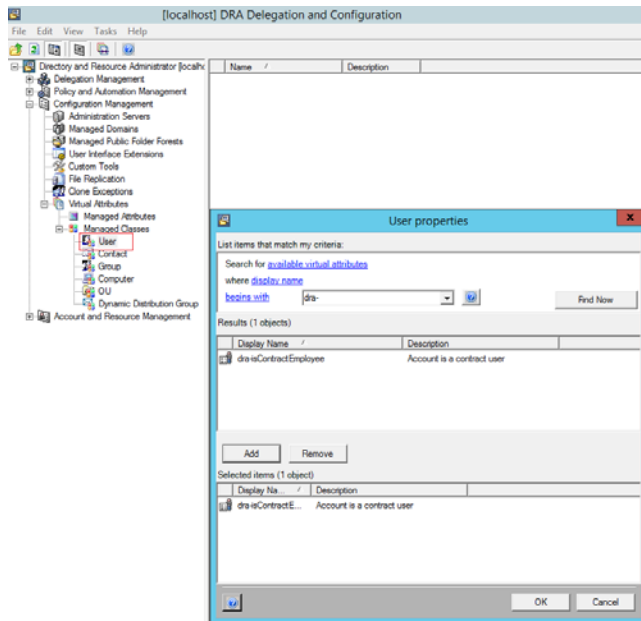


Define Attributes

Create the following three attributes with the wizard:

- 'Name' = "dra-isContractEmployee", 'Type' = Boolean
- 'Name' = "dra-cubicleNumber", 'Type' = Integer
- 'Name' = "dra-orgResourceIDList", 'Type' = String, Multi Value = True

2. Associate the three new Virtual Attributes with the User object class.



- To add the properties to the User Create page, start by customizing the User Create page in the Web Console and add new fields for each property by navigating to Administration > Customization > Property Pages and edit the User > Create Object.

Add Property Fields

Add the following three fields to the form on the 'General Tab':

- Field Type = 'Checkbox'
- Field Type = 'Number'
- Field Type = 'Multi Valued Text'

4. Configure each of the new fields for the new Virtual Attributes that we extended on the User object class.

The screenshot shows a configuration window with tabs: GENERAL, CHECKBOX OPTIONS, INLINE FIELDS, SUB FIELDS, and CUSTOM HANDLERS. The 'GENERAL' tab is active. It contains the following fields and options:

- Label ***: Contract Employee
- Attribute ***: additionalAttributes.dra-isContractEmployee
- ☐ This is a form only attribute
This attribute is only available on the local form and its value will not be loaded from or submitted to the server
- Description**: (empty text area)
- ☐ Hidden
- ☐ Read only
- ☐ Extend control content to the label area (label will be hidden, if applicable)
- ☐ Do not retrieve the source object's current value when cloning

At the bottom right are buttons for **OK** and **CANCEL**.

Configure Form Fields

Configure the fields that were added to the form in the previous step as follows:

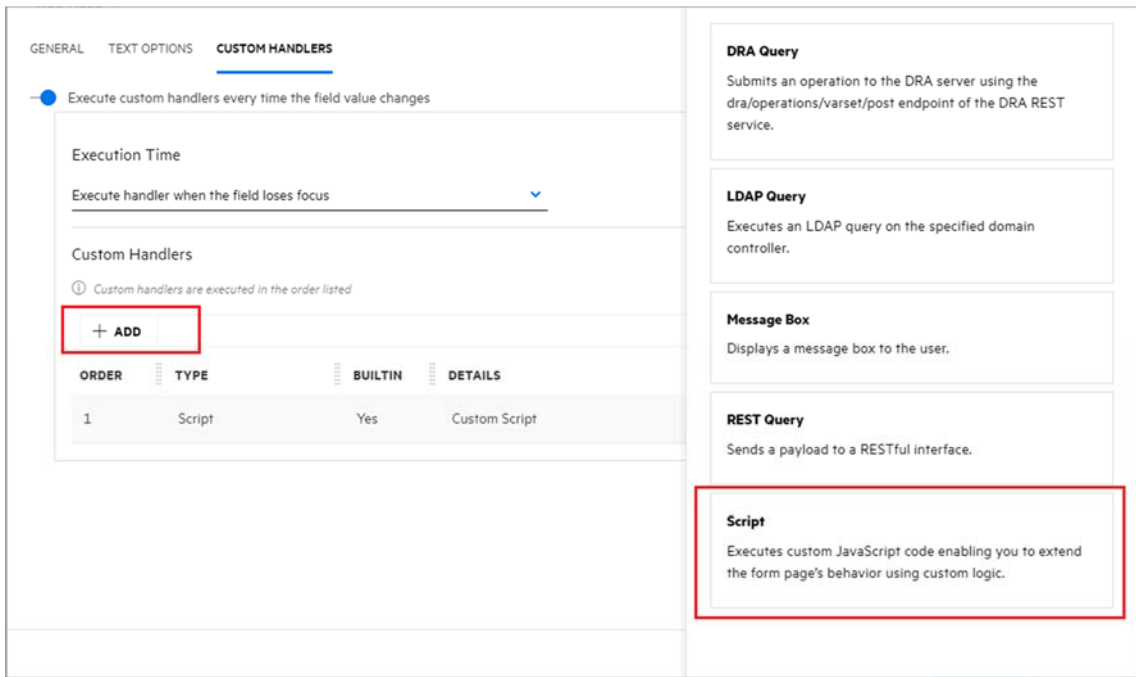
- Checkbox Field: "Contract Employee", 'Attribute' = "additionalAttributes.dra-isContractEmployee"
- Number Field: "Cubical Number", 'Attribute' = "additionalAttributes.dra-cubicleNumber"
- Multi Valued Field: 'Label' = "Resource", 'Attribute' = "additionalAttributes.dra-orgResourceIDList"

Indicate Field (A) is Required When Field (B) has a Value at Runtime

This example demonstrates how to make a field required at runtime as soon as a second field on the form has a value entered. We'll customize the User Create pages to make Last Name required whenever the user enters a value for First Name. Now when the user enters a value for First Name the Last Name field is marked as required and the form cannot be submitted until either a Last Name is entered or First Name is cleared.

Step-by-step Guide

1. Start by customizing the User Create pages; edit  the properties of the First Name field, and add a Script handler on the Custom Handlers tab. Click **Add > Script** to configure the script.



GENERAL TEXT OPTIONS **CUSTOM HANDLERS**

● Execute custom handlers every time the field value changes

Execution Time

Execute handler when the field loses focus

Custom Handlers

① Custom handlers are executed in the order listed

+ ADD

ORDER	TYPE	BUILTIN	DETAILS
1	Script	Yes	Custom Script

DRA Query
Submits an operation to the DRA server using the dra/operations/varset/post endpoint of the DRA REST service.

LDAP Query
Executes an LDAP query on the specified domain controller.

Message Box
Displays a message box to the user.

REST Query
Sends a payload to a RESTful interface.

Script
Executes custom JavaScript code enabling you to extend the form page's behavior using custom logic.

NOTE:

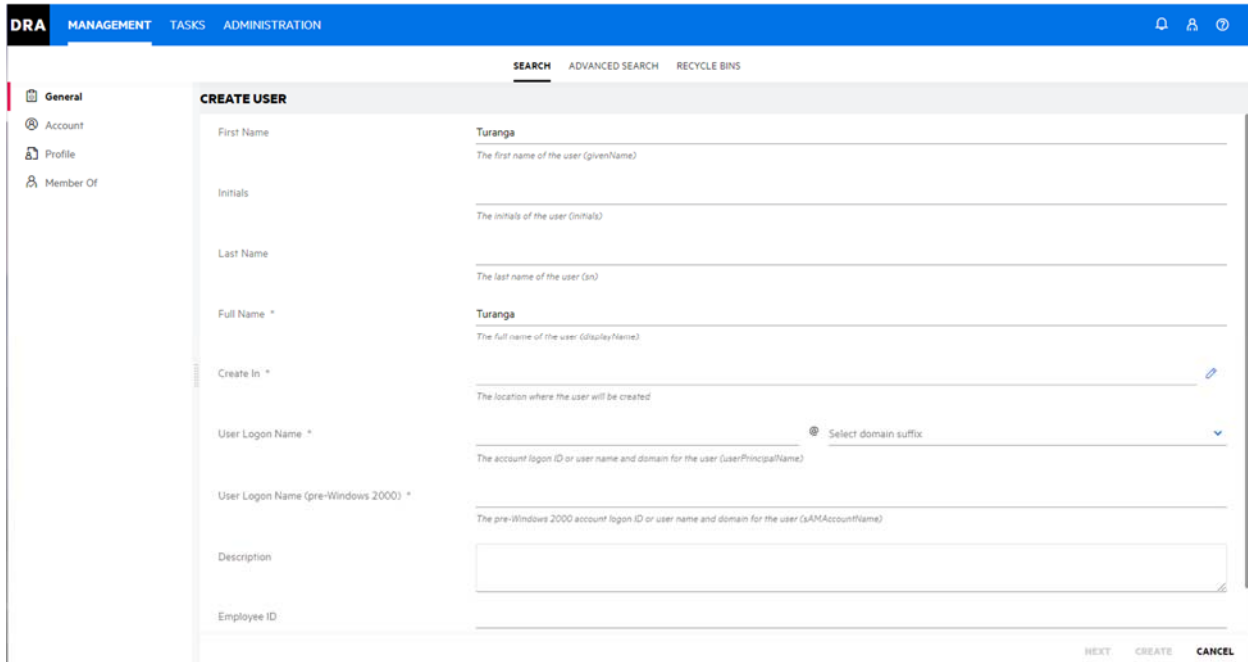
To allow customizations to include custom JavaScript code add `allowCustomJavaScript="true"` to the `<consoleConfiguration>` element found in `%ProgramData%\NetIQ\DRARESTProxy\restProxy.config` on the web server.

2. This script will set the required state of the lastName field to equal the truthiness of the firstName field.

```
draApi.fieldRequiredStates.lastName = Boolean(draApi.fieldValues.firstName);
```

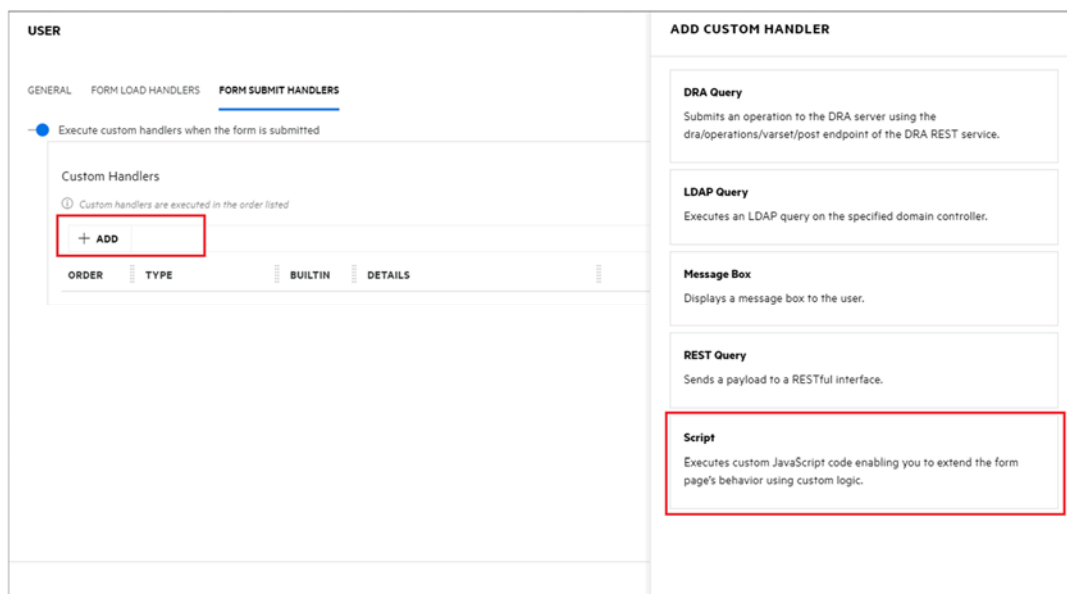
On Form Submit, Validate Field (A) has a Value When Field (B) has a Value

This example demonstrates how to validate a form 'on submit' to verify the user has entered a value for a field whenever a second field was filled in. In this example we'll customize the User Create pages to make sure that if the user fills in the First Name field that they also fill in the Last Name field. When the user attempts to create a user with a first name but not a last name they are shown a warning message indicating that the last name field needs to be entered before the form can be submitted.



Step-by-step Guide

1. Start by customizing the User Create page; open the Form Properties dialog, and add a Script custom handler in the Form Submit Handlers tab. Click **Add > Script** to create the script.



NOTE:

To allow customizations to include custom JavaScript code add `allowCustomJavaScript="true"` to the `<consoleConfiguration>` element found in `%ProgramData%\NetIQ\DRARESTProxy\restProxy.config` on the web server.

You will need to log out of the console and log back in to see this dialog go away

2. This script will check if the value of the firstName field is truthy but lastName field is not. If the condition returns true, we will cancel the form submission and inform the user they need to enter a value for Last Name.

CONFIGURE SCRIPTHELP

```
1 if (draApi.fieldValues.firstName && !draApi.fieldValues.lastName) {  
2   flowControl.cancel('Last Name is required when First Name is specified. Please enter a value for Last Name');  
3 }
```

OK **CANCEL**

Script

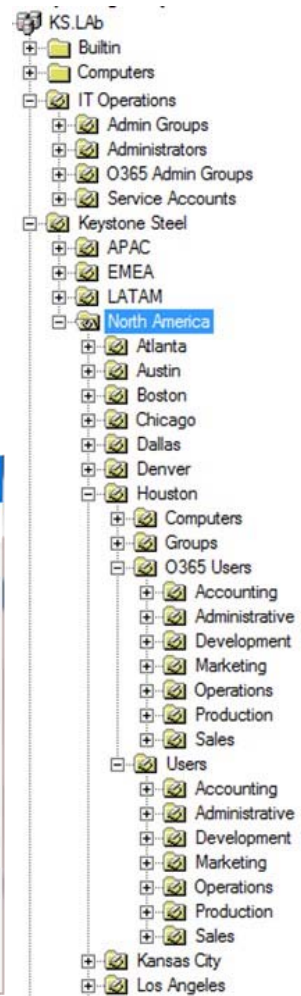
```
if (draApi.fieldValues.firstName && !draApi.fieldValues.lastName) {  
  flowControl.cancel('Last Name is required when First Name is specified. Please enter a  
  value for Last Name');  
}
```

Populate selection values for a field based on previous field selection

This example shows how to use custom handlers in the DRA web client to populate selection objects based on a previous selection and checkbox. The OU structure that this is based on can be seen in the image to the right. This custom page starts at the Keystone Steel level within the tree and provides the values of the Region to the first selection object, the second selection gets the cities based on the first selection, and the third selection provides a list of the departments based on both the first and second selection, and the checkbox. The checkbox is used to determine if the selection for department should be in either the O365 Users or Users OU. Only the First and Second selection objects require any custom handlers, and the first selection is the only one that requires an LDAP query on the options tab.

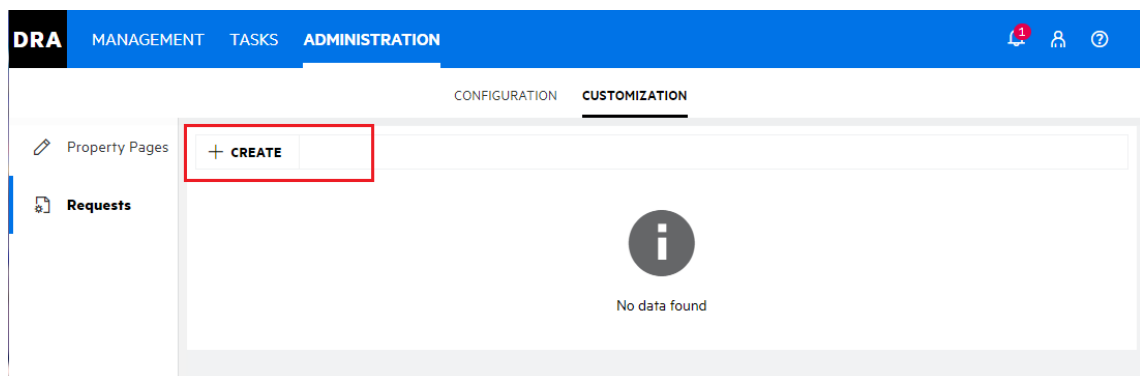
The following pages will contain screen captures of each setting for each of the selection objects and the checkbox in a Workflow Automation Request form.

The screenshot shows the DRA web client interface. The top navigation bar includes 'DRA', 'MANAGEMENT', 'TASKS', and 'ADMINISTRATION'. The main content area is titled 'SUBMIT REQUEST' and 'Get OU Properties'. It features a checkbox for 'Office 365 Users' and three dropdown menus: 'Region' (selected: EMEA), 'City' (selected: London), and 'Department' (selected: Accounting). The 'SUBMIT' and 'CANCEL' buttons are at the bottom right.



Step-by-step Guide

1. Navigate to Administration > Customization > Requests, and click **Create**.



2. In the General tab, name the form, select the icon type and **Trigger Workflow by Event** option, enter an event name for the trigger, and click **OK** to save the form. The new form is added with a blank page titled, "New Tab".

The screenshot shows the 'NEW REQUEST FORM' configuration window with the 'GENERAL' tab selected. The window has a title bar and a main content area. The 'GENERAL' tab is active, and the 'ASSISTANT ADMIN GROUPS' sub-tab is selected. The 'Form Display Name' field is empty. The 'Icon' field shows a 'TRIGGER' icon. The 'Form is hidden' checkbox is unchecked. The 'Configure Workflow' section shows two options: 'Launch Specific Workflow' (unselected) and 'Trigger Workflow by Event' (selected). Under 'Trigger Workflow by Event', the 'Workflow Process event name' field contains 'Select Cascading OUs'. The 'OK' and 'CANCEL' buttons are at the bottom right.

NEW REQUEST FORM

GENERAL ASSISTANT ADMIN GROUPS

Form Display Name *

Icon

Get OU Properties

☐ Form is hidden

Users logging into the Web Console will not see hidden actions but they are still available for editing

Configure Workflow*

☐ Launch Specific Workflow

Select a specific workflow that should be executed when this form is submitted.

☒ Trigger Workflow by Event

Workflow Process event name *

Select Cascading OUs

Submit a named workflow trigger event containing all of the form attributes defined to the Workflow Engine. All workflow templates configured to trigger on an event with the event name specified will be executed.

OK CANCEL

3. Open **Properties** for the page from the ellipsis icon to rename the page and select an icon for the page.
4. From the Add Field drop-down menu, add the checkbox ("Office 365 User") and the three list selection (*Select field*) ("Region", "City", "Department") controls to the custom form.

The screenshot shows the 'CHECKBOX' configuration window with the 'GENERAL' tab selected. The window has a title bar and a main content area. The 'GENERAL' tab is active, and the 'CHECKBOX OPTIONS' sub-tab is selected. The 'Label' field contains 'Office 365 Users'. The 'Attribute' field contains 'O365'. The 'This is a form only attribute' checkbox is unchecked. The 'Description' field is empty. The 'Hidden', 'Read only', and 'Extend control content to the label area' checkboxes are unchecked. The 'OK' and 'CANCEL' buttons are at the bottom right.

CHECKBOX

GENERAL CHECKBOX OPTIONS INLINE FIELDS SUB FIELDS CUSTOM HANDLERS

Label *

Office 365 Users

Attribute *

O365

☐ This is a form only attribute

This attribute is only available on the local form and its value will not be loaded from or submitted to the server

Description

☐ Hidden

☐ Read only

☐ Extend control content to the label area (label will be hidden, if applicable)

OK CANCEL

GENERAL SELECT OPTIONS CUSTOM HANDLERS

Label * Region

Attribute * region

☒ This is a form only attribute
This attribute is only available on the local form and its value will not be loaded from or submitted to the server

Description

☐ Hidden

☐ Required

☐ Read only

☐ Extend control content to the label area (label will be hidden, if applicable)

OK CANCEL

- Configure the "Region" selection list field to populate with base selection values by defining a "Dynamic" LDAP query on the Select Options tab. When configuring the Query Parameters, enter the base distinguished name for the Region OU, choose One Level for the Scope, enter name for the Attributes, and add the string below for the Filter.

Region Options tab LDAP query

```
(&(objectCategory=organizationalUnit)(objectClass=organizationalUnit))
```

SELECT

GENERAL **SELECT OPTIONS** CUSTOM HANDLERS

No Selection Text

Options

Dynamic

Options Labels Query Attribute name

Options Values Query Attribute name

+ ADD

ORDER	TYPE	BUILTIN	DETAILS
Required			

OK CANCEL

6. Add a Custom "LDAP Query" to the "Region" selection list field from the "Custom Handlers" tab and configure the Pre-Query and Post-Query logic, using the same settings as the Select Options in Query Parameters.

```
1 ldapOptions.baseDn = 'ou=' + draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';
2
3 ldapOptions.filter = '(&(objectCategory=organizationalUnit)(objectClass=organizationalUnit))';
```

Region Custom Handler Pre-query LDAP script

```
ldapOptions.baseDn = 'ou=' + draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';

ldapOptions.filter =
'(&(objectCategory=organizationalUnit)(objectClass=organizationalUnit))';
```

Region Custom Handler Post-query LDAP script

```
var labels = [];
var values = [];
var attrName;
var attrValue;

for (var i = 0; i < results.length; i++) {
    labels.push(results[i].attributes[0].values[0]);
    values.push(results[i].attributes[0].values[0]);
}

draApi.fieldOptionLabels.city = labels;
draApi.fieldOptionValues.city = values;
```

7. When configuring the new Select field for City, leave the Select Options tab at Static and use a placeholder for the Add Option.
8. Add a Custom "LDAP Query" to the "City" selection list field from the Properties field dialog on the "Custom Handlers" tab and configure the Pre-Query and Post-Query logic. Use the same configuration in Query Parameters as before, but with the addition of distinguishedName in the Attributes field.

CONFIGURE LDAP QUERY

HELP

CONNECTION INFORMATION

QUERY PARAMETERS

PRE-QUERY ACTION

POST-QUERY ACTION

```

1 if (draApi.fieldValues.0365) {
2   ldapOptions.baseDn = 'ou=0365 Users,ou=' + draApi.fieldValues.city + ',ou=' +
3   draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';
4 } else {
5   ldapOptions.baseDn = 'ou=Users,ou=' + draApi.fieldValues.city + ',ou=' +
6   draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';
7 } }

```

OK

CANCEL

City Custom Handler Pre-query LDAP script

```

if (draApi.fieldValues.0365) {
    ldapOptions.baseDn = 'ou=0365 Users,ou=' + draApi.fieldValues.city + ',ou=' +
draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';
} else {
    ldapOptions.baseDn = 'ou=Users,ou=' + draApi.fieldValues.city + ',ou=' +
draApi.fieldValues.region + ',ou=Keystone Steel,dc=ks,dc=lab';
} }

```

City Custom Handler Post-query LDAP script

```

var labels = [];
var values = [];
var attrName;
var attrValue;

for (var i = 0; i < results.length; i++) {
    for(var x = 0; x < results[i].attributes.length; x++) {
        attrName = results[i].attributes[x].attrName;
        attrValue = results[i].attributes[x].values[0];
        if (attrName.toLowerCase() === 'name') {
            labels.push(attrValue);
        } else if (attrName.toLowerCase() === 'distinguishedname') {
            values.push(attrValue);
        }
    }
}

draApi.fieldOptionLabels.department = labels;
draApi.fieldOptionValues.department = values;

```

Pre-populate Value (A) Based on Value (B) Using Built-in Attribute Setting

Example of how to pre-populate a field value based on another property field value using the built-in attribute setting customization "Generate value automatically based on other values". In this example we will pre-populate the 'Description' of a User with his 'Department' name in the User Properties page as "Department-<Department Name>".

Step-by-step Guide

1. Start by customizing the User Create page and edit the 'Description' property.

The screenshot shows the 'ADMINISTRATION' tab in the DRA interface, specifically the 'CUSTOMIZATION' section for the 'USER' object type. The 'General' tab is selected in the left sidebar. The main area displays several user properties: 'Full Name', 'Create In', 'User Login Name', 'User Login Name (pre-Windows 2000)', 'Description', and 'Employee ID'. The 'Description' field is highlighted with a red box, indicating it is the target for customization. A 'Properties' button is visible next to the 'Description' field.

2. Select the "Generate value automatically based on other values" option in the Text Area Options tab.

The screenshot shows the 'TEXT AREA' configuration page, specifically the 'TEXT AREA OPTIONS' tab. The 'Number of Rows' is set to 3. The 'Validate input using a regular expression' checkbox is checked. Below this, there are fields for 'Regular Expression Type' (set to 'Custom'), 'Regular Expression', 'Match is case sensitive' (unchecked), and 'Test Input'. The 'Generate value automatically based on other values*' checkbox is highlighted with a red box, indicating it is the option to be selected for pre-populating the 'Description' field.

- Specify the value(s) that the field should be defaulted to. For this example we will add a static text string "Department " and a reference to the value of another property "Department" in the form. In this case the Department property was a customization previously added to the form.

NOTE: Adding a space ' ' at the end of the static text string adds a break between concatenated parts of the default value. If you want to add a break between two property values, you can add an additional text element configured with a ' '.

TEXT AREA

GENERAL

TEXT AREA OPTIONS

CUSTOM HANDLERS

Number of Rows

3

☐ Validate input using a regular expression

Regular Expression Type

Custom

Regular Expression

Test Input

☒ Generate value automatically based on other values*

ADD FIELD

ADD TEXT

Text*

Department

Field*

SELECT A FIELD

Required

Preview

Department

OKE

CANCEL

Pre-populate value automatically based on custom handler

This example demonstrates how to pre-populate the manager field of a user with the assistant administrator who is creating the user.

Step-by-step Guide

1. Start by customizing the User Create page and add a new Managed Object Browser Field for manager.

The screenshot shows the 'MANAGED OBJECT BROWSER' configuration page. It has three tabs: 'GENERAL', 'MANAGED OBJECT BROWSER OPTIONS', and 'CUSTOM HANDLERS'. The 'GENERAL' tab is active. It contains the following fields and options:

- Label ***: A text field containing 'Manager'.
- Attribute ***: A text field containing 'manager'.
- ☐ **This is a form only attribute**
This attribute is only available on the local form and its value will not be loaded from or submitted to the server.
- Description**: A large text area.
- ☐ **Hidden**
- ☐ **Required**
- ☐ **Extend control content to the label area (label will be hidden, if applicable)**
- ☐ **Do not retrieve the source object's current value when cloning**

At the bottom right, there are 'OK' and 'CANCEL' buttons.

2. After adding the manager field, open the Form Properties and navigate to Form Load Handlers. Add a script Handler. Click **Add > Script** to configure the script.

The screenshot shows the 'FORM LOAD HANDLERS' configuration page. It has three tabs: 'GENERAL', 'FORM LOAD HANDLERS', and 'FORM SUBMIT HANDLERS'. The 'FORM LOAD HANDLERS' tab is active. It contains the following elements:

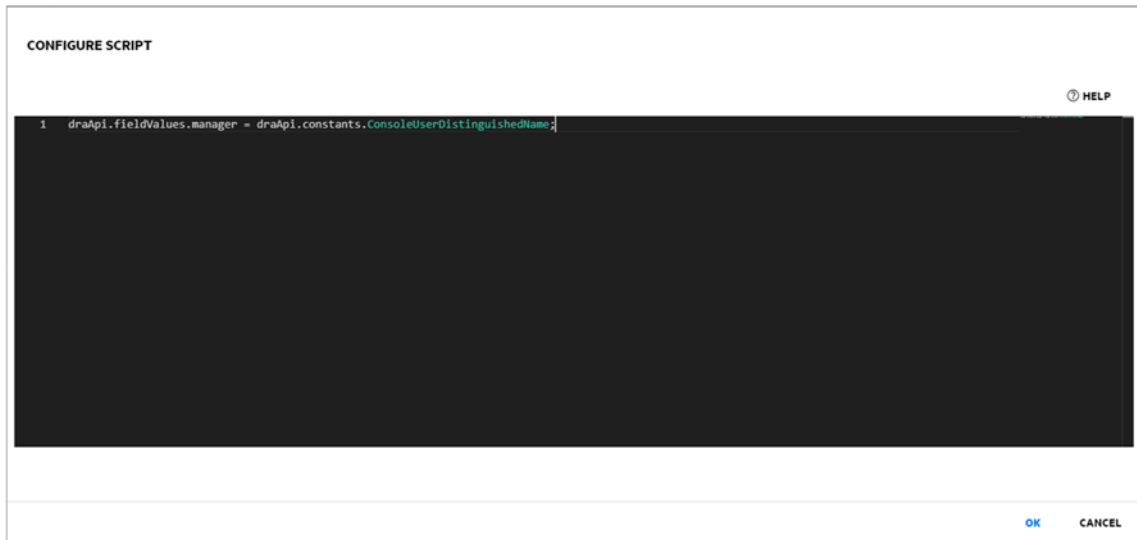
- Execute custom handlers when the form is initially loaded**: A toggle switch that is turned on.
- Custom Handlers**: A section with a note 'Custom handlers are executed in the order listed' and a table with columns 'ORDER', 'TYPE', 'BUILTIN', and 'DETAILS'. An '+ ADD' button is highlighted with a red box.
- Handler List**: A list of available handlers on the right side of the page:

Handler Name	Description
DRA Query	Submits an operation to the DRA server using the dra/operations/varset/post endpoint of the DRA REST service.
LDAP Query	Executes an LDAP query on the specified domain controller.
Message Box	Displays a message box to the user.
REST Query	Sends a payload to a RESTful interface.
Script	Executes custom JavaScript code enabling you to extend the form page's behavior using custom logic.

The 'Script' handler is highlighted with a red box.

NOTE: To allow customizations to include custom JavaScript code add `allowCustomJavaScript="true"` to the `<consoleConfiguration>` element found in `%ProgramData%\NetIQ\DRARESTProxy\restProxy.config` on the web server.

3. In Configure Script editor, assign the value of the current assistant administrator's distinguished name to the manager field. We use `draApi.fieldValues` and `draApi.constants.ConsoleUserDistinguishedName` APIs.



Configure Script

```
draApi.fieldValues.manager = draApi.constants.ConsoleUserDistinguishedName;
```


Validate input using a regular expression

The web client supports regular expressions based on JavaScript syntax, here are some reference documents:

- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions
- <http://www.regular-expressions.info/javascript.html>

NOTE:

JavaScript doesn't support the `\p` Unicode category expression, Unicode support in JavaScript's regular expressions is limited to single character matches. The following expression gets close to the behavior of `\p`: `[\w.\-]*`

NOTE:

you will not need the anchor tags since the whole expression is a single class of characters repeated any number of times. Any matching string will have to start and end with a character in that class.

Built-in example regular expressions:

Use Case	Regular Expression
All Major Credit Cards	<code>^(?:4[0-9]{12} (?:[0-9]{3})?[51-5][0-9]{14} 6011[0-9]{12} 3(?:0[0-5] [68][0-9])[0-9]{11})3[47][0-9]{13})\$</code>
Alpha-Numeric String	<code>^[a-zA-Z0-9]*\$</code>
Alpha-Numeric String With Spaces	<code>^[a-zA-Z0-9\s]*\$</code>
One Or More Digits	<code>^[0-9]+\$</code>
Email Address	<code>^[a-z0-9!#\$%&'*/=?+_-`{} ~.-]+@[a-z0-9]([a-z0-9]*[a-z0-9])?(\.[a-z0-9]([a-z0-9]*[a-z0-9])?)*)*\$</code>
Mail Alias	<code>^[a-zA-Z0-9!#\$%&'*/+_-=?^_`{} ~i-y]*\$</code>
URL	<code>^[a-z][a-z\d+~]*/(?:[^\:@]+(?:[^\@\&]*?)?)?(?:[^\s:/#\&] \\[a-f\d:~+\\])(?:\d+)?(?:/[^#]*)?(?:\?[^#]*)?(?:#.*)?\$</code>
Number	<code>^\s*(\-\ \+)?(\d+ (\d*(\.\d*)))([eE][+-]?[d+])?\s*\$</code>
ISO Date	<code>^\d{4}-[01]\d-[0-3]\dT[0-2]\d:[0-5]\d:[0-5]\d\.\d+([+-][0-2]\d:[0-5]\d Z)\$</code>
IP Address (version 4)	<code>^(?:([0-9]{1,3} 25[0-5] 2[0-4][0-9] 099 011)?[0-9]{0-3})?(\.)(?:25[0-5] 2[0-4][0-9] 099 011)?[0-9]{0-3})\$</code>
IP Address (version 6)	<code>^((((([0-9A-Fa-f]{1,4}){7}([0-9A-Fa-f]{1,4})?)) ((([0-9A-Fa-f]{1,4}){6}(:[0-9A-Fa-f]{1,4}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){5}(:([0-9A-Fa-f]{1,4}){1,2}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){4}(:([0-9A-Fa-f]{1,4}){1,3}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){3}(:([0-9A-Fa-f]{1,4}){1,2}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){2}(:([0-9A-Fa-f]{1,4}){1,1}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){1}(:([0-9A-Fa-f]{1,4}){1,1}) ((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,4}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,5}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,6}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,7}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,8}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,9}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,10}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,11}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,12}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,13}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,14}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,15}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,16}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,17}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,18}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,19}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,20}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,21}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,22}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,23}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,24}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,25}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,26}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,27}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,28}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,29}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,30}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,31}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,32}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,33}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,34}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,35}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,36}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((([0-9A-Fa-f]{1,4}){0,37}(:((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d)\.((25[0-5] 2[0-4]\d 1\d\d 1-9)?\d){3}) :)) ((</code>

This example demonstrates how to validate that a group name entered by a user begins with only specific characters, say it should begin with "GRP". When a group name is entered and if the given name doesn't begin with "GRP", then the name field will outlined with red that indicates input is invalid and the form cannot be submitted.

Step-by-step Guide

1. Start by customizing the Group Create page. Open the properties of the Name field, and select the "Validate input using a regular expression" check box.

The screenshot shows the 'TEXT' properties dialog with the 'TEXT OPTIONS' tab selected. The 'Validate input using a regular expression' checkbox is checked. The 'Regular Expression Type' is set to 'Custom'. The 'Regular Expression' field contains the value '^GRP[a-zA-Z0-9_]*\$'. The 'Match is case sensitive' checkbox is unchecked. The 'Test Input' field is empty. The 'Generate value automatically based on other values' checkbox is unchecked. The 'OK' and 'CANCEL' buttons are at the bottom right.

2. For this example we will be implementing a custom regular expression that will require the property value to begin with "GRP":
 - Set the 'Regular Expression Type' to 'Custom'
 - Enter the regular expression value `"^GRP[a-zA-Z0-9_]*$"`
 - Set the option regular expression option 'Match is case sensitive' (for strict capitalization checking)

NOTE: You can select one of the default regular expressions as a starting point or select custom to create your own.

TEXT

GENERAL **TEXT OPTIONS** CUSTOM HANDLERS

Placeholder

Maximum Length

☐ Enter value masked on screen

☒ Validate input using a regular expression

Regular Expression Type

Custom

Regular Expression

`^GRP[a-zA-Z0-9_]*$`

☒ Match is case sensitive

Test Input

☐ Generate value automatically based on other values

ADD FIELD ADD TEXT

OK CANCEL

- The regular expression can be tested by entering a test value to verify it is working as intended.

TEXT

GENERAL **TEXT OPTIONS** CUSTOM HANDLERS

Placeholder

Maximum Length

☐ Enter value masked on screen

☒ Validate input using a regular expression

Regular Expression Type

Custom

Regular Expression

`^GRP[a-zA-Z0-9_]*$`

☒ Match is case sensitive

Test Input

Passed

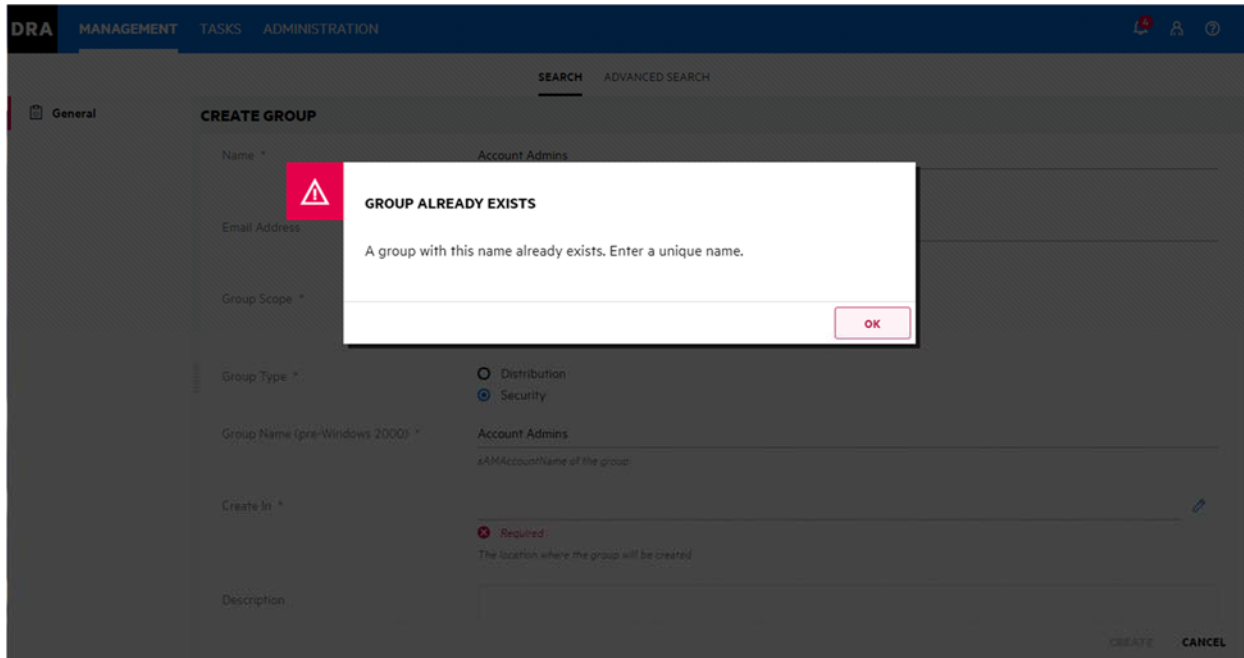
☐ Generate value automatically based on other values

ADD FIELD ADD TEXT

OK CANCEL

Validate Name Uniqueness for a Specified Field Value at Run-time

This example demonstrates how to validate that a group name entered by a user does not already exist in Active Directory at run-time. It will also demonstrate using multiple handlers in sequence and controlling the execution flow of those handlers. When a group with the given name does exist, an alert is shown to the user informing them of the issue. Note that the alert is only informational and the form can still be submitted after the message is cleared by the user.



Step-by-step Guide

1. Start by customizing the Group Create page and open the properties of the Name field. Add an LDAP Query handler on the Custom Handlers tab and set the Execution Time to execute the handler when the field loses focus (otherwise it will be executed after every key stroke). Finally, click **Add > LDAP Query** to configure the LDAP query.

WARNING: Running LDAP, REST, or DRA queries in a change handler for validation like this is generally not recommended and as a best practice should be done in the form's On Submit handler if possible.

TEXT

GENERAL TEXT OPTIONS **CUSTOM HANDLERS**

— ☒ Execute custom handlers every time the field value changes

Execution Time

Execute handler when the field loses focus

Custom Handlers

① Custom handlers are executed in the order listed

+ ADD

ORDER	TYPE	BUILTIN	DETAILS
-------	------	---------	---------

OK CANCEL

- Specify the name of the domain controller where you want the LDAP Query to execute in the Connection Information section.

CONFIGURE LDAP QUERY

② HELP

CONNECTION INFORMATION QUERY PARAMETERS PRE-QUERY ACTION POST-QUERY ACTION

Host Name * VLAB019670

LDAP Server host name or IP address

Port Number * 389

Connection Timeout (seconds) 30

☐ Use UDP connection

OK CANCEL

- Specify the Query Parameter configuration:
 - Set 'Base Distinguished Name' to the distinguished name of the domain to be searched
 - Set 'Scope' to Subtree so our search will be recursive and include all child OUs and containers in the domain
 - Set 'Attributes' to return the name value of the group
 - Leave the 'Filter' empty (we need to generate it dynamically based on the users input in the form)

CONFIGURE LDAP QUERY

?

HELP

CONNECTION INFORMATION

QUERY PARAMETERS

PRE-QUERY ACTION

POST-QUERY ACTION

Base Distinguished Name *

DC=KS,DC=LAB

The DN of the object at which to start the query (i.e. DC=MYCOMPANY,DC=COM)

Scope *

Subtree

Attributes *

name

×

Type in a value and hit Enter

Filter

OK

CANCEL

- In the Pre-Query Action we need to create the LDAP query using the name entered by the user. We use the `draApi.fieldValues` API to access the value of the name field and build the query string which we then set as the query filter using the `LdapOptions.filter` API.

NOTE: To verify uniqueness, we're only interested if the query returns 1 or more results, so we'll just have the query return the name attribute of any matching group.

NOTE: There are 3 possible options for Scope:

- Base - The search is limited to the object specified by the Base Distinguished Name. This is useful if you want to see if that object exists.
- One Level - The search is limited to the Base Distinguished Name.
- Subtree - The search is recursive and will include all children of the object specified by the Base Distinguished Name. More details can be found here: <https://docs.microsoft.com/en-us/windows/desktop/ad/search-scope>

Pre-Query Action

```
ldapOptions.filter = '(&(objectCategory=group)(objectClass=group)(name=' +
draApi.fieldValues.name + '))'
```

CONFIGURE LDAP QUERY

?

HELP

CONNECTION INFORMATION

QUERY PARAMETERS

PRE-QUERY ACTION

POST-QUERY ACTION

```
1 ldapOptions.filter = '(&(objectCategory=group)(objectClass=group)(name=' + draApi.fieldValues.name + '))';
```

OK

CANCEL

- In the Post-Query Action we'll store the number of matching objects so we can use it in a later handler. The results are returned as an array of objects that matched the query so we'll store the length of that array in a variable we can reference later using the variables API. The value passed to this API is just an arbitrary name that we decide to use that describes that information it holds.

CONFIGURE LDAP QUERY

?

HELP

CONNECTION INFORMATION

QUERY PARAMETERS

PRE-QUERY ACTION

POST-QUERY ACTION

```
1 // store the number of matches in resultCount so we
2 // can reference it in other handlers
3 Variable(resultCount) = QueryResults().length;
4 |
```

OK

CANCEL

Post-Query Action

```
// store the number of matches in resultCount so we
// can reference it in other handlers
variables.resultCount = results.length;
```

- After clicking OK to save the LDAP query, we'll now add a Message Box handler.

- Click **Add > Message Box** to configure the handler. We'll configure the message box to have a descriptive title, an informative message, and set its style to Error.

The screenshot shows the 'CONFIGURE MESSAGE BOX' dialog with the 'GENERAL' tab selected. The fields are filled as follows:

- Title ***: Group Already Exists
- Message ***: A group with this name already exists. Enter a unique name.
- Style ***: Error
- Buttons ***: OK, X, Type in a value and hit Enter

At the bottom right are 'OK' and 'CANCEL' buttons. A 'HELP' icon is in the top right corner.

- The Before-Show Action is where we can execute some custom code before the message box is shown to the user. We'll use this to check the value of our resultCount variable and then only display the message box if the resultCount is greater than 0. In order to prevent the message box from being shown when the resultCount is 0 we'll call the flowControl.exit() API which will tell the Web Console to stop processing this handler and to not process any remaining handlers if there were any.

The screenshot shows the 'CONFIGURE MESSAGE BOX' dialog with the 'BEFORE-SHOW ACTION' tab selected. The code editor contains the following JavaScript code:

```
1 if (Variable(resultCount) === 0) {
2   // if we didn't match any existing groups we can exit our handler
3   // to prevent our message box from being shown to the user
4   exit();
5 }
6
7 // execution will continue here when our query did match at least
8 // one existing group so our message box will be shown to the user
```

At the bottom right are 'OK' and 'CANCEL' buttons. A 'HELP' icon is in the top right corner.

Before-Show Action

```
if (variables.resultCount === 0) {
  // if we didn't match any existing groups we can exit our handler
  // to prevent our message box from being shown to the user
  flowControl.exit();
}

// execution will continue here when our query did match at least
// one existing group so our message box will be shown to the user
```


Validate Name Uniqueness for a Specified Field Value When a Form is Submitted

This example demonstrates how to validate that a group name entered by a user does not already exist in Active Directory when the form is submitted. When a group with the given name does exist a notification is presented to the user informing them of the issue and the form is not submitted to the server.

The screenshot shows the 'CREATE GROUP' form in the DRA application. The form fields include Name (set to 'Account Admins'), Email Address, Group Scope, Group Type (with 'Security' selected), Group Name (pre-Windows 2000), Create In (with a 'Required' error message), and Description. A warning dialog box is overlaid on the form, displaying a red triangle icon and the text: 'WARNING: A group with that name already exists, please enter a unique name.' with an 'OK' button.

Step-by-step Guide

1. Start by customizing the Group Create page and open the Form Submit Handlers tab on the Form Properties dialog. Click **Add > LDAP Query** to configure the LDAP query.

The screenshot shows the 'GROUP' Form Properties dialog, specifically the 'FORM SUBMIT HANDLERS' tab. The 'Execute custom handlers when the form is submitted' checkbox is checked. Below this, the 'Custom Handlers' section shows a list of handlers. A table with columns 'ORDER', 'TYPE', 'BUILTIN', and 'DETAILS' is visible, but it is currently empty. An '+ ADD' button is located above the table. At the bottom right of the dialog are 'OK' and 'CANCEL' buttons.

2. Specify the name of the domain controller where you want the LDAP Query to execute in the Connection Information section.

CONFIGURE LDAP QUERY HELP

CONNECTION INFORMATION QUERY PARAMETERS PRE-QUERY ACTION POST-QUERY ACTION

Host Name *
LDAP Server host name or IP address

Port Number *

Connection Timeout (seconds)

☐ Use UDP connection

OK CANCEL

3. Specify the Query Parameter configuration:

- Set 'Base Distinguished Name' to the distinguished name of the domain to be searched
- Set 'Scope' to Subtree so our search will be recursive and include all child OUs and containers in the domain
- Set 'Attributes' to return the name value of the group
- Leave the 'Filter' empty (we need to generate it dynamically based on the users input in the form)

CONFIGURE LDAP QUERY HELP

CONNECTION INFORMATION **QUERY PARAMETERS** PRE-QUERY ACTION POST-QUERY ACTION

Base Distinguished Name *
The DN of the object at which to start the query (i.e. DC=MYCOMPANY,DC=COM)

Scope *

Attributes * Type in a value and hit Enter

Filter

OK CANCEL

NOTE: To verify uniqueness, we're only interested if the query returns 1 or more results, so we'll just have the query return the name attribute of any matching group.

NOTE: There are 3 possible options for Scope:

- Base - The search is limited to the object specified by the Base Distinguished Name. This is useful if you want to see if that object exists. One Level –
- The search is limited to the immediate children of the object specified by the Base Distinguished Name.
- Subtree - The search is recursive and will include all children of the object specified by the Base Distinguished Name.

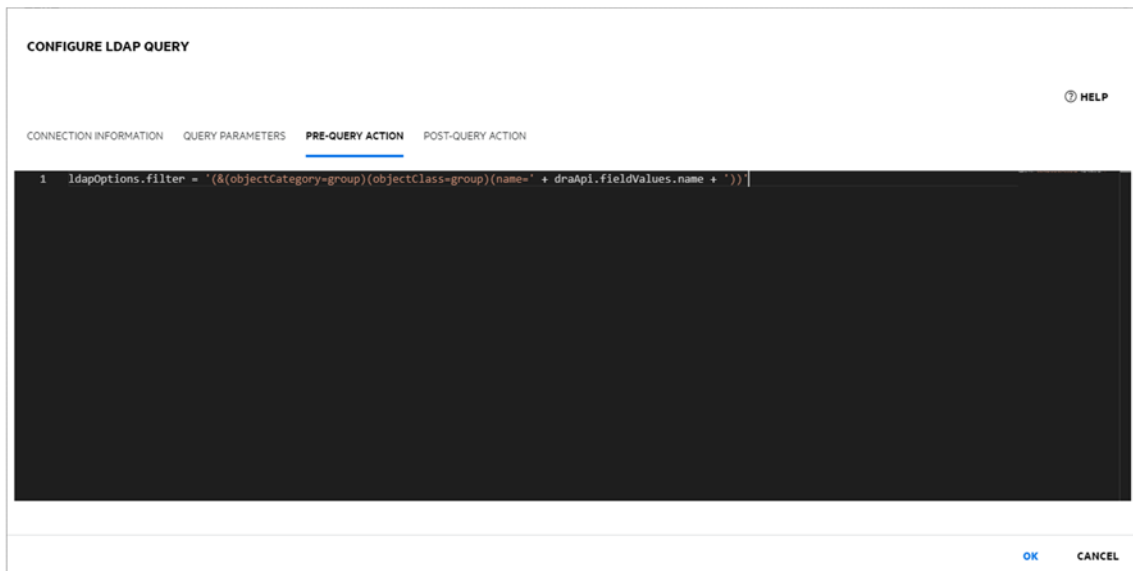
More details can be found here:

<https://docs.microsoft.com/en-us/windows/desktop/ad/search-scope>

4. In the Pre-Query Action we need to create the LDAP query using the name entered by the user. We use the `draApi.fieldValues` API to access the value of the Name field and build the query string which we then set as the query filter using the `LdapOptions.filter` API.

Pre-Query Action

```
ldapOptions.filter = '(&(objectCategory=group)(objectClass=group)(name=' +  
draApi.fieldValues.name + '))';
```



5. In the Post-Query Action we'll check the results returned by the query. The results are returned as an array of objects that matched the query so we just need to check if the length of the array is greater than 0. When a matching group is found we use the `flowControl.cancel()` API to cancel the form submission passing the API an optional message to display to the user.

Post-Query Action

```
if (results.length > 0) {  
    flowControl.cancel('A group with that name already exists, please enter a unique name.');
```

CONFIGURE LDAP QUERY

HELP

CONNECTION INFORMATION QUERY PARAMETERS PRE-QUERY ACTION **POST-QUERY ACTION**

```
1 if (results.length > 0) {  
2     flowControl.cancel('A group with that name already exists, please enter a unique name.');
```

OK CANCEL

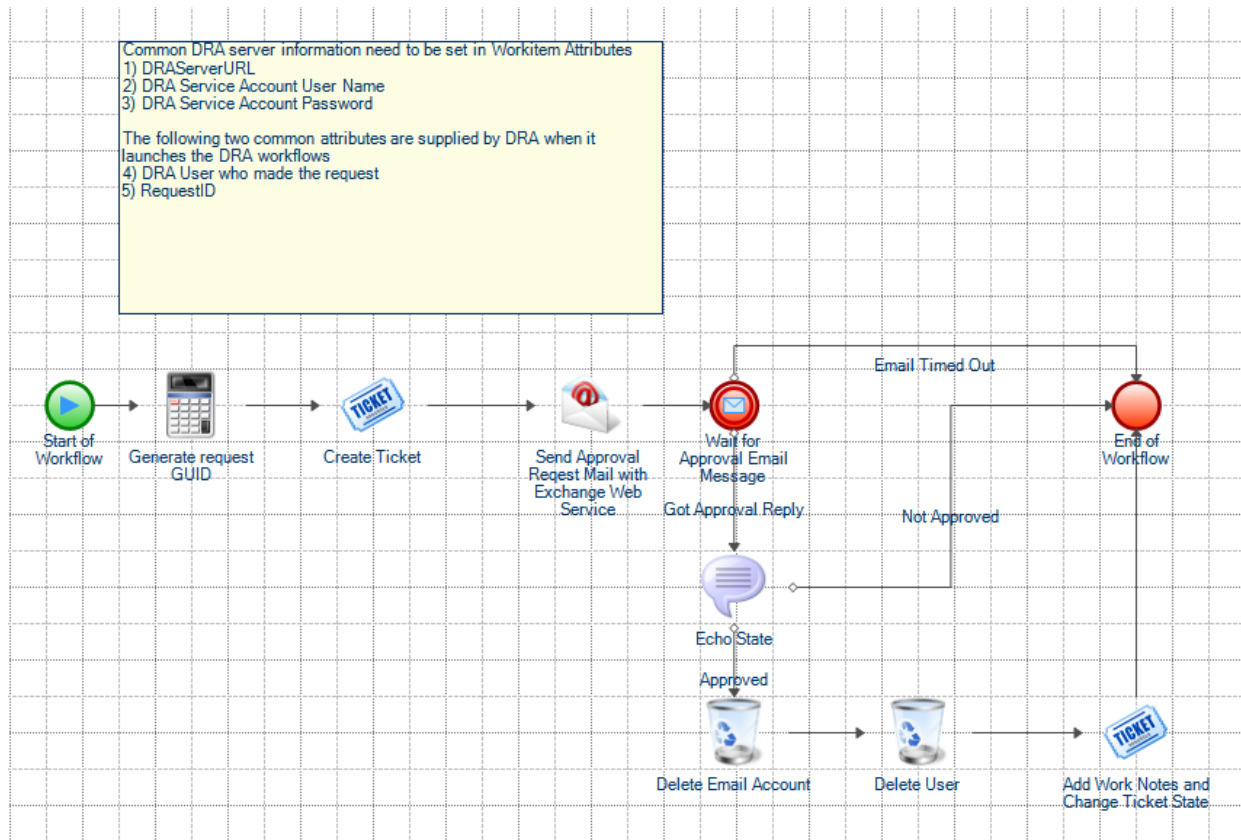
Workflow Customization

This section contains examples for customizing automated workflow forms in in the Web Console.

Custom Workflow Examples:

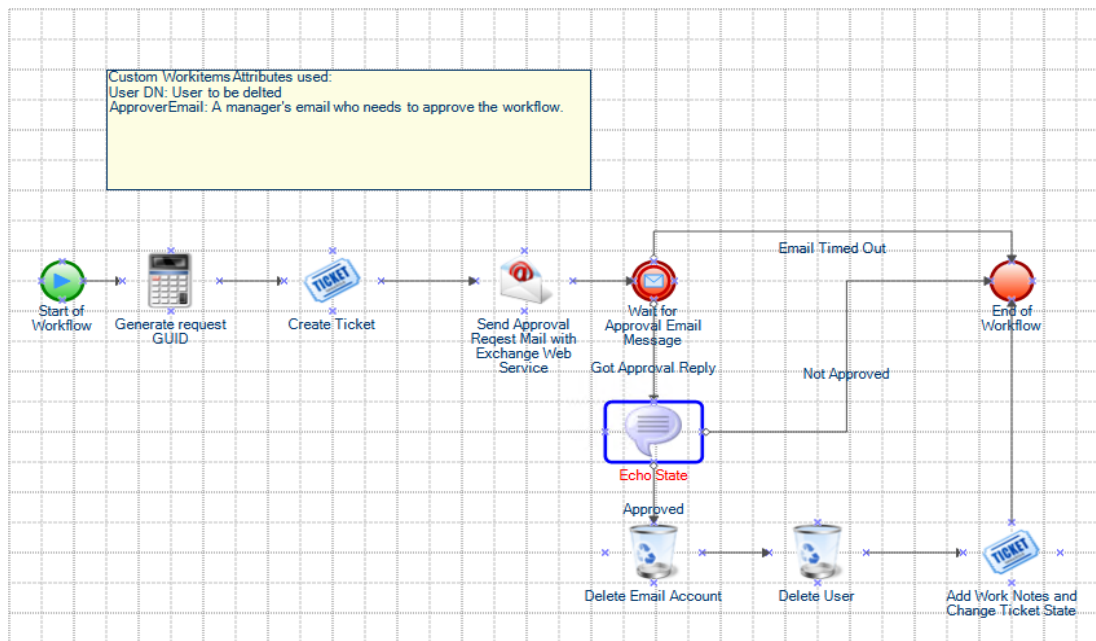
Solution	Use Case	Components for Example	Notes
This solution will be broken into 3 sub-workflows	Create and modify a user account	Legacy DRA activity Builtin DRA REST Activity Custom DRA REST (Include parsing JSON response)	Core Business logic, we may add additional use cases in the future. This workflow will have a corresponding Workflow form in the DRA web client so that it can be launched manually.
	Service Now Integration	Trigger workflow based on ServiceNow ticket Call Sub-workflow Update SericeNow with details from processing ticket	Create ServiceNow Change Request, add the request information. When approved, update ticket information and state
	Approval Workflow	Lookup group owner/manager IQMail EWS approval email request/response	Send EWS email to a user and wait for email reply.
Load users information from Human Resources database to DRA	Bulk Load User Accounts	Load HR DB to DRA	

Delete User Workflow



Custom Form Workflow to delete a user

A DRA Custom Form launch the workflow "Delete User by DN" to delete a user. The user DN is supplied by setting workflow customer workitem attribute User DN. Approver's email is also read from the work



Workitem Attributes used:

Workflow Properties

Workflow Work Item

Properties

Type: DRA_Request_Form1

Subject:

Description: DRA_Request_Form: Delete User by user DN

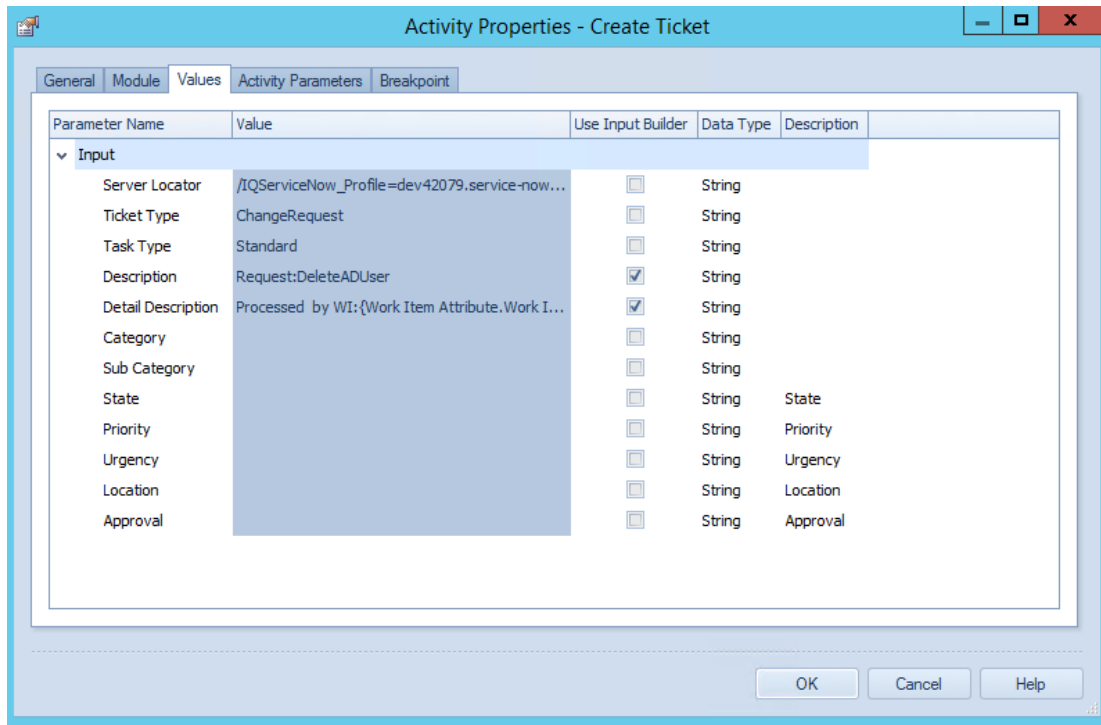
Custom Work Item Attributes (in evaluation order)

Name	Display Name	Type	Configured Value	Hidden	Read Only	Description
UserDN	User DN	String	CN=Jennifer.Whalen,OU=Seattle,OU=North A...			User DN
GroupDN	Group DN	String				Group DN
DRAServerURL	DRAServerURL	String	{Global Setting.DRA Server URL}			DRA Servder URL
AddMembersURL	AddMembersURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		
DRASvcUserName	DRA Service Account User Name	String	demo\drasvc			DRA Service Account User Name
DRASvcPassword	DRA Service Password	Password	*****			
DisableUserURL	DisableUserURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		DisableUserURL
DeleteMailboxURL	DeleteMailboxURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		DeleteMailboxURL
MoveUserURL	MoveUserURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		MoveUserURL
CreateUserURL	CreateUserURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		CreateUserURL
CreateGroupURL	CreateGroupURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		CreateGroupURL
CreateContactURL	CreateContactURL	String	{Work Item Attribute.DRAServerURL}{Work Ite...	✓		CreateContactURL
ApproverEmail	ApproverEmail	String	aegissvc@aedom310.lab			

Add... Edit... Delete Print/Export/Email... Move Up Move Down

OK Cancel Help

Create a new ServiceNow ChangeRequest ticket:

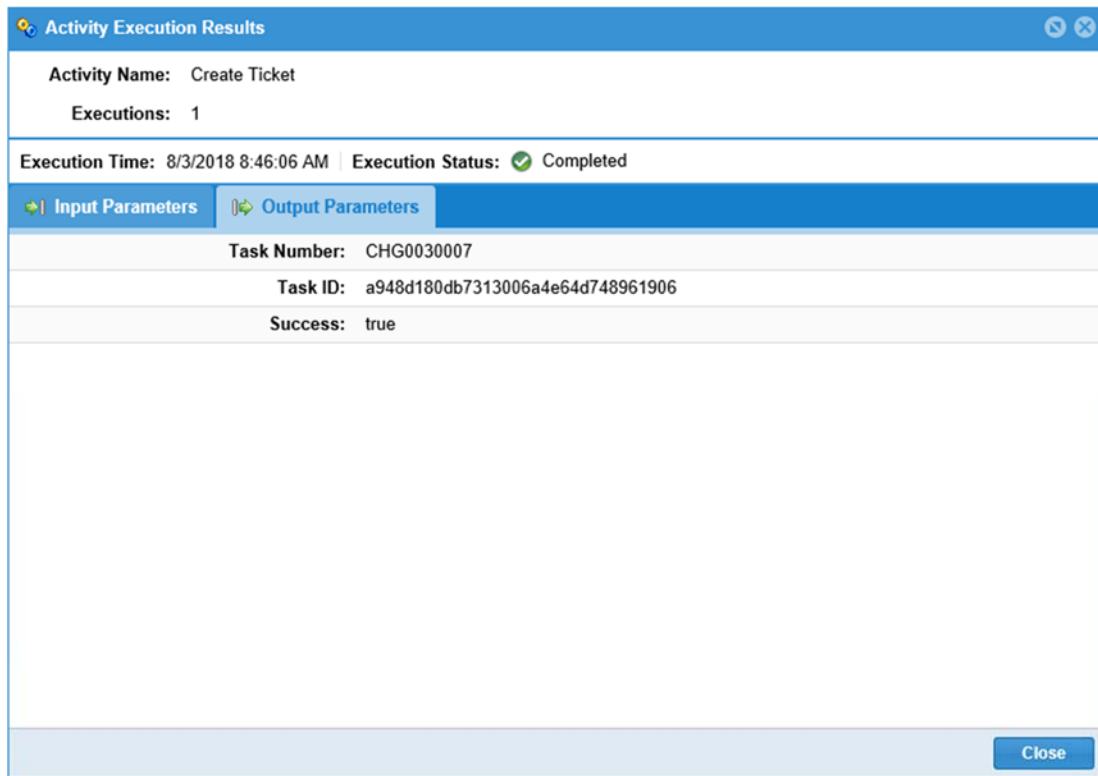


The image shows a dialog box titled "Activity Properties - Create Ticket". It has several tabs: "General", "Module", "Values", "Activity Parameters", and "Breakpoint". The "Activity Parameters" tab is selected. It contains a table with columns: "Parameter Name", "Value", "Use Input Builder", "Data Type", and "Description".

Parameter Name	Value	Use Input Builder	Data Type	Description
Input				
Server Locator	/IQServiceNow_Profile=dev42079.service-now...	☐	String	
Ticket Type	ChangeRequest	☐	String	
Task Type	Standard	☐	String	
Description	Request:DeleteADUser	☒	String	
Detail Description	Processed by WI:{Work Item Attribute.Work I...	☒	String	
Category		☐	String	
Sub Category		☐	String	
State		☐	String	State
Priority		☐	String	Priority
Urgency		☐	String	Urgency
Location		☐	String	Location
Approval		☐	String	Approval

At the bottom of the dialog are buttons for "OK", "Cancel", and "Help".

When the workflow finishes, the Task ID is returned and can be used to update ticket activity.



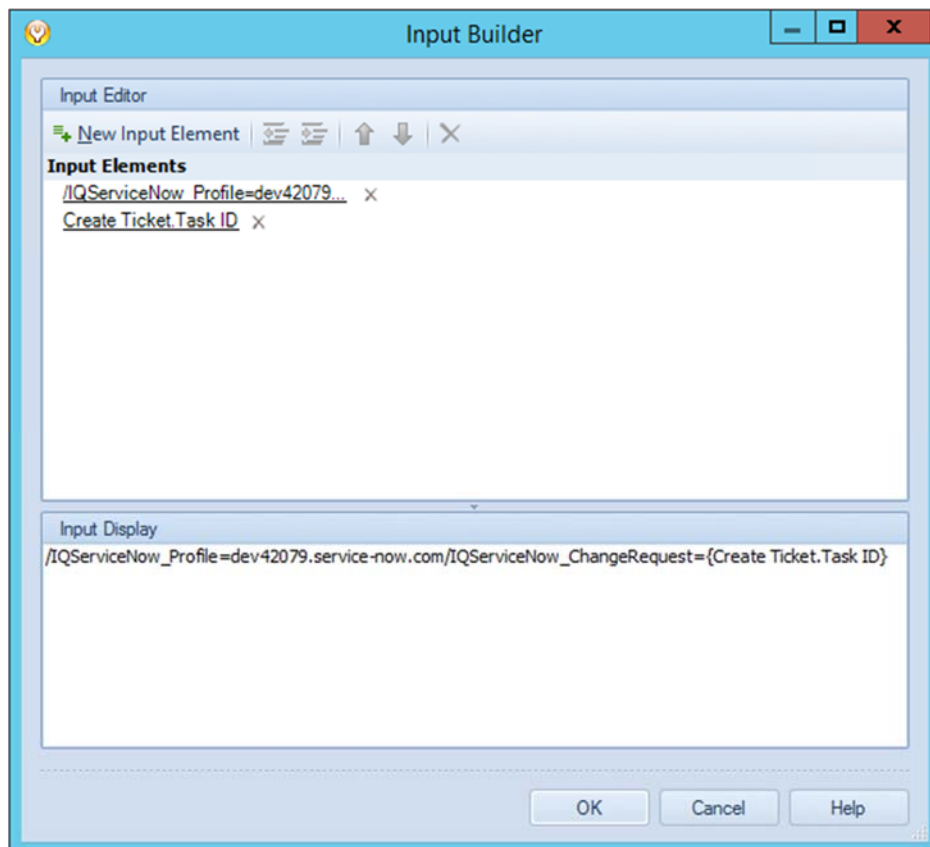
The image shows a dialog box titled "Activity Execution Results". It displays the following information:

- Activity Name:** Create Ticket
- Executions:** 1
- Execution Time:** 8/3/2018 8:46:06 AM
- Execution Status:**  Completed

Below this information are two tabs: "Input Parameters" and "Output Parameters". The "Output Parameters" tab is selected. It displays the following output:

- Task Number:** CHG0030007
- Task ID:** a948d180db7313006a4e64d748961906
- Success:** true

At the bottom right of the dialog is a "Close" button.



Lab Data Setup

User information is stored in HumanResources database. The database can be populated by the attached SQL script.

- CreateDB.sql
- CreateTables.sql
- LoadData.sql
- CreateSprocs.sql

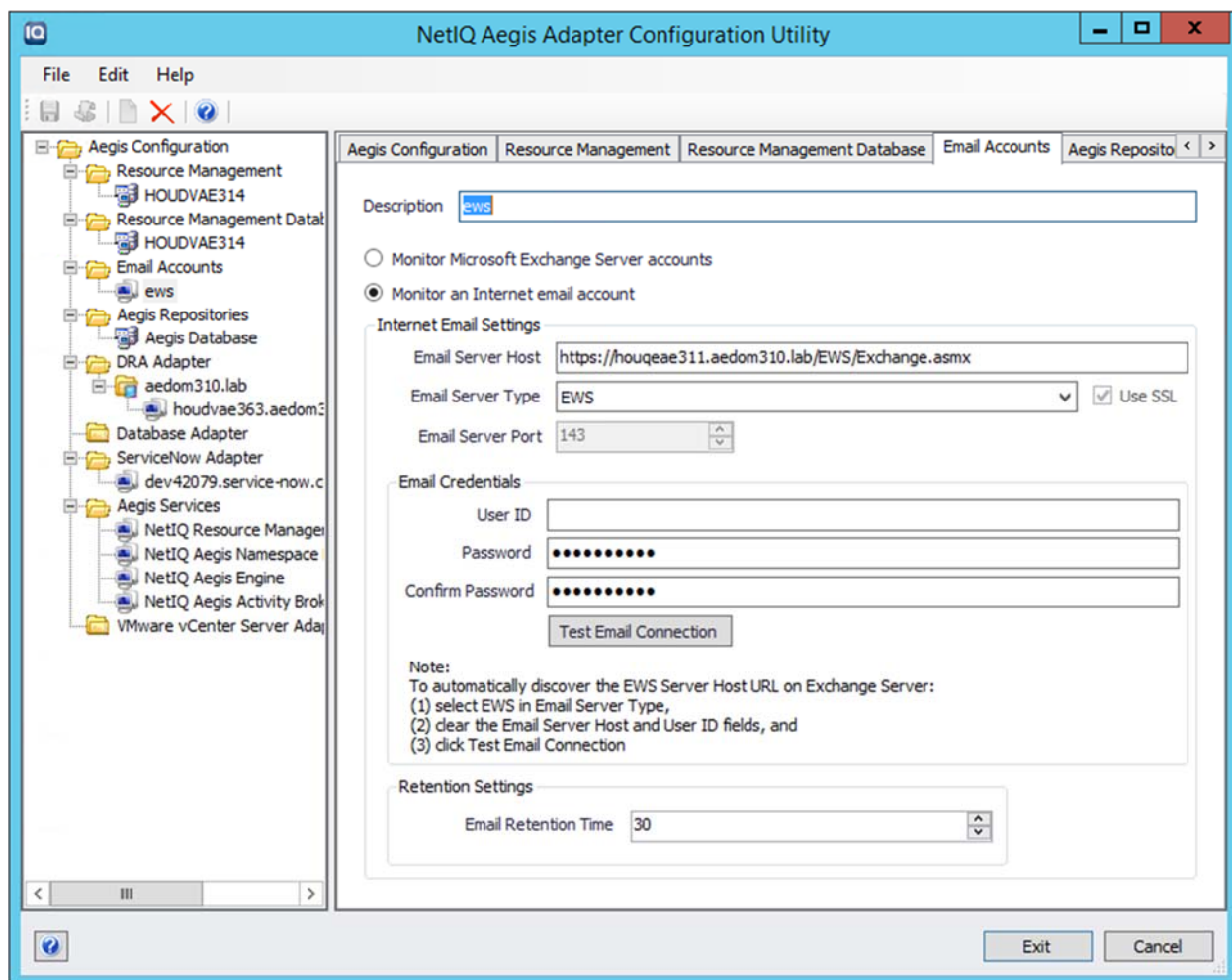
HumanResources Database

dbo.GetAllEmployeesAsJson() return all the employees as User JSON payload which can be used to call DRA Rest create user API.

```
{
  "user": {
    "container": "OU=Seattle,OU=North America,OU=Keystone Steel,DC=ks,DC=lab",
    "name": "Jennifer.Whalen",
    "firstName": "Jennifer",
    "lastName": "Whalen",
    "createEmail": true,
    "emailAddress": "jennifer.whalen@aedom310.lab",
    "department": "Administration",
    "streetAddress": "2004 Charade Rd",
    "city": "Seattle",
    "state": "Washington",
    "country": "US",
    "postalCode": "98199",
    "telephoneNumber": "515.123.4444"
  }
},
{
  "user": {
    "container": "OU=Toronto,OU=North America,OU=Keystone Steel,DC=ks,DC=lab",
    "name": "Michael.Hartstein",
    "firstName": "Michael",
    "lastName": "Hartstein",
    "createEmail": true,
    "emailAddress": "michael.hartstein@aedom310.lab",
    "department": "Marketing",
    "streetAddress": "147 Spadina Ave",
    "city": "Toronto",
    "state": "Ontario",
    "country": "CA",
    "postalCode": "M5V 2L7",
    "telephoneNumber": "515.123.5555"
  }
},
{
  "user": {
    "container": "OU=Toronto,OU=North America,OU=Keystone Steel,DC=ks,DC=lab",
    "name": "Pat.Fay",
    "firstName": "Pat",
    "lastName": "Fay",
    "createEmail": true,
    "emailAddress": "pat.fay@aedom310.lab",
    "department": "Marketing",
    "streetAddress": "147 Spadina Ave",
    "city": "Toronto",
    "state": "Ontario",
    "country": "CA",
    "postalCode": "M5V 2L7",
    "telephoneNumber": "603.123.6666"
  }
},
{
  "user": {
    "container": "OU=Seattle,OU=North America,OU=Keystone Steel,DC=ks,DC=lab",
    "name": "Den.Raphaely",
    "firstName": "Den",
    "lastName": "Raphaely",
    "createEmail": true,
    "emailAddress": "den.ra phaely@aedom310.lab",
    "department": "Purchasing",
    "streetAddress": "2004 Charade Rd",
    "city": "Seattle",
    "state": "Washington",
    "country": "US",
    "postalCode": "98199",
    "telephoneNumber": "515.127.4561"
  }
}
```

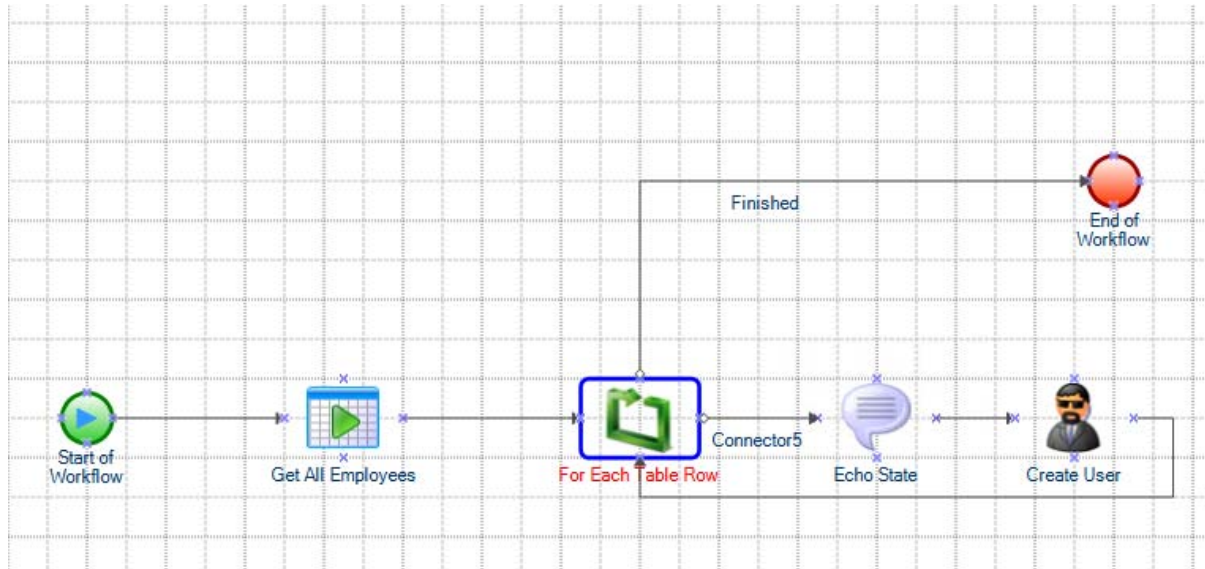
IQMail setup

If Email Server Host is blank, Select Email Server Type to EWS, Test Email Connection will autodiscover the exchange server URL. If user ID and password word is left blank, Aegis server account is going to be used to connect to the exchange server.



Load HR DB to DRA

The workflow calls HumandResources sproc dbo.GetAllEmployeesAsJson to query all users information and returns json format.



Get All Employees:

The screenshot shows the 'Activity Properties - Get All Employees' dialog box. It has tabs for 'General', 'Module', 'Values', 'Supporting Analysis Location', 'Activity Parameters', and 'Breakpoint'. The 'General' tab is selected, showing a table of parameters.

Parameter Name	Value	Use Input Builder	Data Type	Description
Configuration				
Specify Custom Connection String	☒	☐	Boolean	Select the check box to specify a custom connect
Connection String	Provider=SQLNCLI11;Integrated S...	☐	String	Connection string to connect to the database
Specify Database Type	☒	☐	Boolean	Select the check box to specify the database typ
Database Type	SQL Server	☐	String	Type of database to which you want to connect
Server Name	localhost	☐	String	Name of the server on which the database is loca
Server Authentication	☒	☐	Boolean	Select the check box to use authentication to cor
Authentication Type	SQL	☐	String	Type of authentication
Login	sa	☐	String	Login credentials to connect to the database
Password	*****...	☐	String	Password to connect to the database
Database Name	HumanResources	☐	String	Name of the database to which you want to conn
Operation	Execute stored procedure	☐	String	Database operation you want to perform
Value		☐	String	Values to be used by the specified operation
Stored Procedure Name	[dbo].[GetAllEmployeesAsJson]	☐	String	Name of the stored procedure to execute (applie
Save Query Results	☐	☐	Boolean	Select the check box to save the query results to
File Path		☐	String	Path and file name where you want to save the c
Truncate	☐	☐	Boolean	Select the check box to truncate the file before v
Save to CSV	☐	☐	Boolean	Save the results of the SQL query to a .csv file
Separator		☐	String	Character to separate values in the .csv file
Return Type	ResultSet	☐	String	Return type for the query results
Send to Supporting Analysis	☐	☐	Boolean	Select the check box to send the results of the d
Timeout	0	☐	Signed Integer (32 bit)	Length of time (in seconds) to wait for the opera

At the bottom of the dialog box are 'OK', 'Cancel', and 'Help' buttons.

Create User:

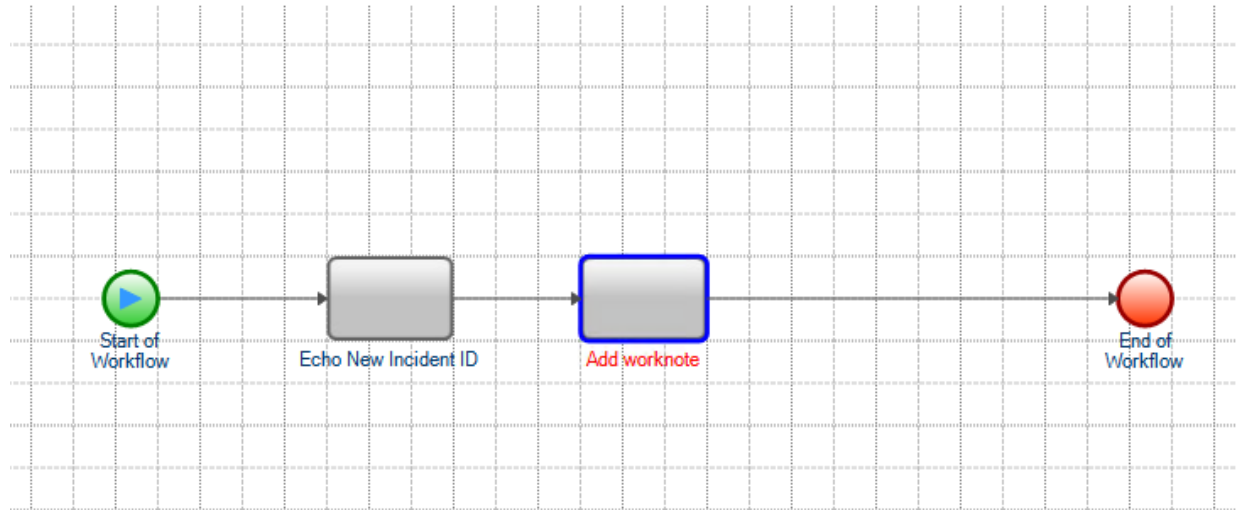
Activity Properties - Create User

General Module Values Activity Parameters Breakpoint

Parameter Name	Value	Use Input Builder	Data Type	Default Value
Configuration				
URL	{Global Setting.DRA Server URL}-{Global Setting.DRA Domain Name}/users/post	☒	String	URL
Request Method	POST	☐	String	Type
Request Timeout	0	☐	Unsigned Integer (32 bit)	Number
Use Authentication	☒	☐	Boolean	Use
Use Integrated Windows Authentication	☐	☐	Boolean	Use
Use Username/Password Authentication	☒	☐	Boolean	Use
User Name	{Global Setting.DRA Service Account Username}	☒	String	Use
Password	{Global Setting.DRA Service Account Password}	☒	String	Password
Disable Server Certificate Validation	☒	☐	Boolean	If
POST Data	{(For Each Table Row, Output Table).GetCell(1,1)}	☒	String	Data
Content Type	application/json	☐	String	Type
HTTP Session				
Cookies	[Count: 0]	☐	String Array	List
Headers	[Count: 0]	☐	String Array	List
Proxy Settings				

OK Cancel Help

ServiceNow Event Trigger



Trigger:

Activity Properties - Start of Workflow

General Trigger

Associated Triggers (in evaluation order)

Name	Type	Revision	Description
Manual Trigger	Manual	1.0	Trigger the workflow manually by user or activity
ServerNowEventTrigger	User Defined	1.*	

Add

View Trigger Revision

General

Trigger Name: ServerNowEventTrigger

Trigger Revision Version: 1.0

Description:

Previous Revision Next Revision

Actions

Initiate New Work Item If Block New Work Item If Append to Previous Work Item If Custom Attributes

Has All of the following (Time window: 2 Minutes)

- One ServiceNow.Event event

How to get event locator:

