

Visibility and Permission Script Builder Extension

User Guide

Version 1.0.0



[Software Change Log 2](#_Toc5899885)

[Introduction and Installation 2](#_Toc5899886)

[About the Visibility and Permission Manager Extension 2](#_Toc5899887)

[Installing the Visibility and Permission Manager Extension 3](#_Toc5899888)

[Usage 4](#_Toc5899889)

[Known Limitations 10](#_Toc5899890)

[Compatibility 11](#_Toc5899891)

[Document Revision History 11](#_Toc5899892)

# Software Change Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Release Date** | **Changes** | **Contributors** |
| 1.0 | 09/04/2019 | Initial Release | Vladimir Rosu |

# Introduction and Installation

Extensibility is a core aspect of the architecture and design of ThingWorx. Partners, third parties, and general ThingWorx users can easily add new functionality into the system, seamlessly. Extensions can be in the form of Service (function/method) Libraries, Connector Templates, Widgets, and more.

This document provides installation and usage instructions for the Visibility and Permission Manager Extension.

## About the Visibility and Permission Manager Extension

This extension has been designed to allow rapid propagation of Visibility and Permissions between instances. This process is a part of the installation of a ThingWorx Application in a new ThingWorx server.

While there are many ways to propagate the visibility and permissions (eg: writing this setup script in Java or Javascript), this Extension reduces the time needed to do that.

It does this by allowing the Developer to write his Visibility and Permission related operations in a structured format in a DataTable, that will be then used by the provided services.

The Visibility and Permission Manager Extension contains the following entities:

|  |  |
| --- | --- |
| **Entity** | **Description** |
| GitBackup.VisibilityAndPermissionSettings.Thing | Contains 3 services:  -SetVisibility (Version(STRING))  -SetRunTimePermissions (Version(STRING))  -SetDesignTimePermissions(Version(STRING)) |

Note: the Extension contains an additional 3 DataShapes that should not be modified/used by the User.

## Installing the Visibility and Permission Manager Extension

|  |  |
| --- | --- |
| 1. From a web browser, launch ThingWorx. 2. Log into ThingWorx as an administrator. |  |
| 1. Go to **Import/Export > Import**. |  |
| 1. Select Import Option: Extension and Browse and select GitBackupExtensionPack.zip 2. Click **Import**.  Note: If an **Import Successful** message does not display, contact your ThingWorx System Administrator. |  |
| 1. Click **Yes** to refresh Composer after importing the final extension. |  |
| 1. Confirm that the Extension has been imported properly. Check the Application Log for potential problems. |  |

# Usage

Usage of the Visibility and Permission Manager Extension requires 2 steps:

1. Defining the Visibility and Permissions operations using the DataTables Home Mashup.

We chose the DataTables as a storage area for the operations since they have the Home Mashup available as a rapid data entry User Interface.

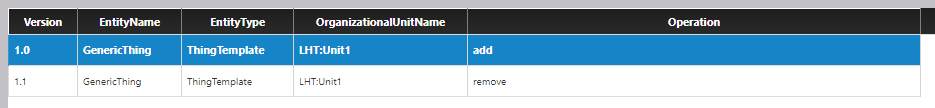
1. We run the 3 services from the GitBackup.VisibilityAndPermissionSettings.Thing that will execute the operations we defined in the DataTables

**Managing Visibility**:

1. Setup Visibility operations

Access the GitBackup.Visibility.DT DataTable, and use the Mashup section to add rows in this table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Possible Values** | **Description** |
| Version | STRING | Any string | This can be any string that represents the application version for which you want to setup the instance, usually 1.0/1.1 and so on. The table should store all the visibility operations for each application version.  The version is the only parameter of the SetVisibility service. |
| EntityName | STRING | Any string | This is the entity name. |
| EntityType | STRING | Project  Thing  ThingTemplate  ThingShape  DataShape  ModelTag  Network  IndustrialConnection  IntegrationConnector  Mashup  Master  MashupTemplate  Gadget  Dashboard  Menu  MediaEntity  StyleDefinition  StyleTheme  StateDefinition  DataTable  Stream  ValueStream  DataTag  PersistenceProvider  Blog  Wiki  UserGroup  User  Organization  ApplicationKey  DirectoryService  Authenticator  LocalizationTable  Resource  Subsystem | Entity type is required since you can have entities with the same name but of different types (eg: you can have a MediaEntity with the same name as a Thing).  **The EntityType is case sensitive.** |
| OrganizationalUnitName | STRING |  | Insert the Organizational Unit Name that will be used for visibility. Example: LHT:Unit1, where LHT is the Org Name and Unit1 is the OU name.  **Assigning Visibility for an entire Organization is not supported.** |
| Operation | STRING | add  remove | **Operation name is case insensitive.** |

An example for filling up this table is shown below:

1. Execute the SetVisibility service from the GitBackup.VisibilityAndPermissionSettings.Thing. Input the Version as a parameter, and the operations associated with that version will be executed.

**Managing RunTime Permissions**:

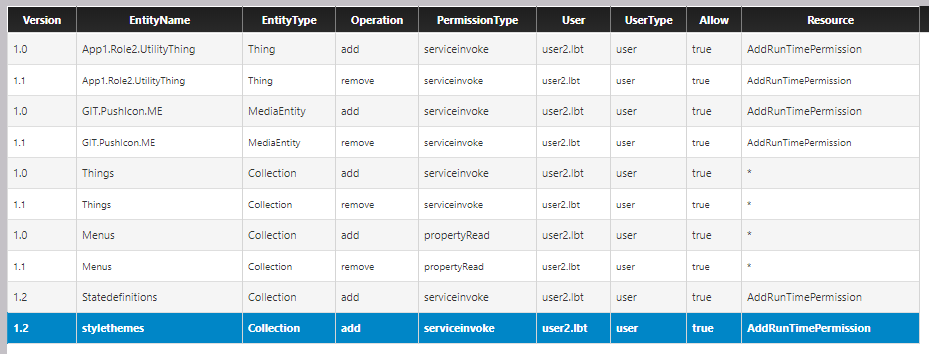
1. Setup Runtime Permissions operations

Access the GitBackup.RunTimePermission.DT, and use the Mashup section to add rows in this table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Possible Values** | **Description** |
| Version | STRING | Any string | This can be any string that represents the application version for which you want to setup the instance, usually 1.0/1.1 and so on. The table should store all the visibility operations for each application version.  The version is the only parameter of the SetVisibility service. |
| EntityName | STRING | Any string  **OR**  **If EntityType is Collection** **only the following values**:  things  thingtemplates  thingshapes  datashapes  networks  modeltags  notifications  mashups  dashboards  menus  mediaentities  styledefinitions  statedefinitions  stylethemes  datatags  persistenceproviders  usergroups  users  organizations  applicationkeys  directoryservices  authenticators  localizationtables  resources  subsystems  logs | This is the entity name OR the Collection Name.  This column has a dual use. It behaves as a normal EntityName entry field for normal entities, but, in case you want to perform operations on Collections, it accepts only specific values, the Collection Names. |
| EntityType | STRING | Project  Thing  ThingTemplate  ThingShape  DataShape  ModelTag  Network  IndustrialConnection  IntegrationConnector  Mashup  Master  MashupTemplate  Gadget  Dashboard  Menu  MediaEntity  StyleDefinition  StyleTheme  StateDefinition  DataTable  Stream  ValueStream  DataTag  PersistenceProvider  Blog  Wiki  UserGroup  User  Organization  ApplicationKey  DirectoryService  Authenticator  LocalizationTable  Resource  Subsystem  **OR**  Collection | Entity type is required since you can have entities with the same name but of different types (eg: you can have a MediaEntity with the same name as a Thing).  **Case sensitive.** |
| Operation | STRING | add  remove | **Case insensitive.** |
| PermissionType | STRING | propertyread  serviceinvoke  propertywrite  eventinvoke  eventsubscribe | This is the permission type. To set multiple permission types for the same entity you need to add multiple rows.  **Case insensitive.** |
| User | STRING | Any string | Must be a valid User or Group.  **Case sensitive.** |
| UserType | STRING | user  group | **Case insensitive.** |
| Allow | BOOLEAN | True/False | This determines if access will be allowed or forbidden for that specific resource. |
| Resource | STRING | Any string | Must be a valid service or property name.  **In case of collections, this field must be set to \***.  **Case sensitive.** |

1. Execute the SetRunTimePermissions service from the GitBackup.VisibilityAndPermissionSettings.Thing. Input the Version as a parameter, and the operations associated with that version will be executed.

An example for filling up this table is shown below:



**Managing DesignTime Permissions**

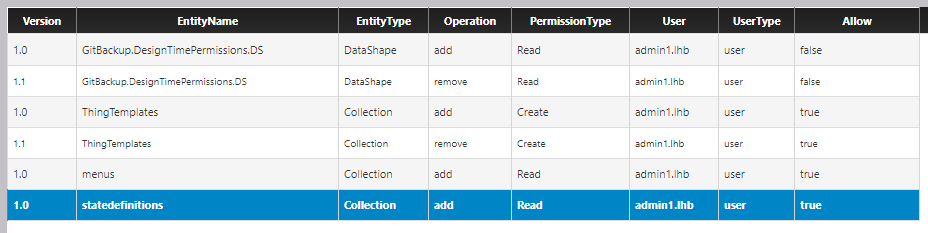
1. Setup DesignTime Permissions operations:

Access the GitBackup.DesignTimePermission.DT, and use the Mashup section to add rows in this table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Possible Values** | **Description** |
| Version | STRING | Any string | This can be any string that represents the application version for which you want to setup the instance, usually 1.0/1.1 and so on. The table should store all the visibility operations for each application version.  The version is the only parameter of the SetVisibility service. |
| EntityName | STRING | Any string  **OR**  **If EntityType is Collection** **only the following values**:  things  thingtemplates  thingshapes  datashapes  networks  modeltags  notifications  mashups  dashboards  menus  mediaentities  styledefinitions  statedefinitions  stylethemes  datatags  persistenceproviders  usergroups  users  organizations  applicationkeys  directoryservices  authenticators  localizationtables  resources  subsystems  logs | This is the entity name OR the Collection Name.  This column has a dual use. It behaves as a normal EntityName entry field for normal entities, but, in case you want to perform operations on Collections, it accepts only specific values, the Collection Names. |
| EntityType | STRING | Project  Thing  ThingTemplate  ThingShape  DataShape  ModelTag  Network  IndustrialConnection  IntegrationConnector  Mashup  Master  MashupTemplate  Gadget  Dashboard  Menu  MediaEntity  StyleDefinition  StyleTheme  StateDefinition  DataTable  Stream  ValueStream  DataTag  PersistenceProvider  Blog  Wiki  UserGroup  User  Organization  ApplicationKey  DirectoryService  Authenticator  LocalizationTable  Resource  Subsystem  **OR**  Collection | Entity type is required since you can have entities with the same name but of different types (eg: you can have a MediaEntity with the same name as a Thing).  **Case sensitive.** |
| Operation | STRING | add  remove | **Case insensitive.** |
| PermissionType | STRING | create  read  update  delete | This is the permission type. To set multiple permission types for the same entity you need to add multiple rows.  **Case insensitive.** |
| User | STRING | Any string | Must be a valid User or Group.  **Case sensitive.** |
| UserType | STRING | user  group | **Case insensitive.** |
| Allow | BOOLEAN | True/False | This determines if access will be allowed or forbidden for that specific resource. |

1. Execute the SetDesignTimePermissions service from the GitBackup.VisibilityAndPermissionSettings.Thing. Input the Version as a parameter, and the operations associated with that version will be executed.

An example for filling up this table is shown below:



# Known Limitations

|  |
| --- |
| Adding operations is only done through the DataTable’s HomeMashup, and there is no way to do syntax checking there. As a result, you need to be careful when adding operations and double check the row contents, as some fields are **not** case insensitive. |
| Visibility operations are not supported for Collections. |

# Compatibility

This extension was tested for compatibility with the following ThingWorx Platform version(s) and Operating System(s):

|  |  |
| --- | --- |
| ThingWorx Platform Version | ThingWorx 8.3.1, 8.2.3, 8.4.0, 8.4.1, 8.4.2 |
| OS | Windows 10 64 bit |

# Document Revision History

|  |  |  |
| --- | --- | --- |
| Revision Date | Version | Description of Change |
| December 20, 2017 | 1.0 | Initial version |
| December 27, 2017 | 1.1 | Added documentation for Branch and Checkout related functionality |
| January 25, 2018 | 1.2 | Updated software changelog and known limitations |
| February 1st, 2018 | 1.2.1 | Updated software changelog |
| April 26th, 2018 | 1.3.0 | Added documentation for the Git Status functionality. Updated software changelog. |
| 04 May 2018 | 1.3.1 | Updated software changelog |
| 11 January 2019 | 2.0.0 | Updated software changelog, updated guide with v. 2.0.0 UI, added contributor list to the changelog |
| 11 February 2019 | 2.0.1 | Updated software changelog |
| 29 March 2019 | 2.0.2 | Updated software changelog  Updated documentation |