Atlas -  Object History Management

**Atlas Copco**

REL-0001263v1

Prepared by Ashok Prasad

16-Apr-2020

ashok.prasad@razorleaf.eu

This document is provided in the strict confidence to Atlas Copco and its subsidiaries. It may not be shared, copied or distributed, in hardcopy or electronically to any other third-party company or subcontractor without the prior written permission of a director of Razorleaf Europe Limited.

Contents

[Contents 2](#_Toc38019070)

[1 Introduction 3](#_Toc38019071)

[2 Monitor History Setup 3](#_Toc38019072)

[2.1 Build Deployment 3](#_Toc38019073)

[2.2 Processor Setup 4](#_Toc38019074)

[3 File checkout Event 7](#_Toc38019075)

[4 Monitoring through the Web UI 8](#_Toc38019076)

[5 3dxprocessor 9](#_Toc38019077)

[5.1 History Purge Processor 10](#_Toc38019078)

[5.1.1 HistoryPurgeProcessorQueueObject 11](#_Toc38019079)

[5.1.2 HistoryPurgeProcessor 11](#_Toc38019080)

[5.1.3 HistoryPurgeProcessorTrigger 12](#_Toc38019081)

[5.2 History Publish Processor 13](#_Toc38019082)

[5.2.1 HistoryPublishProcessorQueueObject 13](#_Toc38019083)

[5.2.2 HistoryPublishProcessor 14](#_Toc38019084)

[5.2.3 HistoryPublishProcessorTrigger 15](#_Toc38019085)

1. Introduction

Object History Management is the event-based mechanism, to get the details of the object’s history. It helps the admin for monitoring an event performed by the PLM users.

A custom object will create for each event with the following details.

**Type**: History Object

**Name**: USERID (Event user)

**Revision**: Timestamp.

**Object Basic/Attribute Details**:

Originated: 12/11/2020 10:22 AM

Event: Checkout

ObjectDisplayName: Title

ObjectId: 1112.2323.1212.1121

Publish: FALSE

Site: SITE1

Type: gapGAPSpecification

Cron job runs every day early morning (Configurable) to process, delete/purge the History Object. and process publish History Object to DB/XML/CSV.

The scope of this document includes detailed documentation on of Object History Management.

1. Monitor History Setup
   1. Build Deployment

* Export Type Schema of (CAD Drawing, CAD Model, Drawing Print, Part Specification) Using 3dxdesigner.
* Compare with MonitorHistoryPackage Schema Folder and add only Delta Changes. i.e. add below content under triggers of each type.

<trigger>

<triggerType>**Checkout**</triggerType>

<programName>**emxTriggerManager**</programName>

<input>**TypeAllCreateHistoryObject**</input>

<programType>**Action**</programType>

</trigger>

* Open 3dxdesigner navigate to Shema Folder inside MonitorHistoryPackage and Select DRLMonitorHistoryPackage.xml and Analyse and Deploy Build
* Create DB Schema Using historymonitor\_2019x\_v1.sql either by running below command in CMD (**in Admin Mode**) or by copy pasting content file in Microsoft SQL Server Management Studio.



sqlcmd -S localhost -i [\\CheckoutHIstoryBuild\Configuration\historymonitor\_2019x\_v1.sql](file:///\\CheckoutHIstoryBuild\Configuration\historymonitor_2019x_v1.sql)

* Merge properties content from Modifications folder to 3dspace([\\3DSpace\win\_b64\code\tomee\webapps\3dspace\WEB-INF\classes](file:///\\3DSpace\win_b64\code\tomee\webapps\3dspace\WEB-INF\classes)) and internal([\\3DSpace\win\_b64\code\tomee\webapps\internal\WEB-INF\classes](file:///\\3DSpace\win_b64\code\tomee\webapps\internal\WEB-INF\classes))

drV6ToolsStringResource.properties

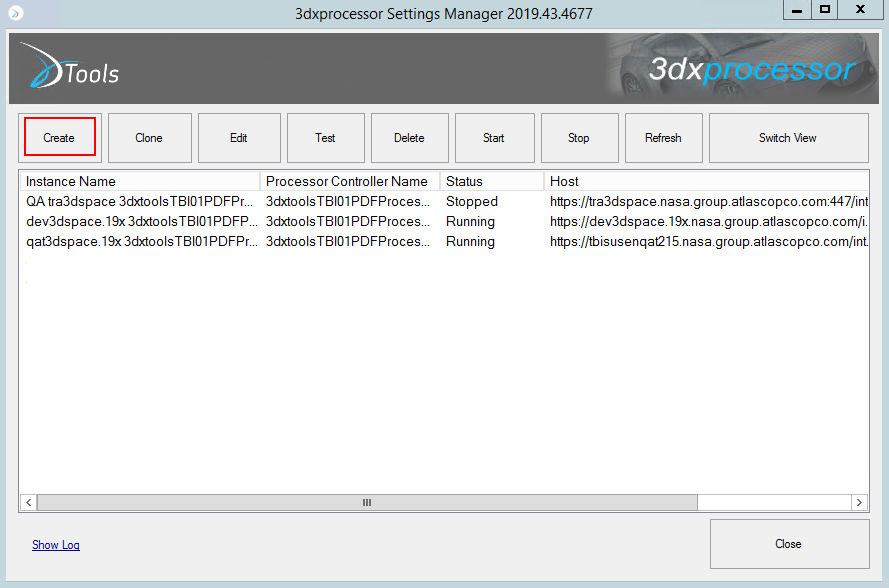
emxFrameworkStringResource.properties

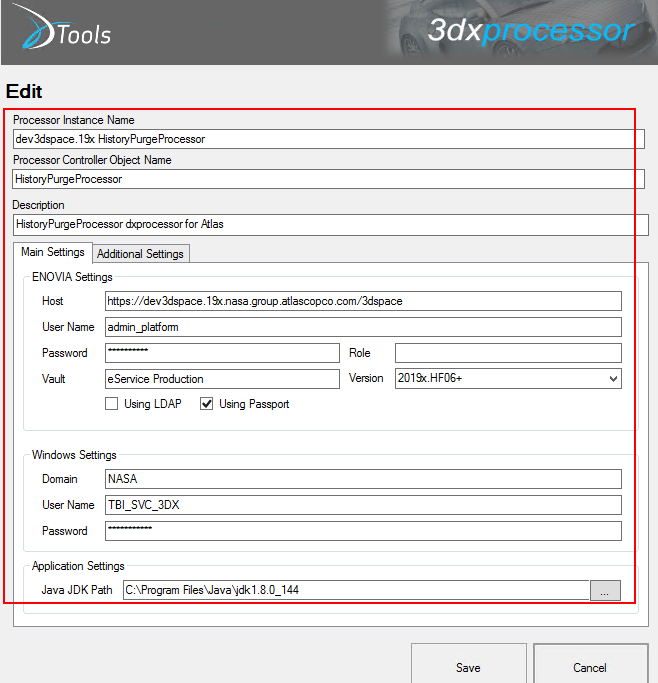
emxFrameworkStringResource\_en.properties

* Run below MQL query

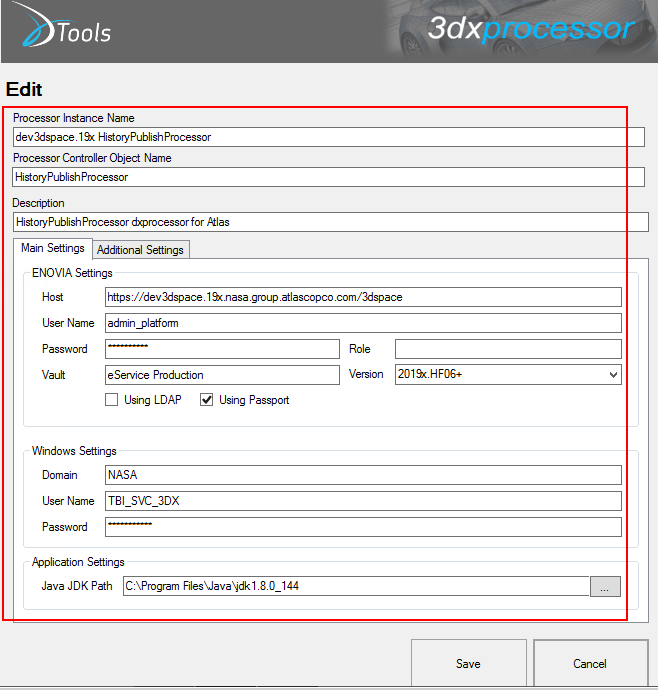
MQL> add bus 'eService Trigger Program Parameters' 'TypeAllCreateHistoryObject' 'createHistoryObject' policy 'eService Trigger Program Policy' vault 'eService Administration' current Active 'eService Method Name' createHistoryObject 'eService Program Name' DRLCaptureHistory 'eService Sequence Number' 1 'eService Error Type' Error 'eService Program Argument 1' '${OBJECTID}' 'eService Program Argument 2' '${EVENT}' 'eService Program Argument 3' '${TYPE}';

* Restart “3DEXPERIENCE R2019x 3DSpace TomEE” and “3DEXPERIENCE R2019x 3DSpaceNoCAS TomEE” Services.
  1. Processor Setup
* Create two Processor by name HistoryPurgeProcessor and HistoryPublishProcessor using 3dxprocessor, fill details according to below screenshot





* Fill up the Details Accordingly
* Processor Instance Name: dev3dspace.19x HistoryPurgeProcessor (Environment Processor Name)
* Processor Controller Object Name: HistoryPurgeProcessor
* Description: HistoryPurgeProcessor dxprocessor for Atlas
* Enovia Settings
  + Host: <https://dev3dspace.19x.nasa.group.atlascopco.com/3dspace> (3DSpace URL)
  + Username: admin\_platform
  + Password: \*\*\* (admin\_platform Password)
  + Vault: eService Production
  + Version: 2019x.HF06+ (Select Application Version)
  + Check Using Passport
* Windows Settings
  + Domain: NASA
  + Username: TBI\_SVC\_3DX
  + Password: \*\*\* (Above Mentioned user password)
  + Java JDK Path: C:\Program Files\Java\jdk1.8.0\_144 (Provide JDK path if Empty/Change)



* Fill up the Details Accordingly
* Processor Instance Name: dev3dspace.19x HistoryPublishProcessor (Environment Processor Name)
* Processor Controller Object Name: HistoryPublishProcessor
* Description: HistoryPublishProcessor dxprocessor for Atlas
* Enovia Settings
  + Host: <https://dev3dspace.19x.nasa.group.atlascopco.com/3dspace> (3DSpace URL)
  + Username: admin\_platform
  + Password: \*\*\* (admin\_platform Password)
  + Vault: eService Production
  + Version: 2019x.HF06+ (Select Application Version)
  + Check Using Passport
* Windows Settings
  + Domain: NASA
  + Username: TBI\_SVC\_3DX
  + Password: \*\*\* (Above Mentioned user password)
  + Java JDK Path: C:\Program Files\Java\jdk1.8.0\_144 (Provide JDK path if Empty/Change)
* After Create Select Processor Click on Test
* If Test Complete Successfully then Processor Configured Properly else fix and Proceed further.
* Copy HistoryPurgeProcessor.jar from Modifications folder to 3DXProcessor setup locations ([\\Designrule\3dxprocessor\drV6Tools.JobServer.Service.Instances\HistoryPurgeProcessor\lib](file:///\\Designrule\3dxprocessor\drV6Tools.JobServer.Service.Instances\HistoryPurgeProcessor\lib))
* Copy HistoryPublishProcessor.jar and mssql-jdbc-8.2.2.jre8.jar from Modifications folder to 3DXProcessor setup locations ([\\Designrule\3dxprocessor\drV6Tools.JobServer.Service.Instances\HistoryPublishProcessor\lib](file:///\\Designrule\3dxprocessor\drV6Tools.JobServer.Service.Instances\HistoryPublishProcessor\lib))
* Select each processor and Click on Start to run processor regularly.

1. File checkout Event

An action trigger will execute on the checkout event for the following type

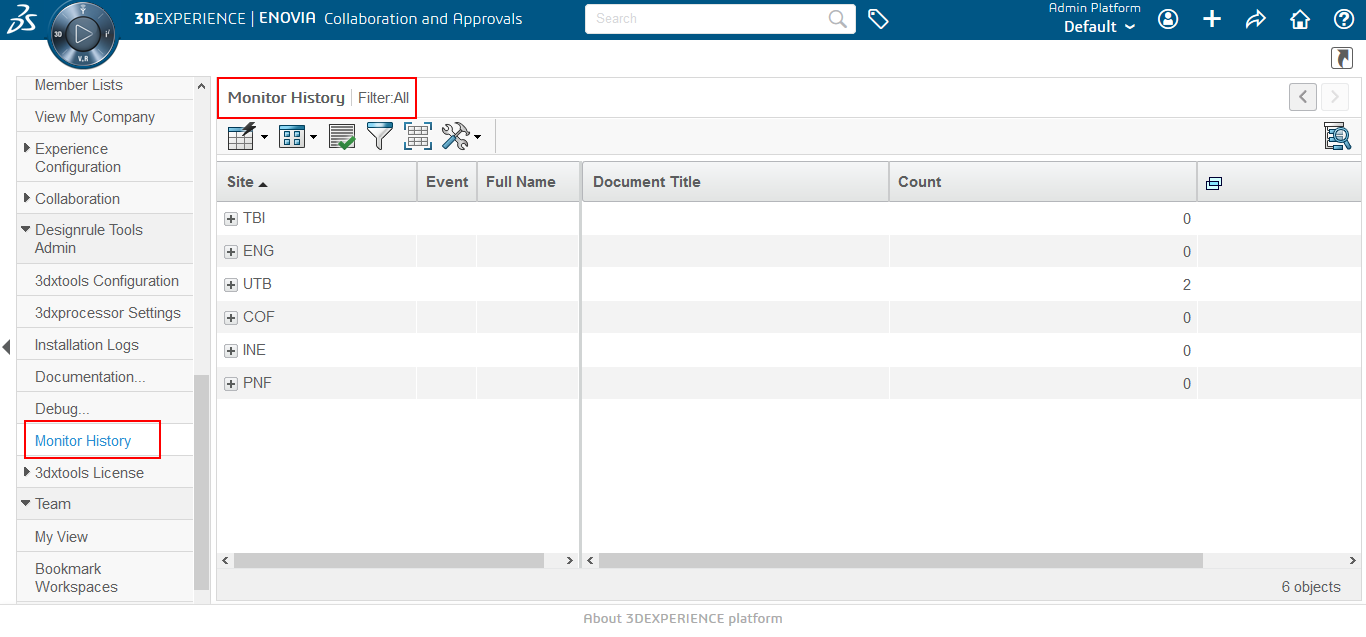
* Part Specification
  + gapGAPSpecification
  + gapCalculatedOutputSpecification
  + gapSoftwareSpecification
* CAD Drawing
  + gapAutoCAD
* CAD Model
* Drawing Print

A custom object will create for each checkout event with above mentioned details

1. Monitoring through the Web UI

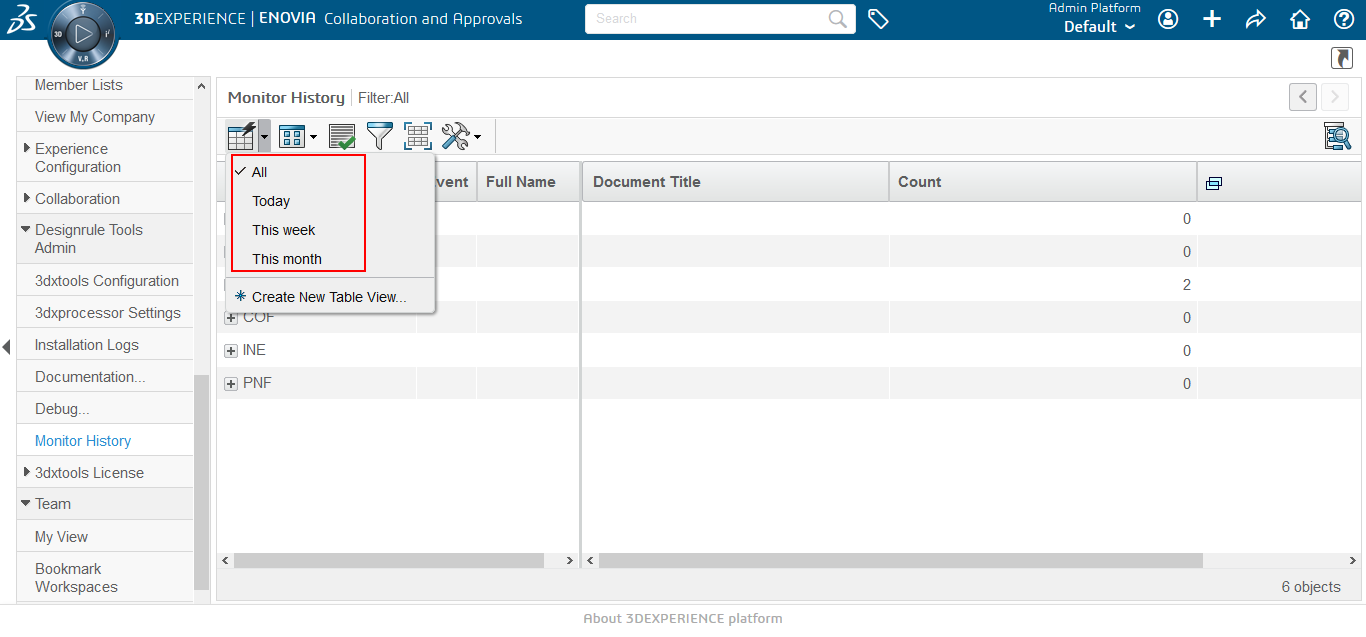
**Monitor History Page**

Will be visible to person who is having “DRLTools Admin” Role.



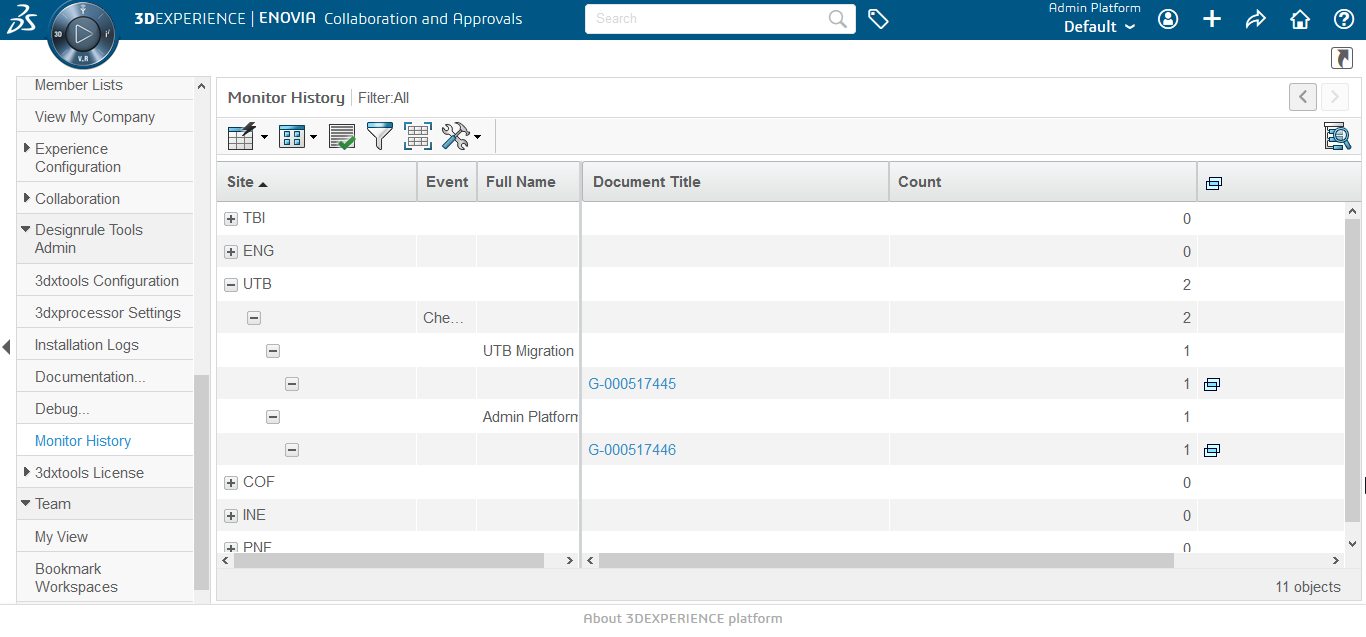
Have 4 different filters based on time, Today will be the default filter.

1. Today – Display History Objects which are created Today
2. This Week - Display History Objects which are created This Week
3. This Month - Display History Objects which are created This Month
4. All – Display All History Objects

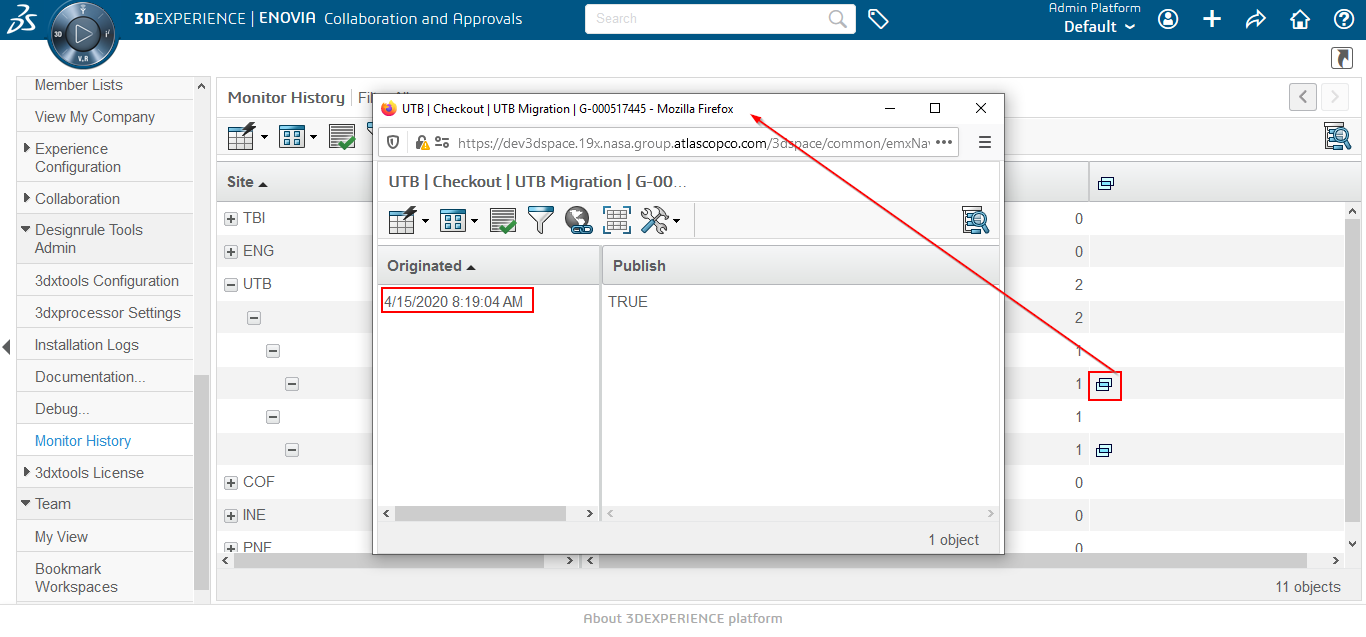


There are 3 Level Expansion of Each Site as shown below

Site 🡪 Event 🡪 Person 🡪 Object

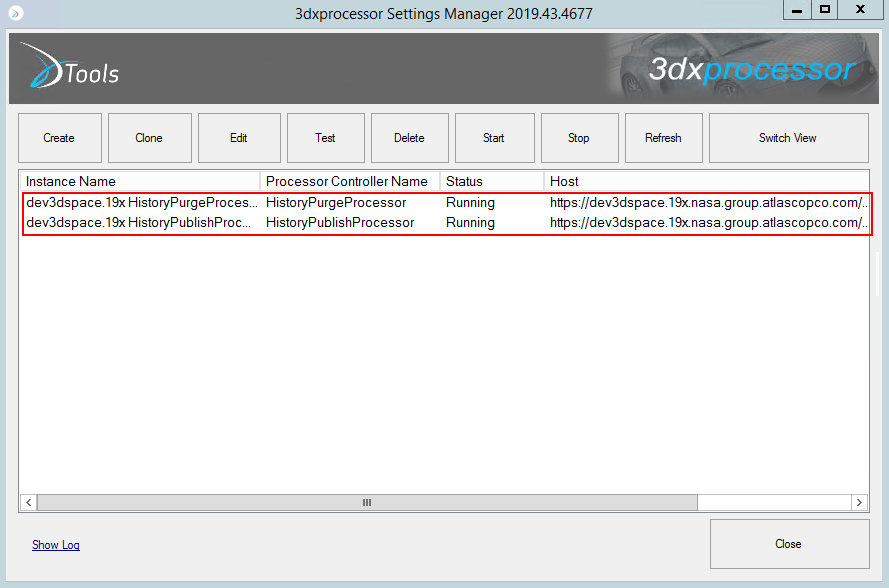


Each Object checkout time can be seen by clicking on popup window at the end of each object row.



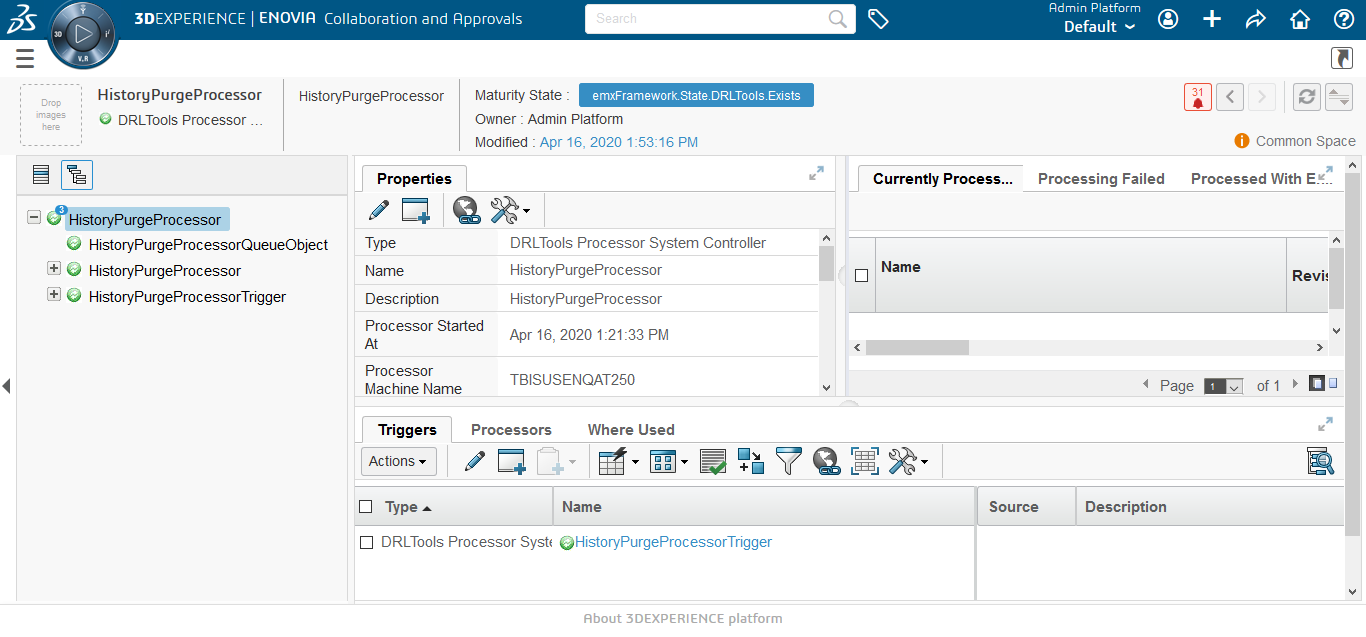
1. 3dxprocessor

With the 3dxprocessor we can monitor objects history, get daily history report and purge history objects.



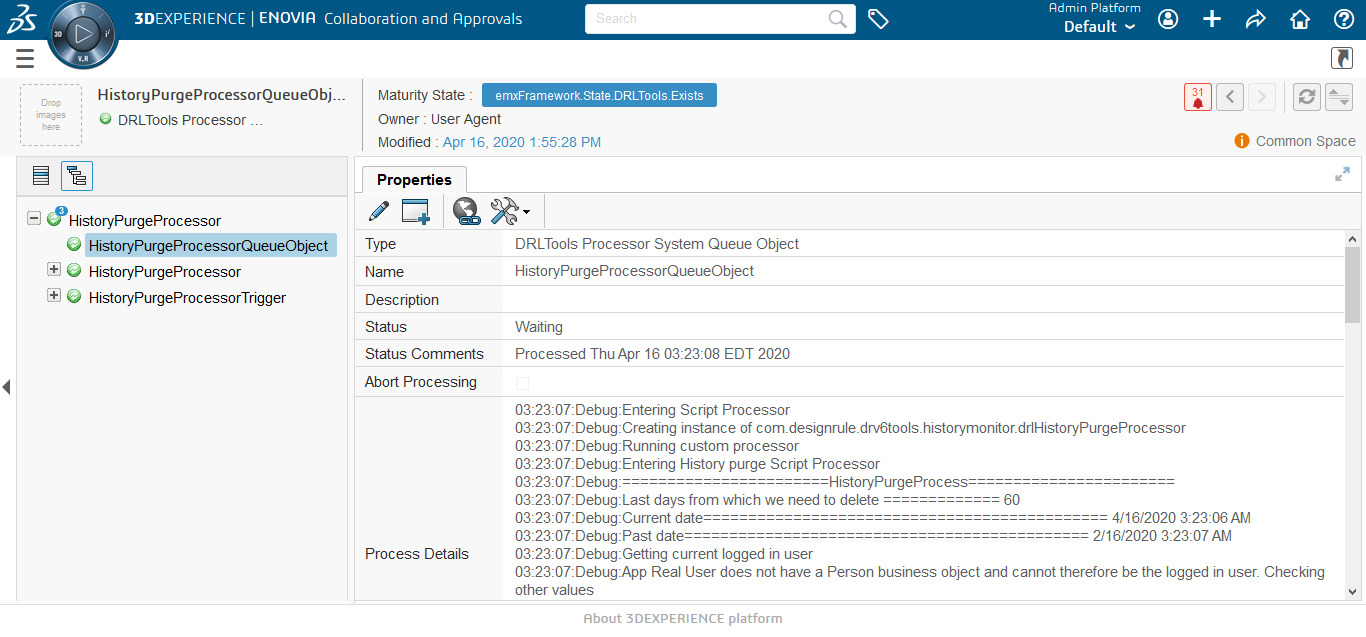
* 1. History Purge Processor

This will run every day early morning and purge history objects older than a certain date (i.e. objects older than 60 days).

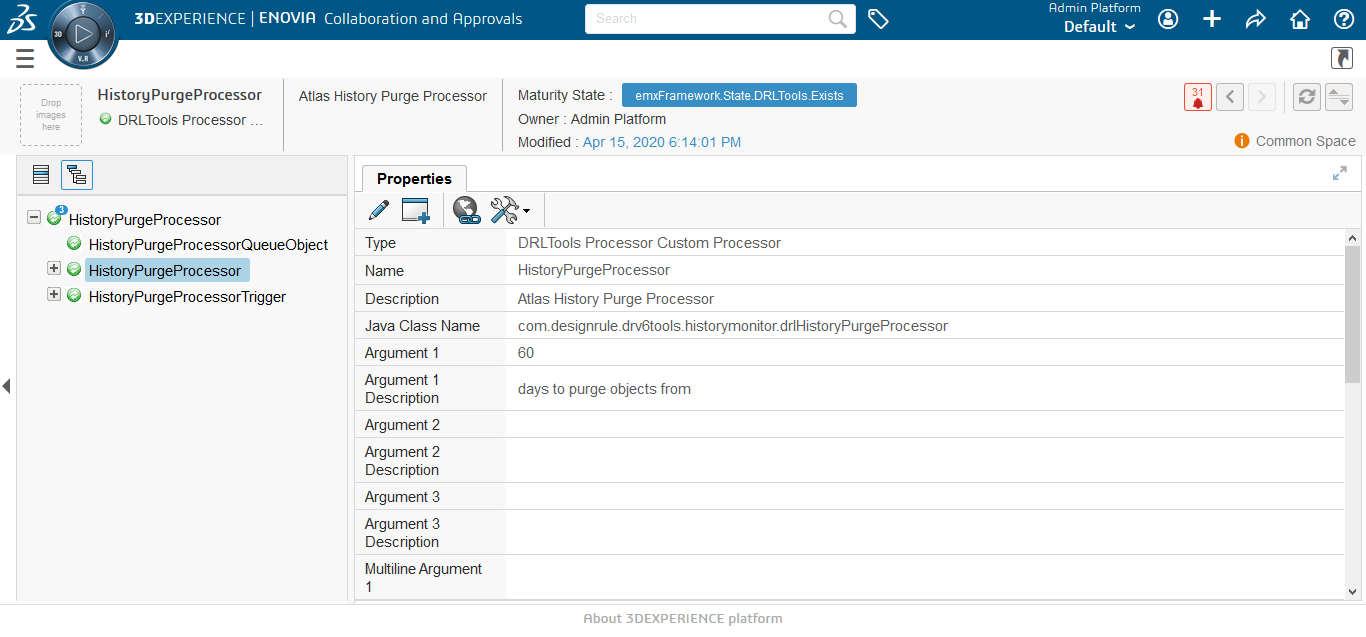


* + 1. HistoryPurgeProcessorQueueObject

Here queue object process to be monitored. Like Queue current status, Process logs.



* + 1. HistoryPurgeProcessor

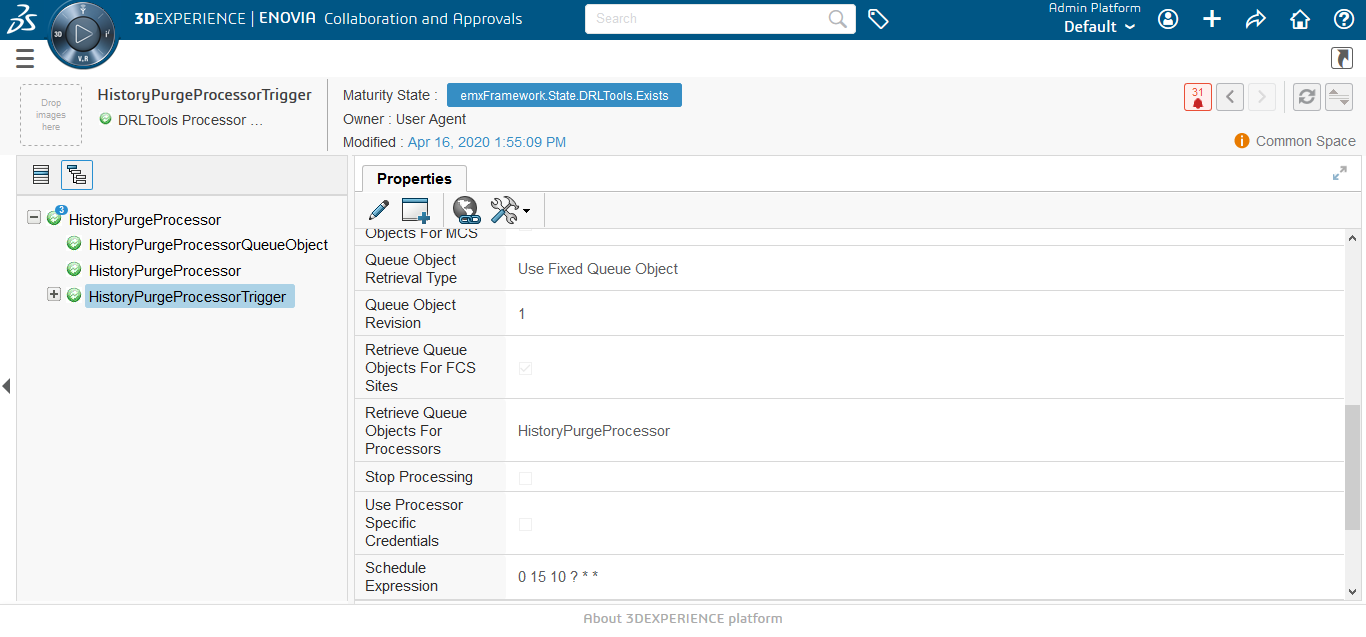


|  |  |
| --- | --- |
| Attributes | Value |
| Argument 1 | 60 |
| Argument 1 Description | days to purge objects from |

Argument 1 Specified how many days older objects need to be deleted. I.e. Objects created/originated before 60 days will be purged.

* + 1. HistoryPurgeProcessorTrigger

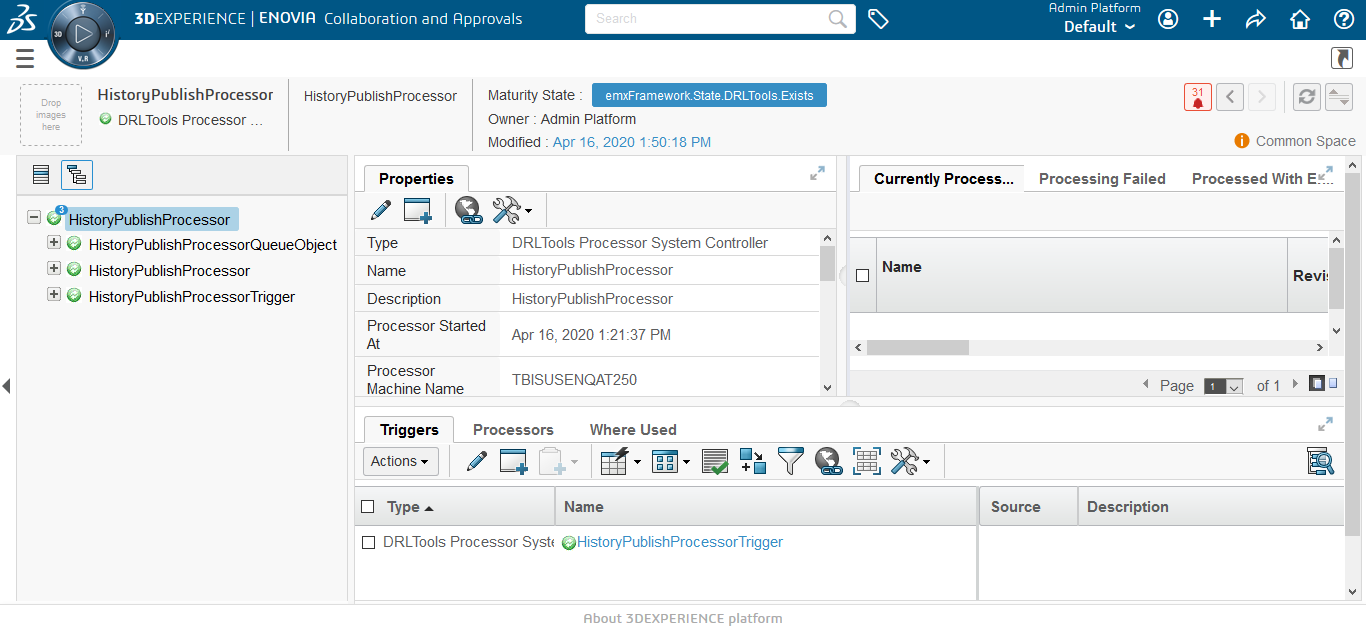
As of now processor will run everyday 10:15 AM u can change it by referring below table.



|  |  |
| --- | --- |
| Expression | Meaning |
| 0 \* \* \* \* ? | Run every minute |
| 0 0/5 \* \* \* ? | Run every 5 minutes |
| 0 0 12 \* \* ? | Run at 12pm (noon) every day |
| 0 15 10 ? \* \* | Run at 10:15am every day |
| 0 15 10 \* \* ? | Run at 10:15am every day |
| 0 15 10 \* \* ? \* | Run at 10:15am every day |
| 0 15 10 \* \* ? 2005 | Run at 10:15am every day during the year 2005 |
| 0 \* 14 \* \* ? | Run every minute starting at 2pm and ending at 2:59pm, every day |
| 0 0/5 14 \* \* ? | Run every 5 minutes starting at 2pm and ending at 2:55pm, every day |
| 0 0/5 14,18 \* \* ? | Run every 5 minutes starting at 2pm and ending at 2:55pm, AND Run every 5 minutes starting at 6pm and ending at 6:55pm, every day |
| 0 0-5 14 \* \* ? | Run every minute starting at 2pm and ending at 2:05pm, every day |
| 0 10,44 14 ? 3 WED | Run at 2:10pm and at 2:44pm every Wednesday in the month of March. |
| 0 15 10 ? \* MON-FRI | Run at 10:15am every Monday, Tuesday, Wednesday, Thursday and Friday |
| 0 15 10 15 \* ? | Run at 10:15am on the 15th day of every month |
| 0 15 10 L \* ? | Run at 10:15am on the last day of every month |
| 0 15 10 ? \* 6L | Run at 10:15am on the last Friday of every month |
| 0 15 10 ? \* 6L | Run at 10:15am on the last Friday of every month |
| 0 15 10 ? \* 6L 2002-2005 | Run at 10:15am on every last Friday of every month during the years 2002, 2003, 2004 and 2005 |
| 0 15 10 ? \* 6#3 | Run at 10:15am on the third Friday of every month |
| 0 0 12 1/5 \* ? | Run at 12pm (noon) every 5 days every month, starting on the first day of the month. |
| 0 11 11 11 11 ? | Run every November 11th at 11:11am. |

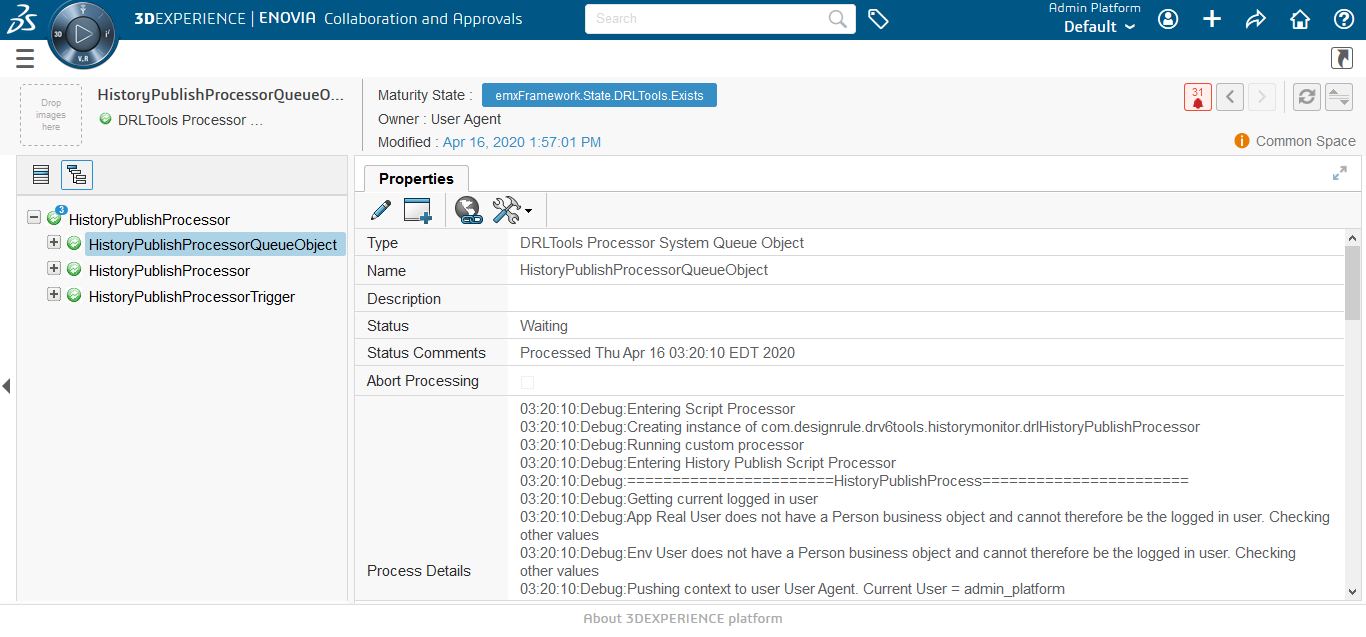
* 1. History Publish Processor

This will run every day early morning and publish history objects to DB which are not published yet.

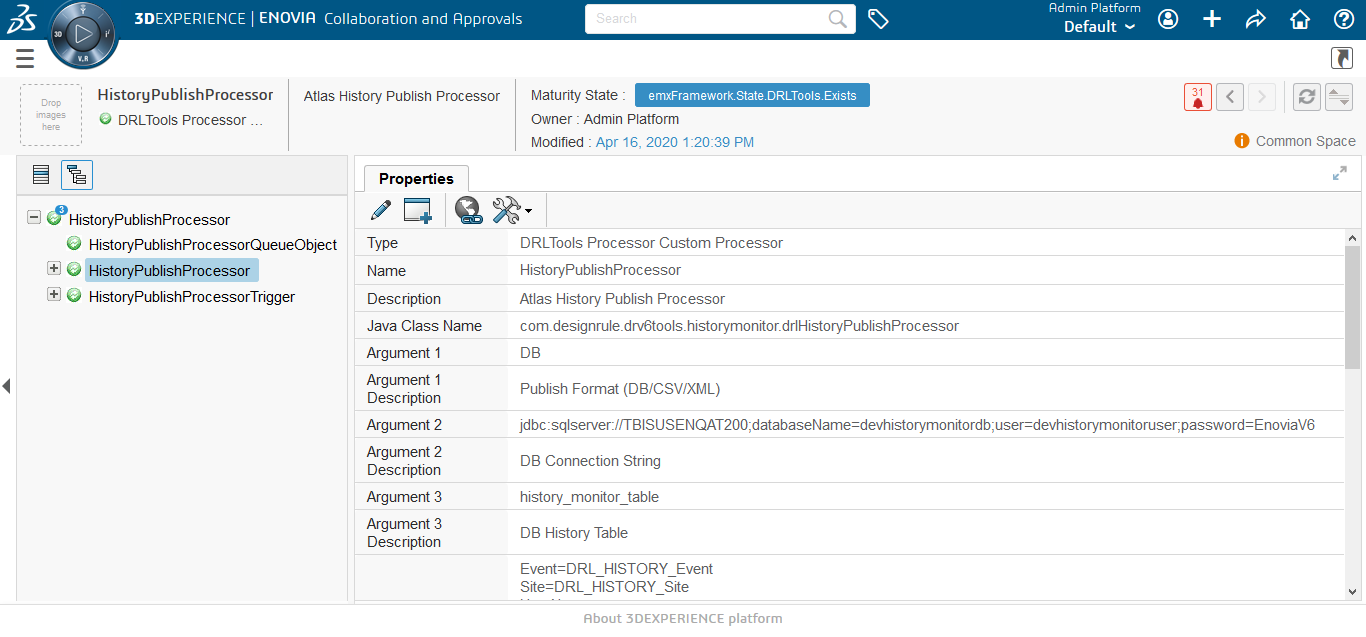


* + 1. HistoryPublishProcessorQueueObject

Here queue object process to be monitored. Like Queue current status, Process logs.



* + 1. HistoryPublishProcessor



Following configuration to be done. For more detail on macros, please refer section 6.2.1

|  |  |
| --- | --- |
| Attributes | Value |
| Argument 1 | DB |
| Argument 1 Description | Publish Format (DB/CSV/XML) |
| Argument 2 | jdbc:sqlserver://TBISUSENQAT200;databaseName=devhistorymonitordb; user=devhistorymonitoruser; password=EnoviaV6 |
| Argument 2 Description | DB Connection String |
| Argument 3 | history\_monitor\_table |
| Argument 3 Description | DB History Table |
| Multiline Argument 1 | Event=DRL\_HISTORY\_Event Site=DRL\_HISTORY\_Site UserName=name ObjectID=DRL\_HISTORY\_ObjectId ObjectType=DRL\_HISTORY\_Type ObjectDisplayName=DRL\_HISTORY\_ObjectDisplayName Timestamp=originated |
| Multiline Argument 1 Description | Attribute and DB Column Mapping |
| Multiline Argument 2 | C:\temp\Export |
| Multiline Argument 2 Description | CSV/XML Export Path |

Argument 1 Specifies Publish Format DB/CSV/XML I.e. Objects created will be Exported to one of the formats.

Argument 2 Specifies JDBC Connection String

**Format for Connection String**

jdbc:sqlserver://[SERVER\_NAME]\[INSTANCE\_NAME]:[PORT\_NUMBER];databaseName=[DB\_NAME]; user=[DB\_USER]; password=[USER\_PASSWORD]

No need to specify INSTANCE\_NAME if default instance is being used

No need to Specify PORT\_NUMBER as well if default is being used

Ex:

**Without Instance Default Port**

jdbc:sqlserver://TBISUSENQAT200;databaseName=devhistorymonitordb; user=devhistorymonitoruser; password=EnoviaV6

**With Instance Default Port**

jdbc:sqlserver://DEVTPMAN02\MSSQLSERVER19X;databaseName=historymonitordb;user=historymonitoruser;password=EnoviaV6

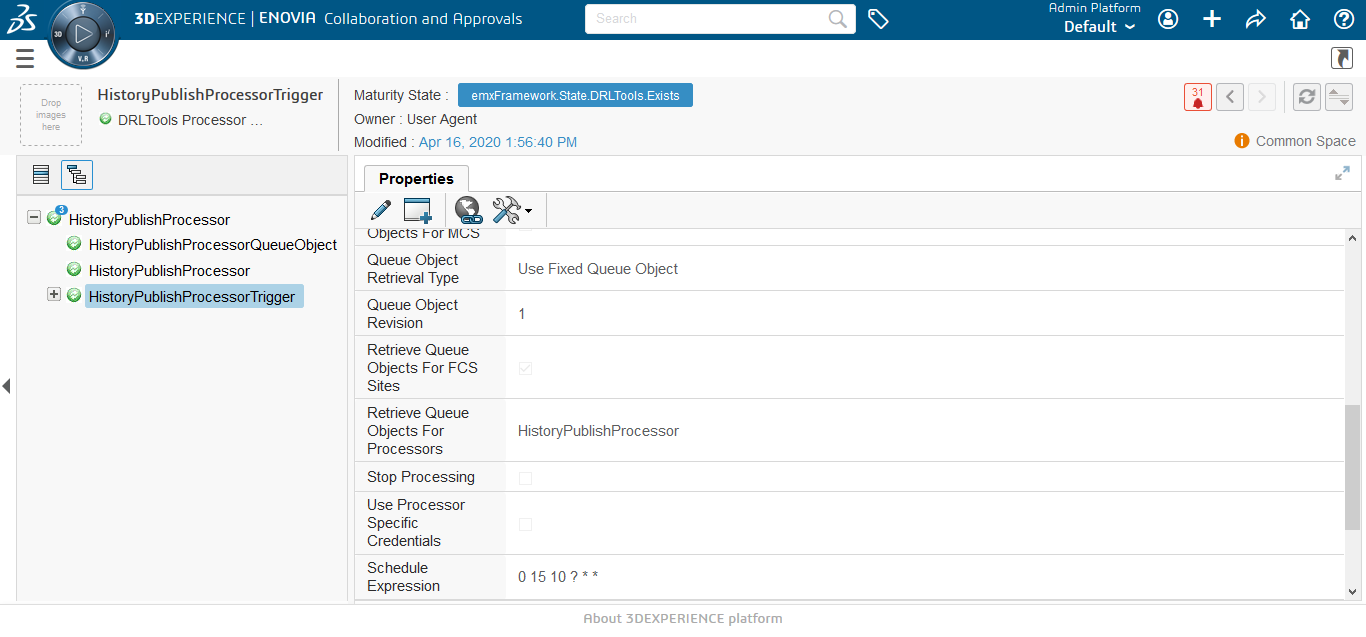
Argument 3 Specifies History Table Name

Multiline Argument 1 Specifies Attribute and DB Column Mapping

Multiline Argument 2 Export Path for Object (if Publish format id CSV or XML)

* + 1. HistoryPublishProcessorTrigger

As of now processor will run every day 10:15 AM u can change it by referring table from section 5.1.3.



And as of now published Data will be stored in above mentioned table history\_monitor\_table in DB devhistorymonitordb as shown below.

