CREATING TARGETED AUDITS IN MAXIMO

Learn how to build a custom audit solution in your Maximo system that will provide pinpoint accuracy for auditing of key system data

Presented by Stephen Hume, MaxTECH Chair

Agenda

1 **Background**

2 **MaxTECH Objectives**

3 **Create Table, Relationships & Modify the App**

4 **Exploring the Automation Script (Original)**

5 **Live demo**

6 **How to audit a related object**

7 **Questions/Ideas for Improvement**

Stephen Hume

Stephen has been working with IBM Maximo for over ten years in a variety of industries (Oil and Gas, Utilities). He has taught Maximo courses to end users for both Technical and Functional audiences and chairs the MaxTECH User Group.

Background – HSE Implementation

During the development of the Incident Management application for a Maximo HSE implementation, the business users indicated that they wanted specific fields of the incident record audited for any changes that were made during the lifecycle of the Incident Record. They also requested that some of the fields on linked tables could be audited.

The second part of this audit requirement was that they wanted this audit data to be visible in the incident application, and that the required information to be displayed was the following:
Database Table, Attribute, Old Value, New Value, Date Changed, Who Changed It.

It was determined that the out of the box audit solution could be used to meet this requirement, but it would not be easy to display the required audit details within the Incident Record. Maximo audit records just store the changed values and hunting down the corresponding audit record to find the "before" value of a changed field would be difficult.

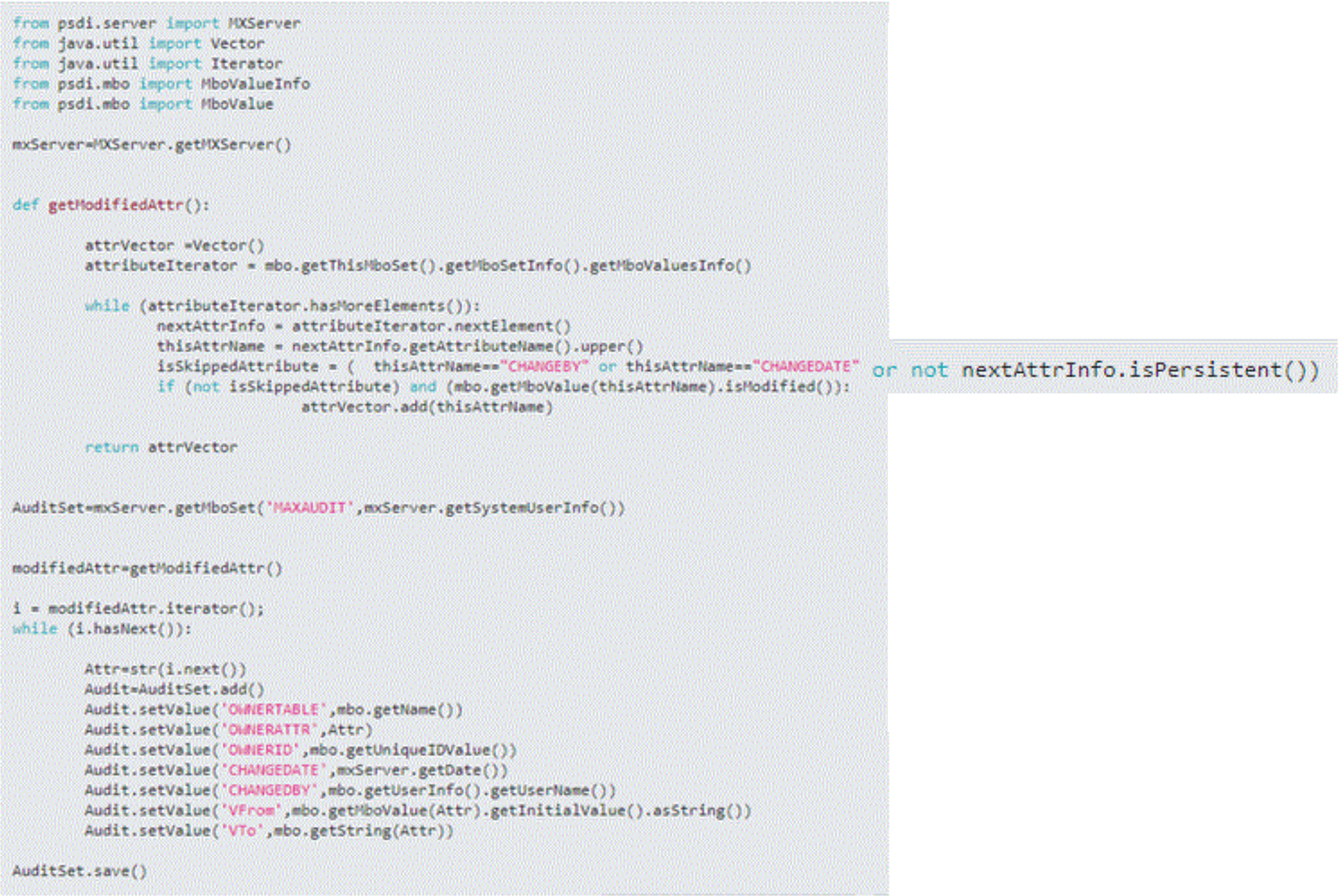
Conclusion: One of the key requirements in the HSE project was to display audit details from related tables. This could not be achieved with the Standard Maximo Audit Solution.

The Original Script

The script was written to produce reports for any attribute that has been changed, except for specific attributes which may be skipped (CHANGEBY, CHANGEDATE and non-persistent attributes).

The problem with this approach is that it is a shotgun approach creating audit records that the business users may not be interested in.

When a new incident was created, this script generated an audit record for every incident in the vie (thousands of records).



```
from psdi.server import MXServer
from java.util import Vector
from java.util import Iterator
from psdi.mbo import MboValueInfo
from psdi.mbo import MboValue

mxServer=MXServer.getMXServer()


def getModifiedAttr():

        attrVector =Vector()
        attributeIterator = mbo.getThisMboSet().getMboSetInfo().getMboValuesInfo()

        while (attributeIterator.hasMoreElements()):
                nextAttrInfo = attributeIterator.nextElement()
                thisAttrName = nextAttrInfo.getAttributeName().upper()
                isSkippedAttribute = (  thisAttrName=="CHANGEBY" or thisAttrName=="CHANGEDATE" or not nextAttrInfo.isPersistent())
                if (not isSkippedAttribute) and (mbo.getMboValue(thisAttrName).isModified()):
                                attrVector.add(thisAttrName)

        return attrVector


AuditSet=mxServer.getMboSet('MAXAUDIT',mxServer.getSystemUserInfo())


modifiedAttr=getModifiedAttr()

i = modifiedAttr.iterator();
while (i.hasNext()):

        Attr=str(i.next())
        Audit=AuditSet.add()
        Audit.setValue('OWNERTABLE',mbo.getName())
        Audit.setValue('OWNERATTR',Attr)
        Audit.setValue('OWNERID',mbo.getUniqueIDValue())
        Audit.setValue('CHANGEDATE',mxServer.getDate())
        Audit.setValue('CHANGEDBY',mbo.getUserInfo().getUserName())
        Audit.setValue('VFrom',mbo.getMboValue(Attr).getInitialValue().asString())
        Audit.setValue('VTo',mbo.getString(Attr))

AuditSet.save()
```

MaxTECH Objectives

The Objective of this presentation is to show you how to build a custom audit solution in your Maximo system to provide pinpoint accuracy for auditing of key system data.

Concepts and tools utilized:

- Database configuration (custom table)
- Relationships
- Application Design – be wary of adding new tab to an application
- Automation Script – Script Creation
- Automation Script – Launchpoint creation

Step by Step: Create Table

Create BPDAUDIT table (substitute your own company naming standards)

TIP

When you create a new table in Maximo, by default the system adds a description attribute. This will not be needed so should be deleted before the new table is saved.

Once you save the table, put the system into admin mode and apply your configuration changes.

Attribute	Description	Type	Length	Title	Same As Object	Same As Attribute	Search Type
CHANGEBY	User ID that made the change	UPPER	*	Change By	PERSON	PERSONID	WILDCARD
CHANGEDATE	Date the change was made	DATETIME	10	Change Date			EXACT
BPDAUDITID	Bpdauditid	BIGINT	19	Bpdauditid			EXACT
NEWVALUE	New value of the field after the change	ALN	2000	New Value			WILDCARD
OLDVALUE	Original value of the field before the change	ALN	2000	Old Value			WILDCARD
OWNERATTR	Attribute which was changed	UPPER	50	Attribute Changed	MAXATTRIBUTE	ATTRIBUTENAME	WILDCARD
OWNERID	Unique ID of the record that was changed	BIGINT	19	Record ID			EXACT
OWNERTABLE	Table of the record which was changed	UPPER	30	Table Changed	MAXOBJECT	OBJECTNAME	WILDCARD

Step by Step: Create Relationship

Create the following relationship on the table you are auditing:

Object: SR
Relationship: BPDAUDIT
Where Clause: ownertable = ‘SR’ and ownerid = :ticketuid

If you build an audit on tables which are linked to the SR Table then build the following relationship
Relationship: BPDAUDITALL
Where Clause:
(ownertable = ‘SR’ and ownerid = :ticketuid) or
(ownertable = “XXXXXX’

and ownerid in (select XXXXXXUID from maximo.XXXXXX where ticketid = :ticketid))

Remarks: Relationship to get the audit details for SR and child tables

‘XXXXXX’ = your child table

Create the following relationships on the BPDAUDIT Table:

Object: BPDAUDIT
Relationship: MAXATTRIBUTE
Where Clause: objectname = :ownertable and attributename = :ownerattr
Child Object: MAXATTRIBUTE
Remarks: Relationship to the maxattribute table to get the attribute details

Relationship: MAXOBJECT

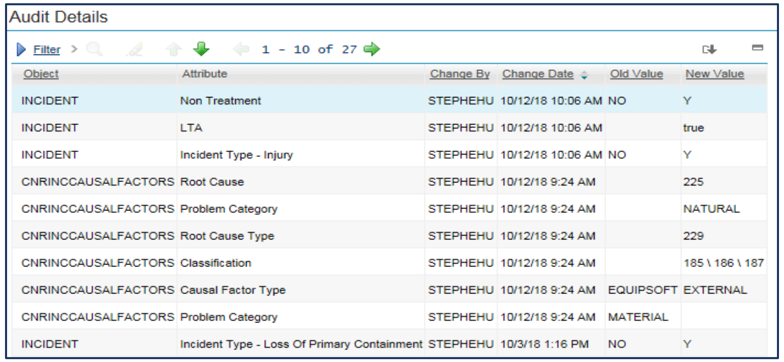
Where Clause: objectname = :ownertable
Child Table: MAXOBECT
Remarks: Relationship to the maxobject table to get the object details

Only use BPDAUDITALL if you are wanting to audit tables linked to your table (in this case SR)

Step by Step: Update SR Application

Now is the time to add a spot for the audit information in the application you will be auditing. In the case of the Incident Application, the users wanted the audit data visible on a new tab in the app. This is the direction that will be demonstrated today.

1. Open SR Application in Application Designer
2. Export the SR application, and save the XML to your desktop. **Create a backup copy of the XML also.**
3. Edit the SR XML using Notepad ++ to manually create a new TAB. Add the SR Header Section to the TAB and add a Section where the Audit Table will be displayed. Save your manual changes and import the revised XML into the application designer in Maximo.
4. Add the Audit table to the application using application designer. Select all the attributes to be displayed in that table, set the table to "read only" and save your changes.



Audit Details

Object	Attribute	Change By	Change Date	Old Value	New Value
INCIDENT	Non Treatment	STEPHEHU	10/12/18 10:06 AM	NO	Y
INCIDENT	LTA	STEPHEHU	10/12/18 10:06 AM		true
INCIDENT	Incident Type - Injury	STEPHEHU	10/12/18 10:06 AM	NO	Y
CNRINCCAUSALFACTORS	Root Cause	STEPHEHU	10/12/18 9:24 AM		225
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM		NATURAL
CNRINCCAUSALFACTORS	Root Cause Type	STEPHEHU	10/12/18 9:24 AM		229
CNRINCCAUSALFACTORS	Classification	STEPHEHU	10/12/18 9:24 AM		185 \ 186 \ 187
CNRINCCAUSALFACTORS	Causal Factor Type	STEPHEHU	10/12/18 9:24 AM	EQUIPSOFT	EXTERNAL
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM	MATERIAL	
INCIDENT	Incident Type - Loss Of Primary Containment	STEPHEHU	10/3/18 1:16 PM	NO	Y

Take care when adding a new tab to an application in Maximo.

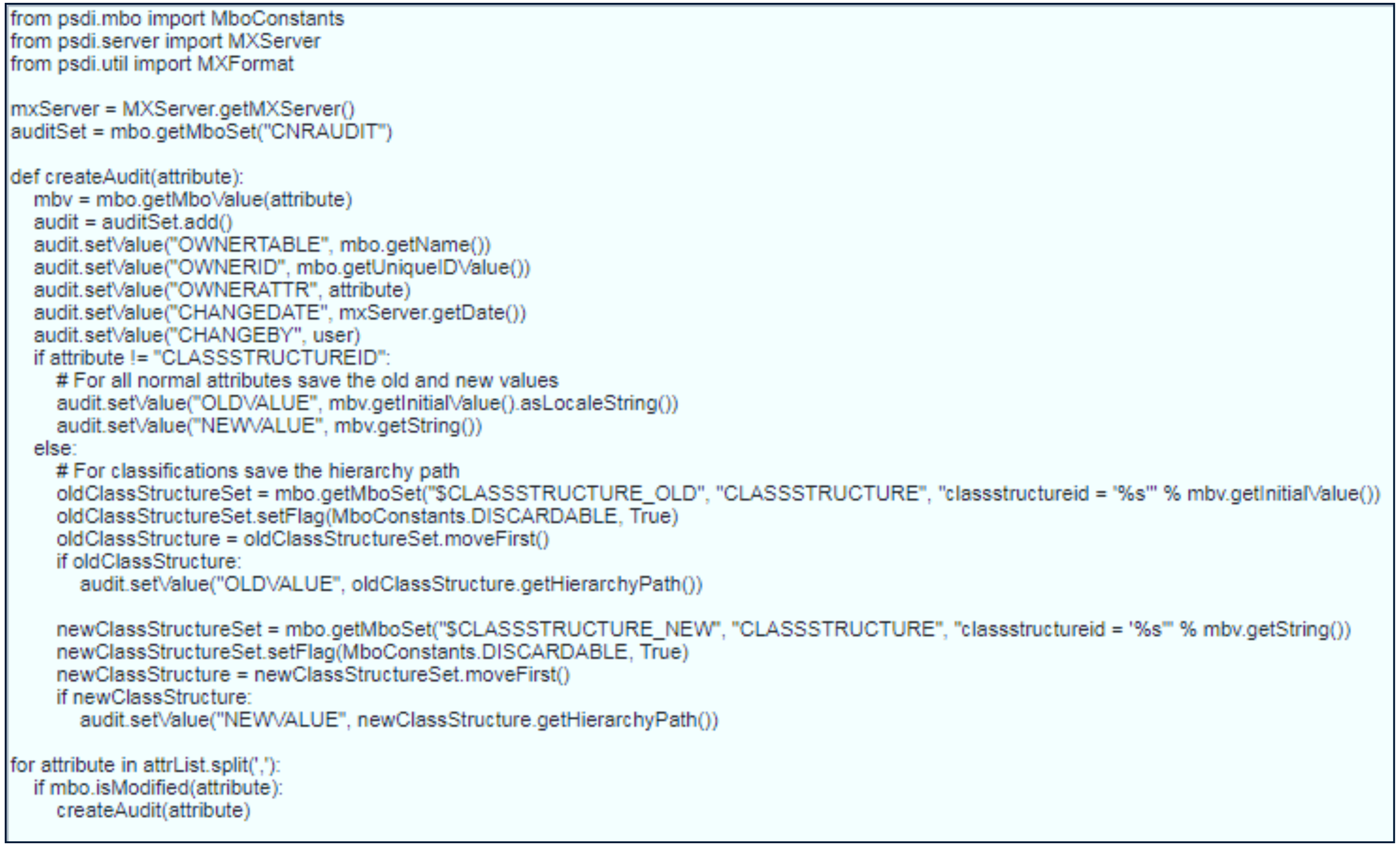
If you use application designer the application can become corrupt.

TIP **When you open the SR application the Audit table will initially be blank until the script is created to write audit records.**

Step by Step: Automation Script

This script achieves the following:

- It uses a variable to store the list of attributes to be audited (so that not all changes will result in creation of an audit record)
- It retrieves the hierarchy path if the attribute being audited is a classstructureid (more user friendly)
- It converts booleans (yes or no) to Yes and No instead of zero and 1
- It is completely reusable to audit any table in Maximo. Just need a new launch point.
- The audits will only begin when values are changed, not when they are first created.



```
from psdi.mbo import MboConstants
from psdi.server import MXServer
from psdi.util import MXFormat

mxServer = MXServer.getMXServer()
auditSet = mbo.getMboSet("CNRAUDIT")

def createAudit(attribute):
    mbv = mbo.getMboValue(attribute)
    audit = auditSet.add()
    audit.setValue("OWNERTABLE", mbo.getName())
    audit.setValue("OWNERID", mbo.getUniqueIDValue())
    audit.setValue("OWNERATTR", attribute)
    audit.setValue("CHANGEDATE", mxServer.getDate())
    audit.setValue("CHANGEBY", user)
    if attribute != "CLASSSTRUCTUREID":
        # For all normal attributes save the old and new values
        audit.setValue("OLDVALUE", mbv.getInitialValue().asLocaleString())
        audit.setValue("NEWVALUE", mbv.getString())
    else:
        # For classifications save the hierarchy path
        oldClassStructureSet = mbo.getMboSet("$CLASSSTRUCTURE_OLD", "CLASSSTRUCTURE", "classstructureid = '%s'" % mbv.getInitialValue())
        oldClassStructureSet.setFlag(MboConstants.DISCARDABLE, True)
        oldClassStructure = oldClassStructureSet.moveFirst()
        if oldClassStructure:
            audit.setValue("OLDVALUE", oldClassStructure.getHierarchyPath())

        newClassStructureSet = mbo.getMboSet("$CLASSSTRUCTURE_NEW", "CLASSSTRUCTURE", "classstructureid = '%s'" % mbv.getString())
        newClassStructureSet.setFlag(MboConstants.DISCARDABLE, True)
        newClassStructure = newClassStructureSet.moveFirst()
        if newClassStructure:
            audit.setValue("NEWVALUE", newClassStructure.getHierarchyPath())

for attribute in attrList.split(','):
    if mbo.isModified(attribute):
        createAudit(attribute)
```

Step by Step: Create Script

In Automation Scripts Create >> Script

Script: BPD Audit

Description: Object: Multiple, Event: Save, Description: Audit specific attributes

Script Language: Jython

Variable: attrList Comma-separated list of attributes to be audited

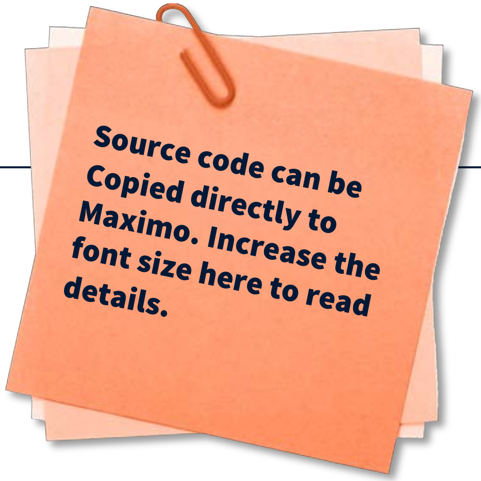
Variable Type: IN

Binding Type: LITERAL

Binding Value: UNDEFINED

Source Code:

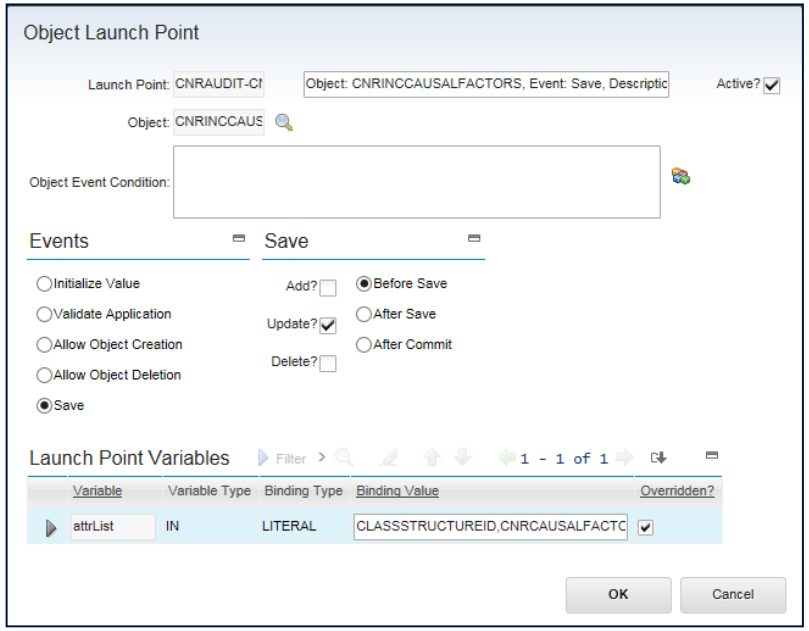
```
from psdi.mbo import MboConstants
from psdi.server import MXServer
from psdi.util import MXFormat

mxServer = MXServer.getMXServer()
auditSet = mbo.getMboSet(“BPDAUDIT")

def createAudit(attribute):
    mbv = mbo.getMboValue(attribute)
    audit = auditSet.add()
    audit.setValue("OWNERTABLE", mbo.getName())
    audit.setValue("OWNERID", mbo.getUniqueIDValue())
    audit.setValue("OWNERATTR", attribute)
    audit.setValue("CHANGEDATE", mxServer.getDate())
    audit.setValue("CHANGEBY", user)
    if attribute != "CLASSSTRUCTUREID":
        # For all normal attributes save the old and new values
        audit.setValue("OLDVALUE", mbv.getInitialValue().asLocaleString())
        audit.setValue("NEWVALUE", mbv.getString())
    else:
        # For classifications save the hierarchy path
        oldClassStructureSet = mbo.getMboSet("$CLASSSTRUCTURE_OLD", "CLASSSTRUCTURE", "classstructureid = '%s'" % mbv.getInitialValue())
        oldClassStructureSet.setFlag(MboConstants.DISCARDABLE, True)
        oldClassStructure = oldClassStructureSet.moveFirst()
        if oldClassStructure:
            audit.setValue("OLDVALUE", oldClassStructure.getHierarchyPath())

        newClassStructureSet = mbo.getMboSet("$CLASSSTRUCTURE_NEW", "CLASSSTRUCTURE", "classstructureid = '%s'" % mbv.getString())
        newClassStructureSet.setFlag(MboConstants.DISCARDABLE, True)
        newClassStructure = newClassStructureSet.moveFirst()
        if newClassStructure:
            audit.setValue("NEWVALUE", newClassStructure.getHierarchyPath())

for attribute in attrList.split(','):
    if mbo.isModified(attribute):
        createAudit(attribute)
```



Source code can be Copied directly to Maximo. Increase the font size here to read details.

Step by Step: Create Launchpoint

Object Launch Point: BPDAUDIT-SR Object: SR, Event: Save, Description: Audit specific attributes
Object: SR
Event: Save – Update
Launch Point Variable: attrList
Binding Value:
AFFECTEDDATE,AFFECTEDPERSON,LOCATION,ASSETNUM,PLUSGINCTYPE,REPORTEDPRIORITY,INTERNALPRIORITY,REPORTEDBY,URGENCY



Object Launch Point

Launch Point: CNRAUDIT-CI Object: CNRINCCAUSALFACTORS, Event: Save, Descriptic Active? ☑

Object: CNRINCCAUS

Object Event Condition:

Events

○ Initialize Value
○ Validate Application
○ Allow Object Creation
○ Allow Object Deletion
● Save

Save

Add? □
Update? ☑
Delete? □

● Before Save
○ After Save
○ After Commit

Launch Point Variables

Variable	Variable Type	Binding Type	Binding Value	Overridden?
attrList	IN	LITERAL	CLASSSTRUCTUREID,CNRCAUSALFACTC	☑

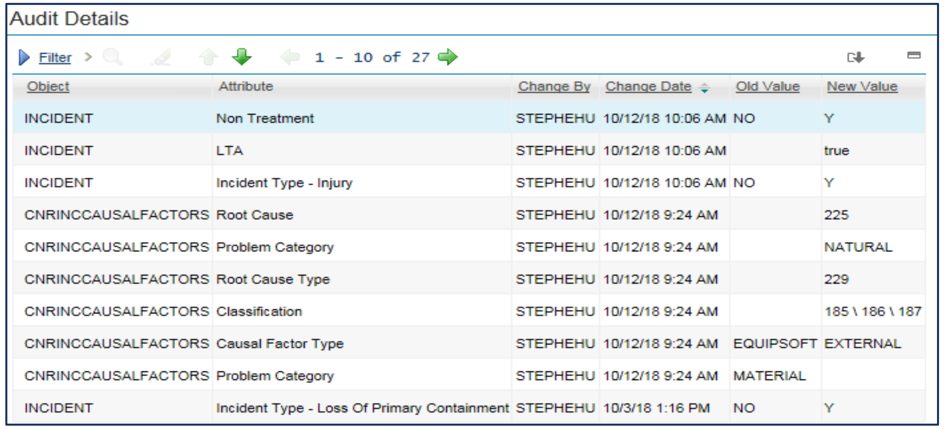
NOTE

Attribute maximo.launchpointvars.varbindingvalue needs to be increased in length to handle a longer list of attributes. We increased it to 1000 characters but it could bemade any length you require. This should be completed before doing the launchpoint.

The Binding Value is the list of attributes to be audited. It can include core or custom attributes.

Step by Step: Demonstration with SR

Go to the SR application in Maximo. Open any existing SR that is not CLOSED or CANCELLED.
Click on the New Audit Tab.
There should be no records displayed in the Audit Table.
Go to the SR – Service Request Tab.
Change the data in one or more of the audited attributes.
Click on the Audit Tab.
There should be one record visible for each of the audited attributes that were modified.
Go back to the Service Request tab, change one of the fields that is not audited.
Back to the Audit tab and there should not be any new records created.



Audit Details

Object	Attribute	Change By	Change Date	Old Value	New Value
INCIDENT	Non Treatment	STEPHEHU	10/12/18 10:06 AM	NO	Y
INCIDENT	LTA	STEPHEHU	10/12/18 10:06 AM		true
INCIDENT	Incident Type - Injury	STEPHEHU	10/12/18 10:06 AM	NO	Y
CNRINCCAUSALFACTORS	Root Cause	STEPHEHU	10/12/18 9:24 AM		225
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM		NATURAL
CNRINCCAUSALFACTORS	Root Cause Type	STEPHEHU	10/12/18 9:24 AM		229
CNRINCCAUSALFACTORS	Classification	STEPHEHU	10/12/18 9:24 AM		185 \ 186 \ 187
CNRINCCAUSALFACTORS	Causal Factor Type	STEPHEHU	10/12/18 9:24 AM	EQUIPSOFT	EXTERNAL
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM	MATERIAL	
INCIDENT	Incident Type - Loss Of Primary Containment	STEPHEHU	10/3/18 1:16 PM	NO	Y

The targeted audit function will only create audit records when one of the tracked attributes is changed.

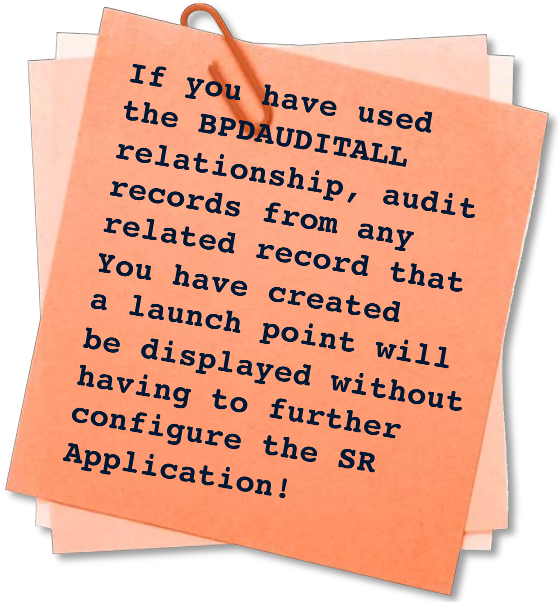
TIP

When you open the SR application the Audit table will initially be blank until the script is created to write audit records.

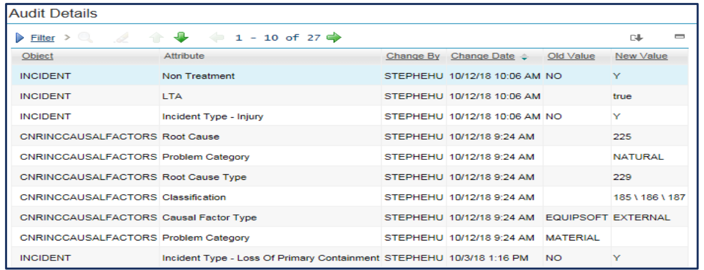
Step by Step: Audit of Related Table

The next use-case to explore is the need to audit attributes (fields) from a table other than the SR table that is related to the SR.
(In layman’s terms a sub-table).
In the incident application there were several related tables added as part of the implementation, each of which has specific audit requirements. In the out of the box SR application there are only a few related tables (Multiple Assets, Work Logs, Related Records). In this example will use the Multiple Assets Table.

1. Go to the Automation Scripts application.
2. Create Script with Object Launch Point (BPDAUDIT-MULTIASSTLOCCI)
3. Description: Object: MULTIASSETLOCCI , Event: Save, Description: Audit specific attributes
4. Launch Point Variables attrList Binding Value ASSETNUM,COMMENTS,LOCATION
5. Script – Existing, BPDAUDIT
6. Click next, until you see the create button – click Create


If you have used the BPDAUDITALL relationship, audit records from any related record that You have created a launch point will be displayed without having to further configure the SR Application!

This will create the new launch point – check to make sure that the Object Launch point has Events – Save and Update checked along with the Before Save selected.


Audit Details

Object	Attribute	Change By	Change Date	Old Value	New Value
INCIDENT	Non Treatment	STEPHEHU	10/12/18 10:06 AM	NO	Y
INCIDENT	LTA	STEPHEHU	10/12/18 10:06 AM		true
INCIDENT	Incident Type - Injury	STEPHEHU	10/12/18 10:06 AM	NO	Y
CNRINCCAUSALFACTORS	Root Cause	STEPHEHU	10/12/18 9:24 AM		225
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM		NATURAL
CNRINCCAUSALFACTORS	Root Cause Type	STEPHEHU	10/12/18 9:24 AM		229
CNRINCCAUSALFACTORS	Classification	STEPHEHU	10/12/18 9:24 AM		185 \ 186 \ 187
CNRINCCAUSALFACTORS	Causal Factor Type	STEPHEHU	10/12/18 9:24 AM	EQUIPSOFT	EXTERNAL
CNRINCCAUSALFACTORS	Problem Category	STEPHEHU	10/12/18 9:24 AM	MATERIAL	
INCIDENT	Incident Type - Loss Of Primary Containment	STEPHEHU	10/3/18 1:16 PM	NO	Y

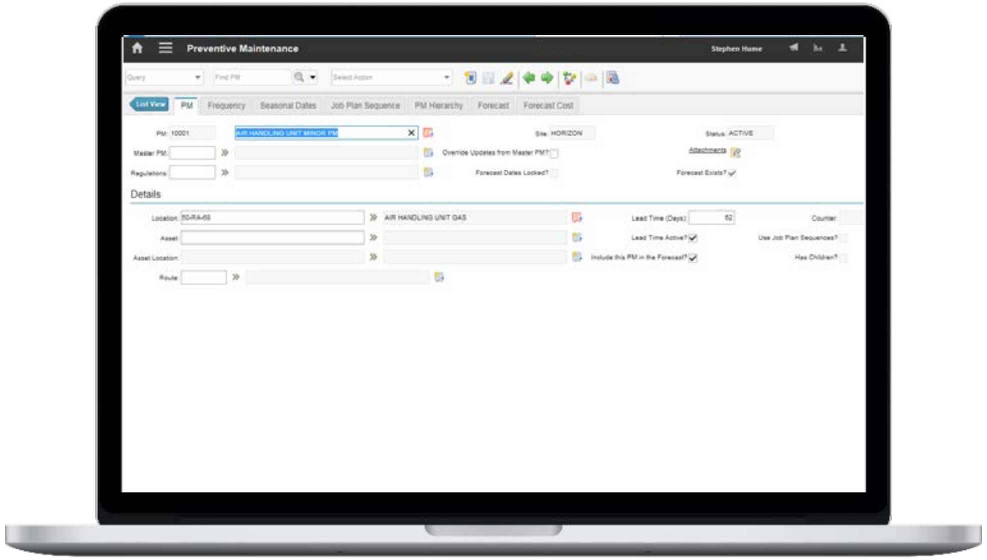
TIP

This will create the new launch point – check to make sure that the Object Launch point has Events – Save and Update checked along with the Before Save selected.

Success!

Other uses for this concept include:

- PM Status Audits
- Work Order Details
- Asset Information



PM: 10001 AIR HANDLING UNIT MINOR PM

Site: HORIZON

Status: ACTIVE

Master PM:

Override Updates from Master PM? □

Attachments

Regulations:

Forecast Dates Locked? □

Forecast Exists? ✓

Details

Location: 50-RA-68 AIR HANDLING UNIT GAS

Lead Time (Days): 52

Counter:

Asset:

Lead Time Active? ✓

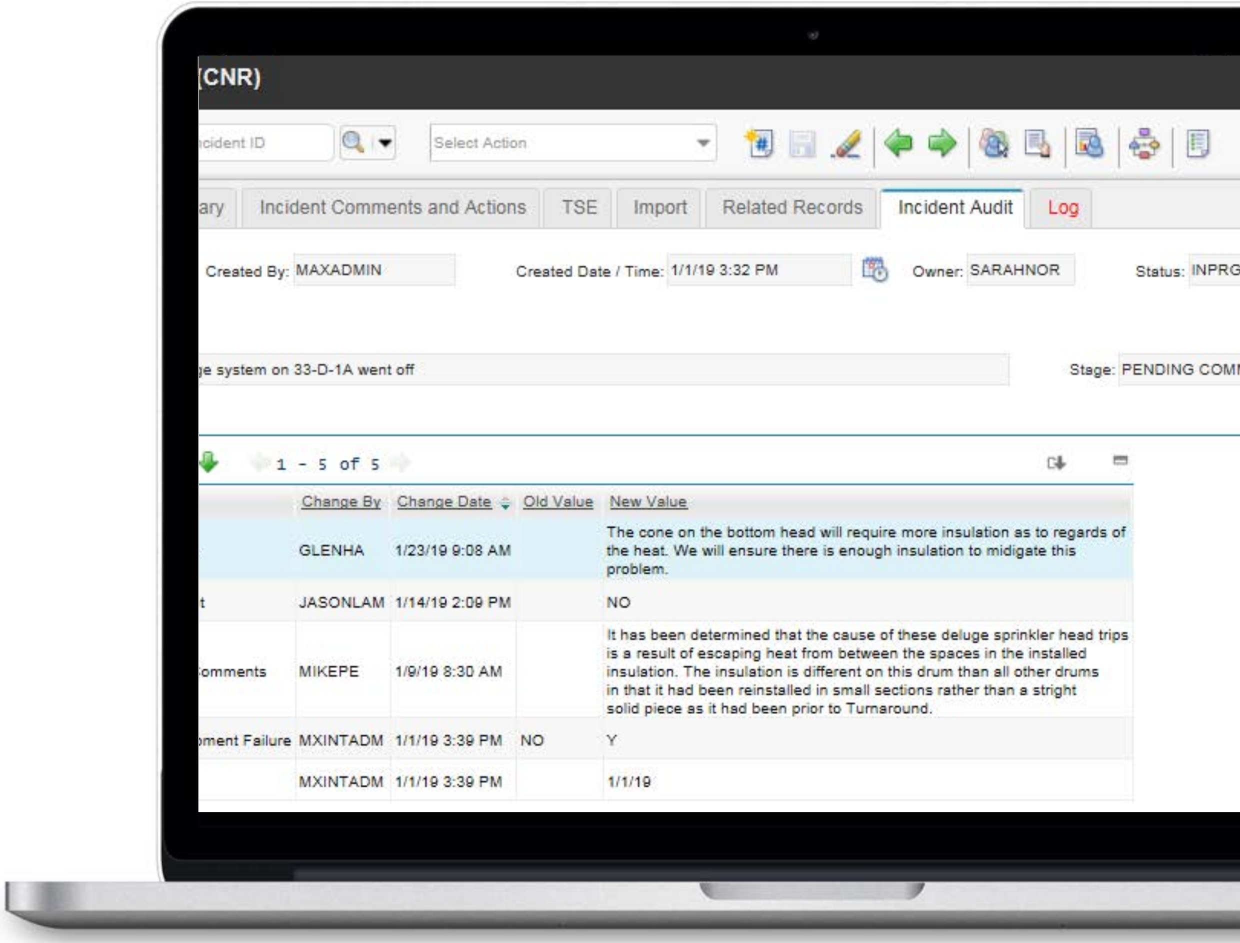
Use Job Plan Sequences? □

Asset Location:

Include this PM in the Forecast? ✓

Has Children? □

Route:



Created By: MAXADMIN

Created Date / Time: 1/1/19 3:32 PM

Owner: SARAHNOR

Status: INPRG

ge system on 33-D-1A went off

Stage: PENDING COM

1 - 5 of 5

	Change By	Change Date	Old Value	New Value
	GLENHA	1/23/19 9:08 AM		The cone on the bottom head will require more insulation as to regards of the heat. We will ensure there is enough insulation to midigate this problem.
t	JASONLAM	1/14/19 2:09 PM		NO
omments	MIKEPE	1/9/19 8:30 AM		It has been determined that the cause of these deluge sprinkler head trips is a result of escaping heat from between the spaces in the installed insulation. The insulation is different on this drum than all other drums in that it had been reinstalled in small sections rather than a stright solid piece as it had been prior to Turnaround.
pment Failure	MXINTADM	1/1/19 3:39 PM	NO	Y
	MXINTADM	1/1/19 3:39 PM		1/1/19

THANK YOU!

Success! You should have learned how to create a targeted audit in Maximo. We would like to credit:

Amir Samir

Amir Samir submitted the concept for a custom audit solution on Linked In.

Amir is a member of the MaxTECH User Group and graciously provided the idea upon which the demonstrated solution was developed.

Steven Cliff

Steven Cliff is also a member of MaxTECH and provided the programming for the automation script which was presented here.

Steven works for BPD Zenith as a Senior Maximo Technical Specialist.

Ask us a technical question:

maxtech@bpdzenith.com

www.bpdzenith.com/maxtech

MaxTECH is the first ever dedicated Maximo Technical User Group aimed at Maximo Administrators, Developers and Technical Support staff.

It is a great place for users to ask and answer technical questions, learn from each other, collaborate and help improve Maximo in your organization.

MaxTECH was founded in 2017 by BPD Zenith and is chaired by Maximo Consultant Stephen Hume. We host several events every year (Calgary, St. Louis, MaximoWorld, Houston, MUWG, Northern California MUG, Maximo UK & Ireland User Group) including digital events.

MaxTEACH is a free online user group designed to go in depth into a Maximo topic.

MAXTECH