

### PROJECTIZING ISOGEN

Below are the steps to have setup AutoPLANT so that each PROJECT will read its own OPCFHDR.EBS script.

#### INSTALLATION AND SETUP CHANGES FOR 8.5 OR 8.6

1. Edit the [PCF] section in the Program Files\Bentley\Plant 2004\XCHG\SYSPREF.INI to point to the project root for the Begin\_out and the Header\_out keys.

```
[PCF]
BEGIN_OUT = %PROJECT_ROOT%\CONFIG\XCHG\PCF\opcfhdr.ebs; at_PCFOut_begin
HEADER_OUT = %PROJECT_ROOT%\CONFIG\XCHG\PCF\opcfhdr.ebs; AT_PCF_HEADER_OUT
HEADER_IN = %Network_Root%\XCHG\PCF\ipcfhdr.ebs; AT_PCF_HEADER_IN
FOOTER_OUT = %Network_Root%\XCHG\PCF\opcfftr.ebs; at_PCFOut_footer
POST_OUT = %Network_Root%\XCHG\PCF\opcfftr.ebs; at_PCFOut_post
END_OUT = %Network_Root%\XCHG\PCF\opcfftr.ebs; at_PCFOut_end
MAP_FILE_OUT = %Network_Root%\XCHG\PCF\pcf.ini
FILE_EXT = PCF
COMP_TRAVERSAL_MODE=1
```

2. Create a folder in the project directory under CONFIG\XCHG\. Copy your project OPCFHDR.EBS into this project directory.

NOTE: If you do this, ALL projects will look into the project's CONFIG\XCHG\ directory for the OPCFHDR.EBS. It will NOT default back to the Program Files\Bentley\Plant 2004\XCHG\OPCFHDR.EBS if there is no file here.

#### EXPECTED SETUP FOR 8.9 (ALTHOUGH IT MAY WORK WITH THE 8.6SR)

8.9 is expected to allow first the project then the out-of-box software to be searched. (Tracked as enhancement item# 10367)

1. Modify the project config.ini to point to where the xchg/syspref.ini is located for this project.

```
[Xchg]
SYSPREF_FileName = %PROJECT_ROOT%\Xchg\syspref.ini
```

2. Allow syspref.ini to optionally provide a search path e.g.

```
[PCF]
BEGIN_OUT =
%PROJECT_ROOT%\xchg\pcf\opcfhdr.ebs;%Network_Root%\XCHG\PCF\opcfhdr.ebs;
at_PCFOut_begin
```

In the above scenario the file opcfhdr.ebs will be located in the

%PROJECT\_ROOT%\xchg\pcf

directory, if not found the file will then be searched for in

%NETWORK\_ROOT%\xchg\pcf

### **CODING OPCFHDR TO READ FROM THE PROJECT DB PROCESS TABLE**

The following script changes to the OPCFHDR will output values from the process table where the line number of the 3D item is the same as the one for the PROCESS record. It also outputs 2 items from the PROJECT table.

By “output” I mean it writes the values as ATTIBUTES in the header section of the PCF file.

If you implement this, YOU MAY NOT CALL SUPPORT for help. This is strictly a services customization.

1. Add these lines above the “ATTRIBUTE99 END” call in the OPCFHDR.EBS

```
Dim compID As At_ComponentID
Dim LINENUM As String

nStat = at_Component_getCurrID(compID)
nStat = at_Component_getString (LineNum, compID, "LineNumber")
Call GetProcessInfo (LineNum)           '// Added for US 7NOV03
nStat = at_XchgWritelnText(" ATTRIBUTE99 END")
```

2. Add these lines to the end of the script.

```
//
// All script below this point
// 7NOV03 - U. Sadiq
```

```
//  
  
Sub GetProcessInfo(lineno As String)  
  
Dim Lkey As String      ' line number keytag  
Dim bstatus As Boolean  
Dim rs As Object  
Dim i As Integer  
Dim status As Integer  
Dim querystring As String  
Dim linename As String  
Dim projectName As String  
Dim ProjectID As String  
Dim rs1 As Object  
Dim value As String  
Dim rs2 As Object  
  
Call startproj()  
  
' Get the project  
status = at_Project_getName (projectName)  
projectID = ProjectIDfromProjectName (projectName)  
    'do a silent load  
        bstatus = projobj.Init(projectid,0&)  
  
querystring = "Select * from Tag_reg where TAG_NO = " & lineno & ""  
status = ProjObj.ExecuteSql(querystring, rs, "PROJDATA")  
    rs.movefirst  
  
    If (rs.recordcount > 0 ) Then  
        Lkey = Trim$(Rs.Fields("KEYTAG"))  
  
        querystring = "Select * from PROCESS where KEYTAG = " & Lkey & ""  
        status = ProjObj.ExecuteSql(querystring, rs1, "PROJDATA")  
        rs1.movefirst  
  
        ' fields output also to the PCF, if they are not blank  
        For i = 1 To (rs1.fields.count - 1)  
            If Len(Rs1.Fields(i).value) > 0 Then  
                nStat = at_XchgWriteInText("  ATTRIBUTE" & i & " " & Rs1.Fields(i).value)  
            End If  
        Next  
  
        rs1.close  
    End If  
rs.close
```

```
'PROJECT INFO from the PROJECT.INI/PROJECT table
QUERYSTRING = "SELECT DESCRIPT FROM PROJECT WHERE ITEM = 'JOBNO'"
status = ProjObj.ExecuteSql(querystring, rs, "PROJDATA")
nStat = at_XchgWriteInText(" ATTRIBUTE90 " & Rs.Fields(0).value)
rs.close

QUERYSTRING = "SELECT DESCRIPT FROM PROJECT WHERE ITEM = 'SUBJOBNO'"
status = ProjObj.ExecuteSql(querystring, rs, "PROJDATA")
nStat = at_XchgWriteInText(" ATTRIBUTE91 " & Rs.Fields(0).value)
rs.close
```

**End Sub**

```
Sub StartProj()      '//
Dim bStatus As Boolean

If (Rapp Is Nothing) Then
    Set Rapp = CreateObject("APApplication1.RebisApp")
End If

If (Rapp Is Nothing) Then
    MsgBox "Failed to create Rapp. Ending"
End
Else
    If (Rapp.db Is Nothing) Then
        MsgBox "The Rapp.DB Is Nothing - how can this be? Ending"
        End
    End If

    If (ProjObj Is Nothing) Then
        Set ProjObj = CreateObject("APProjectDB1.ProjectDB")
        End If

    If (ProjObj Is Nothing) Then
        msgbox "ProjObj is nothing - problem with ProjDB2? Ending"
        End
    End If

    If (RebUser Is Nothing) Then
        Set RebUser = CreateObject("APUser1.RebisUser")
        End If

    If (RebUser Is Nothing) Then
        MsgBox "Could not set the Rebis User. Ending"
        End
    End If

```

```
                End If

                bstatus = RebUser.Login
                If (bStatus = False) Then
                    msgbox "Login is false. Ending"
                End
            End If
        End If
    End Sub

Private Function ProjectIDfromProjectName(projname2 As String) As String // Added for CPIA
    Dim Rs2 As Object
    Dim Rapp2 As Object

    If (Rapp2 Is Nothing) Then

        Set Rapp2 = CreateObject("APApplication1.RebisApp")
    End If

    If (Rapp2 Is Nothing) Then
        ProjectIDfromProjectName = vbNullString
        MsgBox "Could not initialize RebisApplication.DLL", vbCritical, "Error"
        Set Rapp2 = Nothing
        Exit Function
    End If

    Set Rs2 = Rapp2.GetRecordSet("Select PROJ_ID from At_proj where PROJ_NAME = " +
projname2 + """)
    If (Err.Number > 0) Or (Rs2.RecordCount = 0) Then
        MsgBox "There are no records that meet the SQL Criteria", vbInformation, "No Records"
        ProjectIDfromProjectName = vbNullString
    Else
        ProjectIDfromProjectName = Rs2.fields("PROJ_ID").Value
    End If
    Rs2.Close
    Set Rs2 = Nothing
    Set Rapp2 = Nothing
End Function
```