

PSST0101

January 5, 2011

Explorations in Component Interface: PeopleCode

Filed under: [App Engine](#), [Component Interface](#), [PeopleCode](#) — digitaleagle @ 8:36 am

This is a continuation of the following posts:

- [Part 1: Creating the Component Interface](#)
- [Part 2: Security](#)

Now, we need to write some code to use the Component Interface. Let's use the Application Engine program from [this step by step post](#).

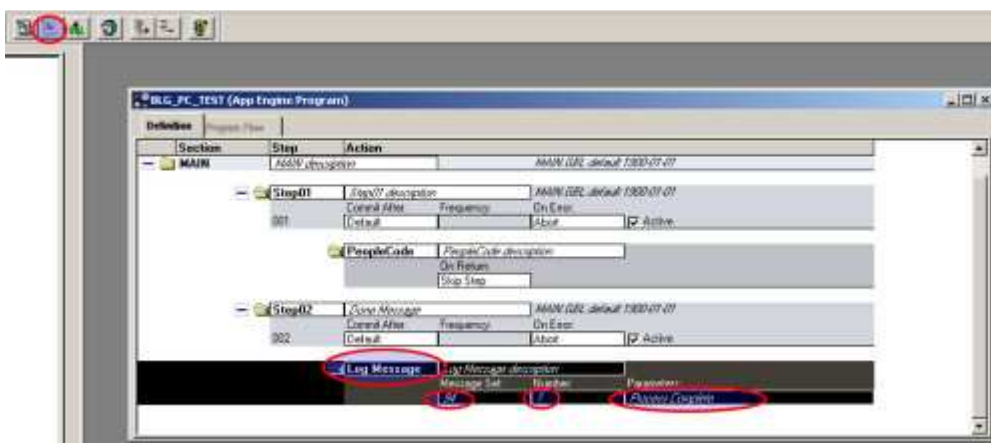
Adding a Log Step to the Program

Before we dive into the PeopleCode, we need to add a second step to the program. This step will come in handy for testing how the program works. It will just simply log a message to the standard out file.

I used this SQL to find a message catalog that would work easily:

```
SELECT * FROM PSMMSGCATDEFN  
WHERE MESSAGE_TEXT = '%1'
```

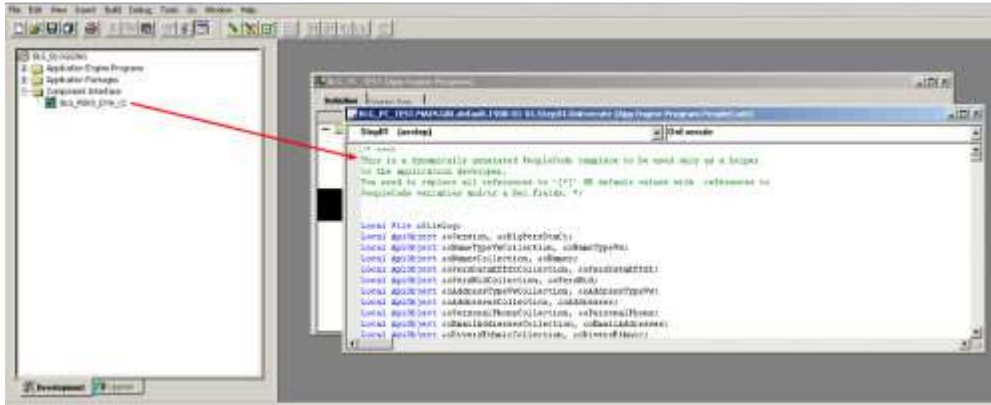
Then, I added a new step. First, you select Step 01. Then, you can either use the menu Insert > Step/Action or the toolbar icon. When you get the new step, change the action from SQL to Log Message. Then, set the Message Set to 94 and the Number to 1. For the parameters, enter something like "Process Complete".



Generating a PeopleCode CI Template

Now, we are ready to begin adding PeopleCode to Step 01. You can open the PeopleCode by double clicking anywhere in the gray of the PeopleCode action. If you have been following other tutorials, you probably have something in that program. For simplicity's sake, let's just delete that and start from scratch. This tutorial will assume you have a blank program at this point.

One of the cool things about Component Interfaces is that you can have Application Designer generate a template PeopleCode program. If you have the component interface in the project, you can just drag it from the development tab of the project list and drop it into the program. This generates a template in the program. You will want to have a blank program here because it will generate a lot of code. Note that this is a template in that you have to make some changes before you can even save it.



For this purpose, you can edit this template if you want, but it would probably be easier to delete the template back out and just type the code that I have written.

Writing the Program

First, we'll start with some variables. The file variable is to write a special log file with output information. The session variable stores your session and the oBlgPersDtaCi stores the reference to the component interface. The Emplid variable is going to be used to store the ID we want to update.

```
Local File &fileLog;
Local ApiObject &oSession, &oBlgPersDtaCi;
Local string &emplid;
```

Then, we need this function to display any error messages. It basically just loops through the session's message collection and prints out any messages that were triggered since the last call.

```
Function errorHandler()
    Local ApiObject &oPSMessageCollection, &oPSMessage;
    Local number &i;
    Local string &sErrMsgSetNum, &sErrMsgNum, &sErrMsgText, &sErrType;
    &oPSMessageCollection = &oSession.PSMessages;
    For &i = 1 To &oPSMessageCollection.Count
        &oPSMessage = &oPSMessageCollection.Item(&i);
        &sErrMsgSetNum = &oPSMessage.MessageSetNumber;
        &sErrMsgNum = &oPSMessage.MessageNumber;
        &sErrMsgText = &oPSMessage.Text;
        &fileLog.WriteLine(&sErrType | " (" | &sErrMsgSetNum | "," | &sErrMsgNum |
    End-For;
    rem ***** Delete the Messages from the collection *****;
    &oPSMessageCollection.DeleteAll();
End-Function;
```

Then, we'll start a function so we can update multiple times. The employee parameter is the ID we want to update, and we'll use the makeError parameter later. We put everything in a Try to attempt to handle errors gracefully.

```
Function updateCI(&emplid As string, &makeError As boolean)
    try
```

Then, we need to tell the tools what to do with the messages. We will tell it to use the collection only in an effort to handle the messages gracefully:

```
rem ***** Set the PeopleSoft Session Error Message Mode *****;
rem ***** 0 - None *****;
rem ***** 1 - PSMessages Collection only (default) *****;
rem ***** 2 - Message Box only *****;
rem ***** 3 - Both collection and message box *****;
&oSession.PSMessagesMode = 1;
```

Then, we need to initialize the component interface. This is simulating the user clicking the link on the menu to go into the component.

```
rem ***** Get the Component Interface *****;
&oBlgPersDtaCi = &oSession.GetCompIntfc(CompIntfc.BLG_PERS_DTA_CI);
If &oBlgPersDtaCi = Null Then
    errorHandler();
    throw CreateException(0, 0, "GetCompIntfc failed");
End-If;
```

Then, we need to set some modes. Setting Interactive Mode to true is like turning off deferred processing. Get History Items is like checking the Include History box on the search page. Edit History is like checking the correction mode on the search page.

```
rem ***** Set the Component Interface Mode *****;
&oBlgPersDtaCi.InteractiveMode = True;
&oBlgPersDtaCi.GetHistoryItems = True;
&oBlgPersDtaCi.EditHistoryItems = True;
```

Then, we update the Empl ID field on the search page:

```
rem ***** Set Component Interface Get/Create Keys *****;
&oBlgPersDtaCi.EMPLID = &emplid;
```

Then, we click the search button. If this fails, that means no rows were returned.

```
rem ***** Execute Get *****;
If Not &oBlgPersDtaCi.Get() Then
    rem ***** No rows exist for the specified keys.*****;
    errorHandler();
    throw CreateException(0, 0, "Get failed");
End-If;
```

Then, we should be on the page where we can make changes. To keep things simple, we are going to work with just one single field on the page — the birth state. First, we print the value to the log. Then, we will either set it to XX to generate an invalid value error or we will toggle the state between FL and TX. Finally, we call the error handler to see if we generated any errors.

```
rem ***** Begin: Get/Set Component Interface Properties *****;
rem ***** Get/Set Level 0 Field Properties *****;
&fileLog.WriteLine("&oBlgPersDtaCi.BIRTHSTATE = " | &oBlgPersDtaCi.BIRTHST.
If &makeError Then;
    &oBlgPersDtaCi.BIRTHSTATE = "XX";
Else;
    If &oBlgPersDtaCi.BIRTHSTATE = "TX" Then;
        &oBlgPersDtaCi.BIRTHSTATE = "FL";
    Else;
        &oBlgPersDtaCi.BIRTHSTATE = "TX";
```

```

        End-If;
    End-If;
    errorHandler();
    rem ***** End: Get/Set Component Interface Properties *****;

```

Then, we save the component:

```

    rem ***** Execute Save *****;
    If Not &oBlgPersDtaCi.Save() Then;
        errorHandler();
        throw CreateException(0, 0, "Save failed");
    End-If;

```

Then, we cancel. This is like going back to the search page.

```

    rem ***** Execute Cancel *****;
    If Not &oBlgPersDtaCi.Cancel() Then;
        errorHandler();
        throw CreateException(0, 0, "Cancel failed");
    End-If;

```

Then, we finish up our try, catch and our function.

```

    catch Exception &ex
        rem Handle the exception;
        &fileLog.WriteLine(&ex.ToString());
    end-try;
    End-Function;

```

Now, this is the start of the main part of the program. We will set our Empl ID, open the log file and get the current session:

```

&emplid = "KU0001";
rem ***** Set the Log File *****;
&fileLog = GetFile("C:\temp\BLG_PERS_DTA_CI.log", "w", "a", %FilePath_Absolute);
&fileLog.WriteLine("Begin");
rem ***** Get current PeopleSoft Session *****;
&oSession = %Session;

```

Then, we call our function to use the CI:

```

updateCI(&emplid, False);

```

Finally, we wrap up the log file.

```

&fileLog.WriteLine("End");
&fileLog.Close();

```

Full Program

And, here is the full program:

```

Local File &fileLog;

Local ApiObject &oSession, &oBlgPersDtaCi;

Local string &emplid;

Function errorHandler()

Local ApiObject &oPSMessageCollection, &oPSMessage;

```

```
Local number &i;

Local string &ErrMsgSetNum, &ErrMsgNum, &ErrMsgText, &ErrType;

&oPSMessageCollection = &oSession.PSMessages;

For &i = 1 To &oPSMessageCollection.Count

&oPSMessage = &oPSMessageCollection.Item(&i);

&ErrMsgSetNum = &oPSMessage.MessageSetNumber;

&ErrMsgNum = &oPSMessage.MessageNumber;

&ErrMsgText = &oPSMessage.Text;

&fileLog.WriteLine(&ErrType | " (" | &ErrMsgSetNum | "," | &ErrMsgNum | ") -

End-For;

rem ***** Delete the Messages from the collection *****;

&oPSMessageCollection.DeleteAll();

End-Function;

Function updateCI(&emplid As string, &makeError As boolean)

try

rem ***** Set the PeopleSoft Session Error Message Mode *****;

rem ***** 0 - None *****;

rem ***** 1 - PSMessages Collection only (default) *****;

rem ***** 2 - Message Box only *****;

rem ***** 3 - Both collection and message box *****;

&oSession.PSMessagesMode = 1;

rem ***** Get the Component Interface *****;

&oBlgPersDtaCi = &oSession.GetCompIntfc(CompIntfc.BLG_PERS_DTA_CI);

If &oBlgPersDtaCi = Null Then

errorHandler();

throw CreateException(0, 0, "GetCompIntfc failed");

End-If;

rem ***** Set the Component Interface Mode *****;

&oBlgPersDtaCi.InteractiveMode = True;

&oBlgPersDtaCi.GetHistoryItems = True;

&oBlgPersDtaCi.EditHistoryItems = True;

rem ***** Set Component Interface Get/Create Keys *****;
```

```
&oBlgPersDtaCi.EMPLID = &emplid;

rem ***** Execute Get *****;

If Not &oBlgPersDtaCi.Get() Then

rem ***** No rows exist for the specified keys.*****;

errorHandler();

throw CreateException(0, 0, "Get failed");

End-If;

rem ***** Begin: Get/Set Component Interface Properties *****;

rem ***** Get/Set Level 0 Field Properties *****;

&fileLog.WriteLine("&oBlgPersDtaCi.BIRTHSTATE = " | &oBlgPersDtaCi.BIRTHSTATE);

If &makeError Then;

&oBlgPersDtaCi.BIRTHSTATE = "XX";

Else;

If &oBlgPersDtaCi.BIRTHSTATE = "TX" Then;

&oBlgPersDtaCi.BIRTHSTATE = "FL";

Else;

&oBlgPersDtaCi.BIRTHSTATE = "TX";

End-If;

End-If;

errorHandler();

rem ***** End: Get/Set Component Interface Properties *****;

rem ***** Execute Save *****;

If Not &oBlgPersDtaCi.Save() Then;

errorHandler();

throw CreateException(0, 0, "Save failed");

End-If;

rem ***** Execute Cancel *****;

If Not &oBlgPersDtaCi.Cancel() Then;

errorHandler();

throw CreateException(0, 0, "Cancel failed");

End-If;

catch Exception &ex
```

```
rem Handle the exception;

&fileLog.WriteLine(&ex.ToString());

end-try;

End-Function;

&emplid = "KU0001";

rem ***** Set the Log File *****;

&fileLog = GetFile("C:\temp\BLG_PERS_DTA_CI.log", "w", "a", %FilePath_Absolute);

&fileLog.WriteLine("Begin");

rem ***** Get current PeopleSoft Session *****;

&oSession = %Session;

updateCI(&emplid, False);

&fileLog.WriteLine("End");

&fileLog.Close();
```

The Result

[See the App Engine test post](#) for instructions on how to run the program. If it worked, you should get this in the log file:



The standard out of the program should look like this:



And, the page should have changed the birth state:

ORACLE

Menu

- My Favorites
- Stephens Development
- Windstar Interface
- Self Service
- Manager Self Service
- Recruiting
- Workforce Administration
 - Personal Information
 - Biographical
 - Organizational
 - Relationships
 - Personal Relationships
 - Citizenship
 - Disability
 - OEE Workforce Survey
 - CAN
 - Diversity Compliance
 - NLD
 - I-9 Forms
 - Add a Person
 - Manage Hires
 - Modify a Person
 - Person Organizational Summary
 - Search for People
 - Badge

Biographical Details **Contact Information** **Regional**

Douglas Lewis Person ID: KU0001

Name Find View All First 1 of 1 Last

Effective Date: 01/01/1980

Format Type: English

Display Name: Douglas Lewis [View Name](#)

Biographic Information

Date of Birth: 06/29/1947 63 Years 6 Months Date of Death:

Birth Country: USA United States

Birth State: **FL** Florida

Birth Location:

Waive Data Protection ☐

Biographical History Find View All First 1 of 1 Last

*Effective Date: 01/01/1980

*Gender: Male

*Highest Education Level: A-Not Indicated

Share this: [Facebook](#) [Print](#) [Digg](#) [Reddit](#) [StumbleUpon](#)

Possibly related posts: (automatically generated)

- [Siebel Introduction](#)
- [Sharing, Writing, Creating](#)
- [Pre-Sterilized Single Use Assemblies](#)

[Leave a Comment](#)

Like Be the first to like this post.

Leave a Comment »

No comments yet.

[RSS feed for comments on this post.](#) [TrackBack URI](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Name *

Email *

Website

Comment

You may use these HTML tags and attributes: <abbr title=""> <acronym title=""> <blockquote cite=""> <cite> <code> <pre> <i> <q cite=""> <strike>

Post Comment

☐ Notify me of follow-up comments via email.

☐ Subscribe to this site by email

• Search

• Pages

- [About](#)
- [PeopleTools Tables](#)
- [Resources on the Internet](#)
- [Step By Step](#)

• a

Select Category

• Blogroll

- [Blog at WordPress.com.](#)
- [Blog at WordPress.com.](#)

• Statistics

-  Site Meter

• Top Posts

- [PeopleTools Tables](#)
- [Step By Step](#)
- [Explorations in Component Interface](#)
- [Y2K10 PeopleSoft Style](#)
- [Component Interfaces Explorations: Security](#)
- [Explorations in Component Interface: PeopleCode](#)
- [ORA-01502: Indexes in unusable state](#)
- [Server Administration Tip: Clearing Web Server Cache](#)

- [Step By Step Virtual PS Install: PeopleTools Install](#)
- [Step By Step Virtual PS Install: Tuxedo Install](#)

• Archives

Select Month



• Meta

- [Register](#)
- [Log in](#)
- [Entries RSS](#)
- [Comments RSS](#)
- [WordPress.com](#)

Theme: [Rubric](#). [Blog at WordPress.com](#).

