Using DecodePCODE_Merge.bat to reapply your customizations during an upgrade

Background: if you are doing an application upgrade (say, from HCM 8.9 to 9.1), and you have PeopleCode customizations, you'll have to reapply those customizations (if you want to keep them).

The upgrade process will lead you to create a project called UPGCUST , which you can use to create compare reports with the new vanilla environment (here called HRDMO91).

Now, hopefully you know Oracle's best-kept secret: the option to generate reports you can view in your browser (after selecting Tools > Compare and Report > To Database, click Options, then check Generate Browser Reports on the Report Options tab).

You can then see your compare reports in a browser:

It's certainly better than the old compare report, but it's still just a report – the task of reapplying your customizations remains to be done. The report shows you the diffs but not whether it's you or Oracle that changed something.

Technically, you want to do a three-way merge. There are two branches of the code with a common ancestor: the vanilla environment that you customized (HRDMO89, in this case). Oracle has worked on it, adding new features, fixing bugs and introducing new bugs; the result is HRDMO91 in our example. You have also started from this common ancestor and customized it; the aim is now to get both Oracle's and your changes applied to the code.

The bad news is that this is not always possible – the two sets of changes may interfere and/or it's not always clear which changes must prevail; the good news is that there are tools available for three-way merges.

DecodePCODE_Merge uses one of these tools (svnkit, the library to work with Subversion) to do the actual three-way merge (it uses a temporary repository so you do not need to set up Subversion). Its final result is a modified browser report (see the last screenshot in this document) with links to separate files containing PeopleCode: the three versions you started with (DMO89, DMO91, and HRDEV, in our example) plus a new version with the merged code (if successful). Diff files are also provided.
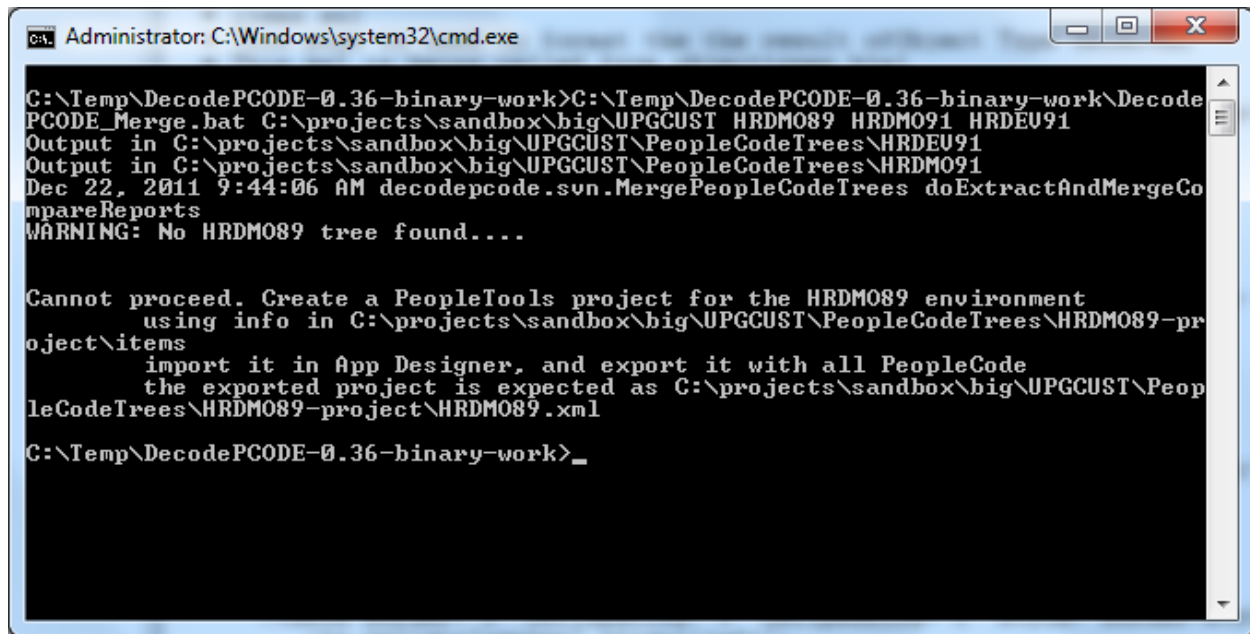
Preparation

Two external programs are required: GNU diff and GNU patch (the patch program is necessary because one of the stylesheets for the browser reports will be modified). The location of the programs must be specified in decodepcode.properties (GNUdiff and GNUpatch entries). It is also necessary to have your UPGCUST browser report available.

Running

As a first step, run DecodePCODE_Merge.bat with four parameters: the directory for the compare report, followed by the names for your environments (old demo, new demo, and new development).

The program will extract the PeopleCode that is in the browser report (which is actually in .XML files) to separate .pcode files in a new subdirectory PeopleCodeTrees below the report directory. It cannot do the actual merge yet, because it does not have the PeopleCode of the old demo environment.

Unfortunately, it's a bit tricky to get this PeopleCode because you need the UPGCUST project definition. Your old demo environment is probably on an older PeopleTools version, so that you cannot just copy the UPGCUST project definition to it. To do this, there are two methods:

1) (easiest, but you need database-level access to your old demo environment): create a PeopleTools project in your old demo environment, with the same name as the environment (HRDMO89 here). Save it and close App Designer. Execute the SQL script in the PeopleCodeTrees\<old demo>-project\items subdirectory (as indicated when in the previous step). Reopen AppDesigner and the project, and copy the project to file. Copy the .xml to the indicated location.

2) (slightly messier, since you have to edit a project .xml file). Create an empty PeopleTools project in your old demo environment, with the same name as the environment (HRDMO89 here). Copy it to file and close App Designer. Open the .XML file with a text editor, locate the line with <lpPit> POINTER (about line 34), and insert the contents of .items file in PeopleCodeTrees\<old demo>-project\items subdirectory below that line. Reopen AppDesigner. Do a Copy From File step to import the project file (overwriting it); since you only want to project definition, you can deselect all object types in this step. Then, do another Copy To File step, this time with all object types selected.

Whichever method you selected, you must now have a project file(with PeopleCode) in the location that the program displayed in the previous step (…\PeopleCodeTrees\HRDMO89-project\HRDMO89.xml).

Now, run DecodePCODE_Merge.bat again (same parameters). This time, it should take considerably longer. When it's done, open the Browser Report again:
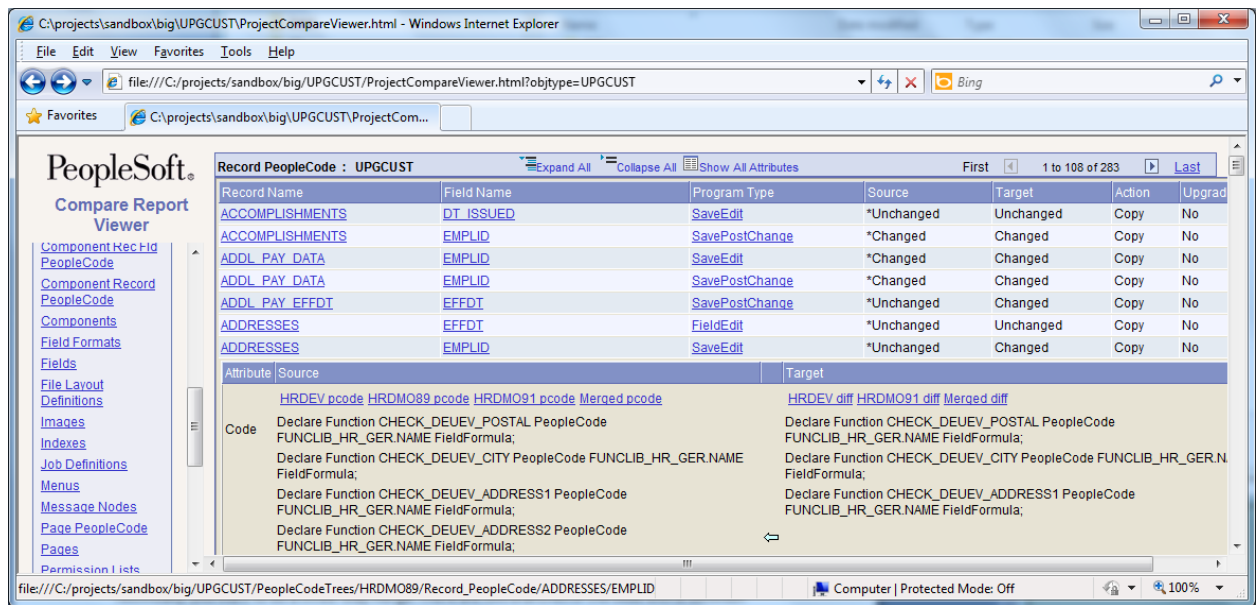
When you click on one of the rows, you will see an extra line with hyperlinks just above the PeopleCode diff. The number of links varies (depending on whether the code was found, whether the merge was successful and whether files are identical), but there can be four links to .pcode files (HRDEV, HRDMO89, HRDMO91 and Merged), as well as three diff files (the diffs themselves contain a line indicating what code has been compared).

As indicated above, a three-way merge is not always successful – it will fail if the three versions of the code are too different, or if the two branches have changed the same lines of code. Even if the Merged.pcode link is absent, you may find it convenient to have the various versions of the code and their diffs available.