Scientific Notebook No. 170E: Total Performance Assessment (TPA) 3.2 Development (April 3, 1996 through September 16, 2003)

SCIENTIFIC NOTEBOOK

170-11E

SCIENTIFIC NOTEBOOK

Christopher E Walker

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

May 14, 2001 – Spent the day reading through the TPA Version 4.0 Users Guide to get a feel for what the TPA project is all about.

May 15 – Spent most of the day reading the TPA Version 4.0 Users Guide. Installed a copy of Visio on my computer so I could create some sample screen shots.

May 16 – I worked with Visio all morning and then decided I didn't know enough about Java to know if I could create the screens I had in mind so I wrote some Java code to make some sample screen shots. The Swingset tutorial was really helpful.

May 17 – Worked with Java, Visio on sample screen shots.

May 18 – Spent the day in the TPA Users guide looking and sorting variables. Received Appendix A in electronic format from Sitikanta.

May 21 – Worked with Java prototyping code.

May 22 – I am beginning to understand what Sitikanta is after but the SRD still needs some shoring up. Continued to sort data and work on the Java prototype.

May 23 – Spent the day with Java prototyping and Visio screen shots.

May 24 – Began thinking about organizing data objects in Java around a single theme. I don't think they can all fit into the scheme so I will have a few different objects for unique data. Worked with data sorting and reading TPA Version 4.0 Users Guide.

May 25 – Continued sorting data from Appendix A and working with Java data classes.

May 29 – Continued reading Users guide and creating data objects.

May 30 – Began programming Java data objects with the sorted parameters from appendix A. I should refer to variables as parameters.

May 31 – Spent the day creating data objects in Java and recording these changes in Visio.

June 2001

June 1 – Spent the day creating data objects in Java and recording these changes in Visio. Same as Yesterday.

June 4 – Began programming GUI classes based upon the data classes.

June 5 – Worked on GUI classes and data classes in Java

June 6 – Continued working on GUI and data classes in Java

June 7 – Spent most of the data updating the Visio Screen shots with those created from Java.

June 8 – Worked with Java and Visio on GUI screens.

June 11 – Continued to program Java prototype.

June 12, 2001 – Continued to program Java prototype.

June 13 – Continued to program Java prototype.

June 14 – Spent most of the day researching how to create certain screens and use some of the Java libraries to their full extent.

June 15 - Continued programming Java prototype.

June 18 – I needed to rework some of the Java data objects because I did not have the data sorted into the proper objects, once I completed that I began to update the Visio documents.

June 19 – Talked to Sitikanta today and he wants me to cease prototyping and work on the SRD until it is complete. I was almost done with the prototype but will now shift gears and work on the SRD. So I worked on SRD and sample screen shots in Visio for most of the day.

June 20 – Worked SRD and sample screen shots in Visio.

June 21 – Worked on SRD and began researching the distribution functions and how difficult they will be to program in Java.

June 22 – continued to analyze distribution functions and work on SRD and Visio documents.

June 25 – Worked on Visio GUI design for screen shots and data classes.

June 26 – Worked on Visio GUI design for screen shots and data classes.

June 27 – Worked on Visio GUI design for screen shots and data classes.

June 28 – Same as yesterday

June 29 – Same as yesterday, began to design Java objects.

July 2001

July 2 – designing Java objects and looking at ways to integrate outside files into the code.

July 3 – Designing Java objects

July 5 – programming TPAI to handle new tree objects.

July 6 – programming TPAI for tree objects and normal data datasets.

July 9 – programming TPAI for tree objects and UZFT screens.

July 10 – programming SZFT and nuclide screens.

July 11 – Programming TPAI, updating subareas and nuclide screens.

July 12 – Programming TPAI, updating subareas and nuclides screens.

July 13, 2001 – Programming the save data file screens.

July 21 – worked on SDP and design issues because of new (old requirements I hadn't considered) requirements.

July 22 – worked on SDP, design issues and researching Java questions

July 23 – worked on SDP, design issues and researching Java questions

July 24 – worked design problems and reworking TPAI code to accept new design.

July 25 – modified TPAI Objects and research Java problems.

July 26 – modified TPAI Objects and worked on design documents.

July 27 – worked on SDP and TPAI code.

July 28 – worked on TPAI Code and researched Java issues.

August 2001

August 17 – Worked on TPAI code. Adding colors to certain areas in the code.

August 20 – Continued to work on the TPAI code for saving files and maintaining the correlate parameters sections.. Had a bunch of trouble putting a JComboBox into the correlate parameters panel but that has been fixed. I was not adding the combo boxes on the proper panel, which caused the boxes to appear in an incorrect fashion.

August 24 – Created a class named DefaultRationalization.java. This class contains an array of BasicData items that are used to hold the default rationalization of the parameters. This class was necessary because I need to know the following states for rationalization.

1. When the current rationalization matches the default rationalization.
2. When the current rationalization does not match the default and the current has not been changed since the user opened a new file.
3. When the current does not match the default and the current has been changed since the user opened a new file.

It was also necessary because I needed to be able to store rationalizations for data parameters that are newly created (such as adding a nuclide or sub area) and save the default for those parameters that were removed (such as removing a nuclide or subarea).

September 2001

September 11- Finally got the background paint functionality to work. I received a snippet of code from Donna Jeffreys: that showed how to draw an Image on the background. All panels located on the Jframe or Jpanel must have transparency set to false (*this->setTransparency(false) and yes I know that is counter-intuitive*) so the background image will show through

```
public void paint(Graphics g)
{
        if (myImage != null)
                g.drawImage(myImage,0,0,getWidth(),getHeight(), 0,0,myImage.getWidth(this),
                myImage.getHeight(this),  this);
}
```

September 20, 2001 – Had a long talk with Sitikanta during which we discussed many of the nuances of what he would like the tpai code to perform. They include making a startup screen, increasing the font size, changing the other input file screens to mono-spaced fonts, moving the files to the tree structure, and creating a Mean Value File. I am still uncertain as to the correctness of the graphs and am looking for some corroborating data to test the graphing capabilities against.

September 27 – Programmed a new routine to display the sub areas. The issue is that we wanted to use the entire space within the grid on the MainPanelDiscrete screen and still maintain the correct aspect ratio. The algorithm to that is as follows:

1. Calculate the minimum and maximum UTM coordinates for all subareas.
2. Decide which direction to stretch the subareas by comparing the following ratios. The two ratios will always be between zero and one. The ratio that is closest to one gives the direction that has the shortest distance to travel and therefore is the controlling ratio on the aspect change.

```
//rationNorth = (YMax - YMin) / (YO2 - Y01)
double ratioNorth = (calculatedNorthMaxHeight - calculatedNorthMinHeight) /  (UTMNorthingMax - UTMNorthingMin);

//ratioEast = (XMax - XMin) / (X02 - X01)
double ratioEast  = (calculatedEastMaxHeight - calculatedEastMinHeight) /  (UTMEastingMax  - UTMEastingMin) ;

// Whichever ratio is closest to 1 but greater than zero wins
if (ratioNorth > ratioEast)
{
    calculatedEastMaxHeight += ratioNorth * (UTMEastingMax – calculatedEastMaxHeight) ;
    calculatedEastMinHeight = ratioNorth * (calculatedEastMinHeight – UTMEastingMin) +  UTMEastingMin;
}
else
{
   calculatedNorthMaxHeight = ratioEast * (UTMNorthingMax – calculatedNorthMaxHeight) + calculatedNorthMaxHeight;
   calculatedNorthMinHeight = ratioEast * (calculatedNorthMinHeight – UTMNorthingMin) + UTMNorthingMin;
}
```

3. Finally, calculate the proper multiples and create the polygons to draw to the screen

```
// calculate the multiple used on the individual pixels to display in given area of screen
double MultipleHeight = (double)height / (double)AreaHeight;
double MultipleWidth  = (double)width  / (double)AreaWidth;

// ScreenData will hold the actual X,Y coordinates to draw to the screen
screenData = null;
screenData = new Polygon[NumSubAreas];

for (int i=0; i<NumSubAreas; i++)
{
  screenData[i] = new Polygon();
  for (int y=0; y<4; y++)
```

```
  {
    screenData[i].addPoint(
      (int)((subAreaData.subAreaData[i][y][0]-calculatedEastMinHeight) * MultipleWidth +100),
      ((getHeight()-25) – (int)((subAreaData.subAreaData[i][y][1]-calculatedNorthMinHeight) *
       MultipleHeight)));
  }
}
```

October 5, 2001 – Received new instructions for the TPAI GUI code from Sitikanta. Most of these are strait forward. Currently I have a TabbedPane and I add 1 – 7 panels into the tabbed pane depending of which module is chosen from the conceptual model screen. Sitikanta wants there to be only one pane at a time on the screen and to change the tree from a tree to a series of panes with buttons on it. The first part is easy enough although time-consuming. The second part may necessitate a change to the underlying data structures. I am going to need to write a new design document to determine where the changes would be less intrusive.

October 9 - I added two new class TreeViewSelection.java and TableViewSelection.java. These classes allow the user to set up his or her individual preferences for the font that the user wants to see on the trees and the tables and the also the foreground and background colors. This information is also written to a configuration file "tpai.cfg" on the harddrive in the same directory that holds tpai.jar. If we switch this over to an applet, this configuration idea will need to be rethought because the applet will not have access to reading and writing files on the harddrive.

Chris Walker
Initials _____

Scientific Notebook 170 – 11E

Printed 10/03/01

Entries into Scientific Notebook #170-11E for pages 1 - 5 have been made by Chris Walker 10/3/01.

No original text entered into this Scientific Notebook has been removed.

Christopher E Walker 10/3/01.

I have reviewed this scientific notebook and find it in compliance with QAP-001. There is sufficient information regarding methods used for conducting tests, acquiring and analyzing data so that another qualified individual could repeat the activity.

James Winterle

11-09-01

7 7

JAW 12-2-05