

SE-280 Software Engineering Process

Dr. Mark Sebern – Winter quarter 2011-2012

Midterm report

Revised: 13 January 2012

Report assignment

Retrospectives are an important part of any quality improvement effort. With any process, it is necessary to periodically review the effectiveness of process elements to determine how those elements can be modified or extended in order to make the process work in your favor: that is, to make the end result more effective, or to make the process easier to work with.

In this assignment, you pause to assess your current personal software process up to this point, in part by reviewing the data you've been accumulating within Process Dashboard. In this review, you are to determine how the personal software process and its various elements have affected the quality of your work.

Use a defined process to analyze your data and prepare a report summarizing the analysis and your conclusions. Based on this analysis, prepare initial versions of your personal design review and code review checklists.

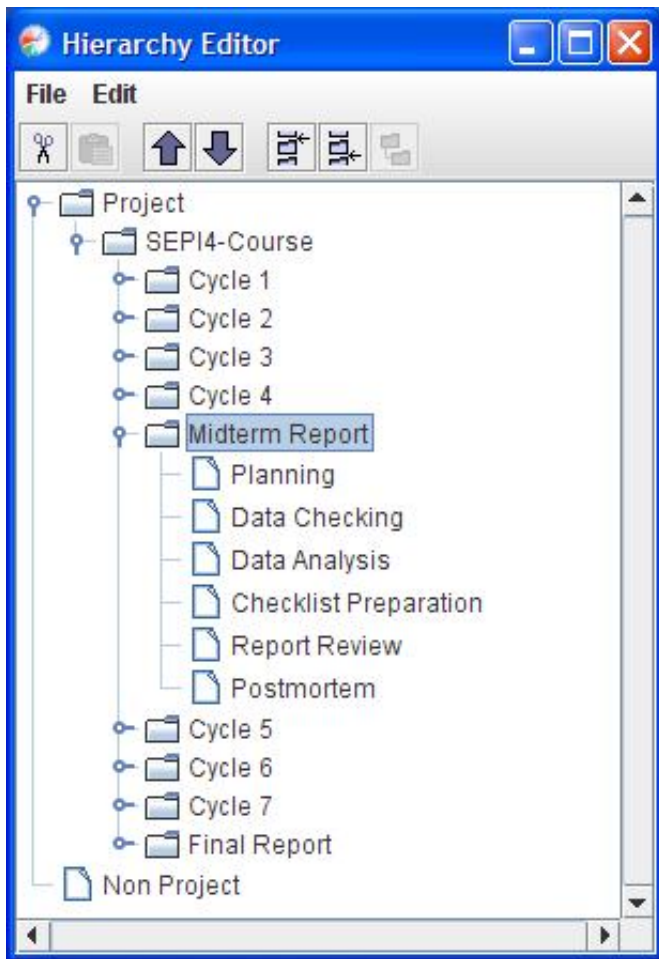
Setting up the report process in Process Dashboard

When you are doing your own work, especially non-programming tasks, you may need to define your own process to plan and track the work. The Process Dashboard tool supports this by means of "generic" processes that can be customized. For the midterm report, you will make use of this capability, using the following process phases (more details follow below):

- Planning
- Data Checking
- Data Analysis
- Checklist Preparation
- Report Review
- Postmortem

The **Planning** and **Postmortem** phases are automatically included in the generic process template. To add the other phases, open the **Hierarchy Editor** window in Process Dashboard and select the "Midterm Report" node. Then, select **Edit->AddTemplate->Generic Phase** and type in the phase name (e.g., "Data Checking"). You can use the **Edit->Move Up** and **Edit->Move Down** functions to correctly order your phases, though the planning phase must always be first and the postmortem phase must be last. Once you have defined these phases, you can use them in a manner similar to the predefined phases you have worked with in the programming projects.

When you have finished, the **Midterm Report** portion of the Process Dashboard hierarchy should look like this:



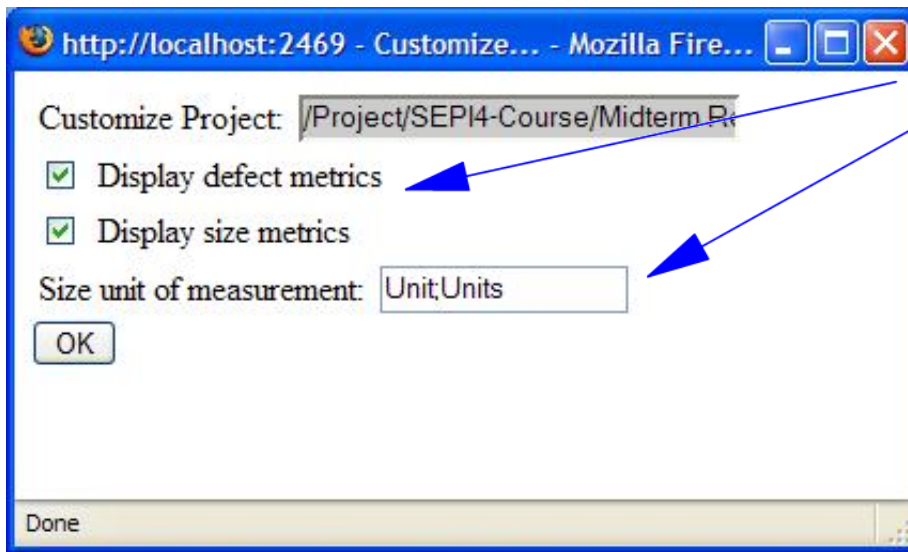
When you are satisfied with your generic project definition, close the hierarchy window. Next, open the generic **Project Summary Form**, accessible from the "document" button on the Process Dashboard main bar.

For some generic processes, it is useful to measure the size of some product (e.g., the number of pages in a report). You must define and use **an appropriate size measure for your midterm report**. To do so, click on the **Customize** button (on the **Project Summary Form**) to pop up the customization dialog box. There, you can set the size unit (e.g., **report pages or number of analysis questions**) to be used for your report size estimates. To **use "report pages" as a size measure**, for example, change the size unit **field from "Unit;Units" to "Page;Pages"** (singular and plural forms).

If, for some other project, you do not wish to estimate and measure report size, you can uncheck the **Display size metrics** option in the **Customize** window.

Similarly, the **Display defect metrics** option allows you to choose whether to track defects; in your report "project", **leave this option enabled and log defects**, just as you do for programming assignments.

When you have finished this customization, click on the OK button to confirm your choices. The Customize window looks like this:

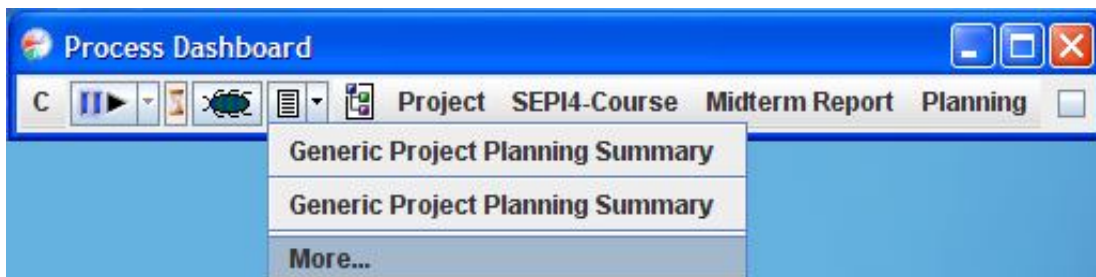


If you have questions, ask the instructor right away!

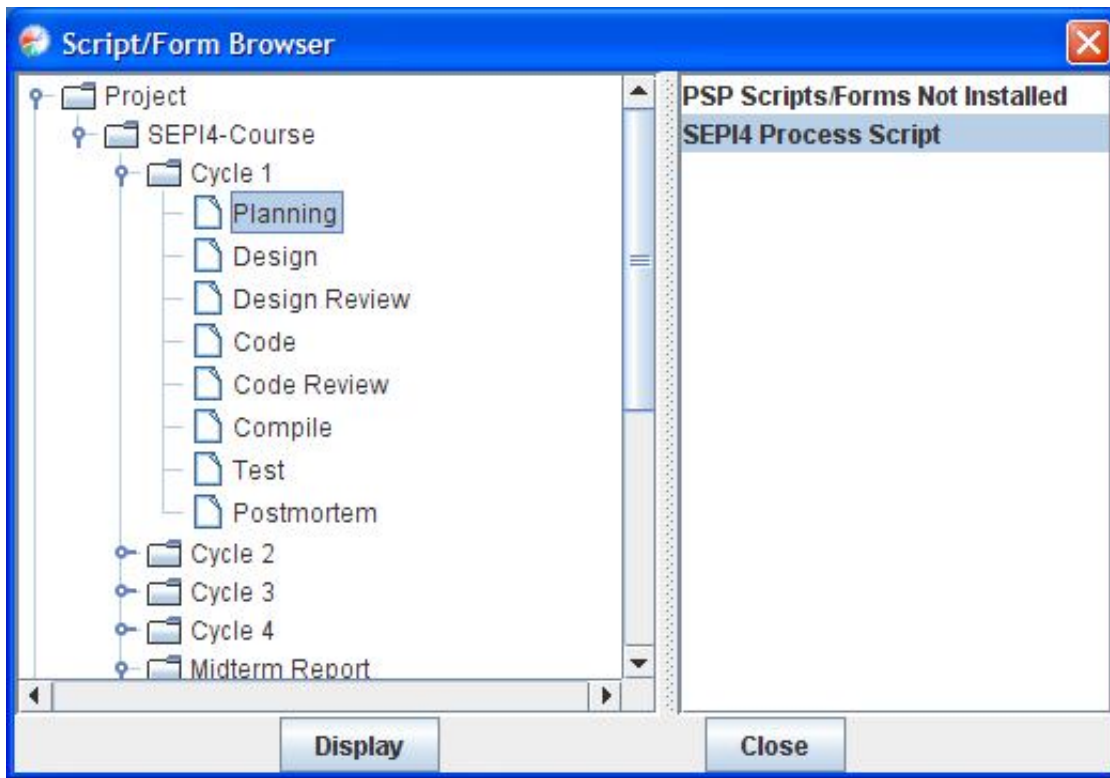
Planning

As you complete this report assignment, **log your time and defects**, and **generate PIPs in the usual way**. In the planning phase, use the *Project Summary Form* to document your plan, **making time estimates for each phase**.

As you are working on your data analysis, you will likely need to go back and look at data or forms from one or more previous projects. **If you simply "navigate" to these other projects, your time logging may be disrupted (e.g., current time logged to previous projects)**. To avoid this problem, open the **Script/Form Browser** by using the drop down menu from the "document" button on the Process Dashboard main window bar:



If you select the **More...** option, a new window will open and allow you to select a cycle or task from your hierarchy. For each of these items, the corresponding scripts and forms will be listed on the right side of the window. You can then open one of these documents by selecting its entry in the list and clicking the **Display** button.



In this example, the Cycle 1 process script can be used as an entry to all the Cycle 1 forms.

Data checking

The Process Dashboard provides a number of data analysis reports. You can find these reports in the following places:

- In the links at the bottom of the process script, in the **Historical** section (accessible only in the context of one of your development cycles, not in the context of the midterm report project):
 - Estimating accuracy summary
 - Defect density summary
 - Defect log summary
- Linked from the C->Data Analysis menu choice:
 - Report R3
 - Report R4
 - Pareto
 - Other charts Some of these reports use outdated terminology (e.g., "new & changed" instead of "added & modified"), but the meaning should be relatively clear.

These reports may help you to address some of the questions in the **Data analysis** section below. First, though, use them to thoroughly check your process data for cycles 1-4. Since you will be drawing conclusions from your data, it is very important that all the data is correct. At a minimum, you should verify the following:

- Using the Defect log summary report, make sure that all defects from cycles 1-4 are included. Verify the injection and removal phases, the defect type, and the fix time. Make sure that the description is clear and represents the actual defect (e.g., "subscript off by one") and not a symptom (e.g., "incorrect average value calculated").
 - If you find any errors, fix them using the **C->Defect Log** menu choice. Then run the Defect log summary report again to verify the corrections.
- Check over time logs for each cycle and correct any errors using the **C->Time Log** menu choice.

- Prepare a draft of a report section presenting the results of this data checking activity.

Data analysis

In preparation for writing your report, you should analyze the data you have gathered to make the following assessments:

1. Consider how adapting to the PSP has affected your previous development process, specifically:
 - a. any issues or problems you've encountered in adapting to the personal software process, and whether they have been a help or a hindrance so far.
 - b. what elements were particularly helpful/enlightening/annoying/.
 - c. your Process Improvement Proposals (PIPs) from cycles 1-4. Make notes of any new ones that occur to you.
2. Analyze the impact of the PSP on the quality of your planning, specifically:
 - a. how the PSP has affected your estimating accuracy for both program size (LOC) and development time.
 - b. whether the data indicates any definite trends thus far, or is it too early to tell - or if the data you have gathered suggests any possibilities for future improvement to your planning.
 - c. any surprises (pleasant or otherwise) you have discovered in using the PSP to help you plan your development projects.
 - d. whether using the PSP has helped you identify any previous problems with your planning, and whether you think you still have problems with planning. Describe these, and indicate which are most important.
 - e. the results of the **Data Checking** activity described above.
3. Analyze the defects you have encountered in the various phases of development.
 - a. Make sure that you have in fact logged all the defects you have injected. If there are some defects you have injected, but for some reason you did not log them, go back and correct the defect log as noted above. If that is not practical, at least consider those defects in your current analysis.
 - b. Identify the defects which occur most frequently. (The R3 and R4 reports from the Process Dashboard Data Analysis tools may be helpful here.)
 - c. Identify the kinds of defects you think are the most time-consuming to fix.
 - d. Identify which defects you think are most serious - that is, whether they are the ones encountered most frequently or the ones that take the longest to fix.
 - e. Categorize your defects according to the phases in which they are injected (where they occur or were created, not where they were found).
 - f. Think about what the defects might suggest to you about your current software practices, and whether any particular phase of development deserves special attention. Determine the types of mistakes that resulted in these defects. Define actions that might help you to prevent these defects.
 - g. Consider particular actions that might help you to find and fix defects before they are discovered in the **Test** phase. Does your process data indicate whether removing defects earlier than the **Test** phase makes sense from a time-savings standpoint?
 - h. Group your own defects, according to similar characteristics, and make a list of these categories. For example, you might group your defects as follows:
 - Missing semicolon
 - Copy and paste logic error
 - Incorrect parameters for a library class method
 - Off-by-one error in loop condition
 - Missing check of possible error/exception condition
 - Missing method in the designThe idea is to look at the defects that you make, and try to find out which kinds are your "favorites", so that you can take steps to avoid them or to find them early in the development process.
4. Prepare a draft of the section of your report that presents the results of this data analysis activity. Along with

your analysis data, tables, and charts, you must include, at a minimum:

- a. A discussion of PSP issues.
- b. A summary of how PSP has impacted the quality of your planning (that is, estimating accuracy).
- c. A defect analysis summary.
- d. Conclusions you have drawn about your software development process and your coursework to date.

Checklist preparation

Based on your defect analysis, **prepare an initial design review checklist and code review checklist, in** the format discussed in class. The design review checklist should contain the kinds of defects you inject during design, while the code review checklist should contain the kinds of defects you inject while coding or design defects that may become more evident when they are expressed in code.

Important note: **As part of the preparation of your design review checklist, consider how you plan to adjust your design documentation methods, in order to make your designs more reviewable. In your report, describe the steps you are taking** and how you will evaluate the results of implementing them.

When you are done, each checklist should have **at least ten items. It is** OK to have more; if you can't find enough based on your defects from cycles 1-4, complete the list with ones that you have had trouble with in the past, or choose some likely candidates from the textbook's sample design review checklist (page 184) and a code review checklist (page 175). Don't make these items too general, or they won't be useful as a focus for future reviews.

Report review

When your draft report (including checklists) is complete, review it. Make sure that all specified topics have been covered, and check the accuracy of your data and calculations. Fix any defects that you find, logging them in the Process Dashboard.

Postmortem

Using the **Project Summary Form**, check the actual time and defect data for this report assignment. Since you were required to estimate the size (e.g., report pages), measure and record the actual size as well. Add any PIPs about the report process, including questions or comments that you have.

Report

Submit your report via Subversion.

Your lab report should include the following items in a tag named "**MidRpt**":

1. Your Process Dashboard data, in a ZIP file named "usernameData.zip", where "username" is your MSOE email username, in your project's **procdData** directory. Use the **C->Tools->Save Data Backup** menu choice to create the ZIP data file. Put this file in a folder named *procdData* inside your top-level project directory (i.e., *procdData* and *src* should be siblings, at the same level). Make sure that the project and all phases are marked complete.
2. A single Acrobat format (PDF) file in your project's **reports** directory, with the following components, **in order**:
 - The (generic process) *Project Summary Form*. In the *PIPs* section:
 - You should include your process improvement proposals and proposed solutions; they don't have to be earth-shaking in importance, but your entries should reflect some thought about your process experience.
 - Note that any questions and comments about the assignment should also be included here.

- Your report text.
- Your midterm report time log (linked from the bottom of the *Project Summary Form*).
- Your midterm report defect log (linked from the bottom of the *Project Summary Form*).
- Your design review checklist. (Make sure you use the format discussed in class.)
- Your code review checklist. (Make sure you use the format discussed in class.)
- Your R3 and R4 reports from the Process Dashboard analysis tools.

Regarding the PDF report document:

- You should be able to create PDF files for each component by "printing to PDF" using software like PDFCreator or Acrobat.
- You can then combine them into one composite PDF file using the "PDF Split and Merge" application (<http://www.pdfsam.org/>).
- Name the PDF file "SE280usernameMidRpt.pdf", where "username" is your (all lower case) MSOE email username.
- Make sure that your PDF file is correct and complete; in particular, make sure that no part of the **Project Summary Form** is "cut off" because it extends beyond the page margins. You may be able to adjust print scaling in your browser to make it fit.
- If you have difficulties, please consult with the instructor immediately.

Put this file in a folder named **reports** inside your top-level project directory (i.e., **reports** and **src** should be siblings, at the same level).

Acknowledgment

Dr. Mark Hornick made major contributions to the development and definition of this assignment.