17-708: Software Product Lines

Assignment 6: Variability in the Wild

Task

Investigate how variability is implemented in an open-source system of your choice and summarize your findings in a report.

1. Pick a *popular* open source project that is *still actively developed*. You may choose a project you are already familiar with. There are no restrictions on domains or used programming languages. Only projects discussed as case study in the lecture (Linux kernel, Debian, Android) should not be selected.

2. Browse the documentation, build infrastructure, source code, branches/forks, and dialogs in the running system to understand which mechanisms are used for configuration and portability (if any). For example, look for command-line options, configuration files, plugin mechanisms, components, forks, preprocessor use, or build-system variability. Attempt to understand the configuration-related part of the implementation. For example, does the build system orchestrate other variability mechanisms; are implementations of options modularized?

3. Assess the variability implementation. For example, how well is this aspect of the implementation documented for developers and end users? Are the used mechanisms appropriate for the project (why, why not, how could it be improved)? How scattered is configuration knowledge? Consider the qualities of and tradeoffs among implementation strategies discussed in class.

4. Document your findings in a report.

5. Be prepared to share your key findings in class (about 3 minutes, no slides)

Deadlines, Technicalities, and Hints

Send the report as a PDF file by email to the instructor by Nov 4, 11:59pm.

Attempt to understand the configuration-related part of the systems in some detail. However in any large-scale system, you will not be able to fully understand all the details. Attempt to cover configuration in breadth (different mechanisms used) and in depth (selectively for some aspect of the implementation).

The report should include

A. a brief description of the system,

B. a characterization of the configuration options identified (selectively in large configuration spaces),

C. a description of the used variability-implementation mechanisms (ideally mentioning also which strategies are not used),

D. a description of how those variabilities are communicated in the project's documentation, and

E. an assessment as described above.

For B-E, be concrete and provide at least two examples; references to concrete code or documentation artifacts are appreciated. A good report will illustrate an understanding of configuration aspects of the system both in breadth and selectively in depth and will relate them to discussions from the lecture/readings.

The report should not exceed 2500 words (soft limit).

Grading

We expect

- A thorough discussion of configuration-aspects in an open source project
- A report that discussed both code-related and documentation-related aspects of configuration in the system
- A discussion of all relevant implementation strategies
- An assessment of the variability implementation that adds additional insights to the technical discussion and exposes an understanding of the implementation strategies and their tradeoffs.