

Code Smells in Software Product Lines (FOSD 2014)

With this survey, we aim at gathering experiences and thoughts about Code Smells in Software Product Lines. It is meant as an additional source of information, supplementary to recent efforts we took for defining and detecting code smells in the presence of variability.

We intend to provide an overview of the results by integrating it into a presentation at FOSD meeting 2014.

This survey will take approx. 20 minutes.

There are 51 questions in this survey

Participant's Code

First, you have to generate a unique code word to enable the reuse of results with future studies (while preserving anonymity).

This survey is anonymous and we cannot track back any data to a concrete person. However, we may conduct follow-up studies where it may be likely that we ask you again for participation. In this case, it is important to match the data from this survey with such a follow-up study (e.g., an experiment).

To this end, it is important to generate a code word that is known only you. In the following, we explain how you can generate this code word:

Second letter of your mother's first name

Third letter of your father's first name

Fourth letter of your star sign

Second letter of your first name

Third letter of your last name

Please write your answer here:

Personal Data

Initially, we need some information regarding your person.

How old are you?

*

Only integer value may be entered in this field.

Please write your answer here:

Gender? *

Please choose **only one** of the following:

- Female
 Male

Please specify your gender.

What is your current position? *

Please choose **only one** of the following:

- undergraduate student
 graduate student
 PhD student
 Post-doc
 professor (associate/assistance/full)
 Other

At what kind of affiliation your are currently employed? *

Please choose **only one** of the following:

- University
 Research Lab
 Company
 Other

Background & Experience

Next, we need information about your working background and experience regarding programming in general and SPLs in particular.

How many years you are familiar with programming? *

Each answer must be at least 0
Only integer value may be entered in this field.

Please write your answer here:

For this question, we consider the time you learned and practiced a programming language regularly as the "first time of programming"

How often do you program seriously? *

Please choose **only one** of the following:

- daily
 weekly
 monthly
 Other

How experienced are you with the following programming paradigms? *

Please choose the appropriate response for each item:

	very inexperienced	inexperienced	mediocre	experienced	very experienced
logical (Prolog)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
functional (Haskell)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
imperative (C)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
object-oriented (Java)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Have you ever worked on one or more larger programming projects in a company or at the university or are you currently working on a larger programming project? *

Please choose **only one** of the following:

- Yes
 No

In which domain were/are those projects?

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '9 [BEproject]' (Have you ever worked on one or more larger programming projects in a company or at the university or are you currently working on a larger programming project?)

Please write your answer here:

How many lines of code did these projects usually have?

*

Please choose **only one** of the following:

- < 900
 900 -- 40,000
 > 40,000

How do you estimate your programming experience compared to other people in academia/industry in your position and how compared to programmers that have 20 years of experience? *

Please choose the appropriate response for each item:

	clearly worse	worse	identical	better	clearly better
Others in equal position	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Experienced Programmers	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

How experienced you are with Software Product Lines (SPLs)? *

Please choose the appropriate response for each item:

	very inexperienced	inexperienced	mediocre	experienced	very experienced
general programming	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Have you ever worked on programming an SPL? *

Please choose **only one** of the following:

- Yes
 No

In which context you worked on SPL programming? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '14 [BEsplProg]' (Have you ever worked on programming an SPL?)

Please choose **only one** of the following:

academia

industry

both

Other:

How large were the SPL(s) typically? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '14 [BEsplProg]' (Have you ever worked on programming an SPL?)

Please choose **only one** of the following:

< 900

900 -- 40,000

> 40,000

How experienced you are with the following variability implementation techniques? *

Please choose the appropriate response for each item:

	very inexperienced	inexperienced	mediocre	experienced	very experienced
Feature-Oriented Programming	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Aspect-Oriented Programming	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
C/Antenna preprocessor	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Parameters	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Which programming tasks you've been involved during SPL development? *

Please choose **all** that apply:

Design/Implementation from scratch

Extending an existing SPL (e.g., by a new feature)

Modification (e.g., due to change request)

Maintenance (e.g., refactoring, bug fixing)

Other:

Beside programming, are you/have you been involved in any of the following tasks/activities (in the context SPLs)? *

Please choose **all** that apply:

- Analysis of SPLs
- Testing of SPLs
- Teaching SPLs
- Tool building for SPL development
- Other:

Code Smells

Next, we would like to know about your expertise with code smells and anti patterns.

Did you ever hear about code smells? *

Please choose **only one** of the following:

- Yes
 No

Which of the following code smells do you know?

Please choose **all** that apply:

- Duplicated Code
 Long Method
 Large Class
 Long Parameter List
 Divergent Change
 Shotgun Surgery
 Feature Envy
 Other:

Advice: with "knowing", we mean that you at least roughly know, which problem the respective smell describes.

Did you ever hear about anti patterns? *

Please choose **only one** of the following:

- Yes
 No

Which of the following anti pattern do you know? *

Please choose **all** that apply:

God Class

Spaghetti Code

Lazy Class

Data Class

Other:

Variability-Aware Code Smells

Finally, we want to know about your experiences and/or opinion about concrete variability-code smells, that is, smells that are defined by taking variability into account.

To this end, we describe six variability-aware code smells in a .pdf file, attached with the email that contains the link to this survey. In the following, we ask the same questions regarding experiences and consequences for each of these code smells (hence, if you see a question twice, it is not Groundhog day ;)).

Inter-Feature Code Clones

The following questions target at the Smell **Inter-Feature Code Clone**. You can find a description in the attached .PDF file

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

For which implementation techniques you observed this code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '26 [VCSclonesKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

- Feature-Oriented Programming
 C preprocessor
 Antenna preprocessor
 Other:

In which context you observed the described code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '26 [VCSclonesKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

student's implementation

industrial code

open source project

Other:

How do you estimate the severity of this code smell for the aspects, listed below? *

Please choose the appropriate response for each item:

	very problematic	rather problematic	don't know	rather unproblematic	unproblematic
program comprehension	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Maintainability (e.g., error detection/removal)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
evolvability (e.g., add/change code)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Long Refinement Chain

The following questions target at the Smell **Long Refinement Chain**. This smell is specific to FOP and you can find a description in the attached .PDF file.

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

In which context you observed the described code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '31 [VCSrefinementKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

- student's implementation
 industrial code
 open source project
 Other:

**How do you estimate the severity of this code smell for the aspects, listed below?

Please choose the appropriate response for each item:

	very problematic	rather problematic	don't know	rather unproblematic	unproblematic
program comprehension	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Maintainability (e.g., error detection/removal)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
evolvability (e.g., add/change code)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Annotation Bundle

The following questions target at the Smell **Annotation Bundle**. This smell is specific to preprocessor-based variability (e.g., C/Antenna preprocessor) and you can find a description in the attached .PDF file.

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

In which context you observed the described code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '35 [VCSbundleKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

- student's implementation
 industrial code
 open source project

Other:

How do you estimate the severity of this code smell for the aspects, listed below?
*

Please choose the appropriate response for each item:

	very problematic	rather problematic	don't know	rather unproblematic	unproblematic
program comprehension	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Maintainability (e.g., error detection/removal)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
evolvability (e.g., add/change code)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Latently Unused Parameter

The following questions target at the Smell **Latently Unused Parameter**. You can find a description in the attached .PDF file

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

For which implementation techniques you observed this code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '39 [VCSparameterKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

- Feature-Oriented Programming
 C preprocessor
 Antenna preprocessor
 Other:

In which context you observed the described code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '39 [VCSparameterKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

student's implementation

industrial code

open source project

Other:

How do you estimate the severity of this code smell for the aspects, listed below? *

Please choose the appropriate response for each item:

	very problematic	rather problematic	don't know	rather unproblematic	unproblematic
program comprehension	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Maintainability (e.g., error detection/removal)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
evolvability (e.g., add/change code)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Large Feature

The following questions target at the Smell **Large Feature**. You can find a description in the attached .PDF file

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

For which implementation techniques you observed this code smell? *

Please choose **all** that apply:

- Feature-Oriented Programming
 C preprocessor
 Antenna preprocessor
 Other:

In which context you observed the described code smell? *

Please choose **all** that apply:

student's implementation

industrial code

open source project

Other:

Switch Statements With Optional Cases

The following questions target at the Smell **Switch Statement with Optional Cases**. You can find a description in the attached **.PDF file**

Have you ever observed this code smell in the described or a similar form? *

Please choose **only one** of the following:

- Yes
 No

Make a comment on your choice here:

If you have any comments on your observations/experiences with this code smell, use the text area to let us know.

For which implementation techniques you observed this code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '48 [VCSswitchKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

- Feature-Oriented Programming
 C preprocessor
 Antenna preprocessor
 Other:

In which context you observed the described code smell? *

Only answer this question if the following conditions are met:

Answer was 'Yes' at question '48 [VCSswitchKnown]' (Have you ever observed this code smell in the described or a similar form?)

Please choose **all** that apply:

student's implementation

industrial code

open source project

Other:

How do you estimate the severity of this code smell for the aspects, listed below? *

Please choose the appropriate response for each item:

	very problematic	rather problematic	don't know	rather unproblematic	unproblematic
program comprehension	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Maintainability (e.g., error detection/removal)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
evolvability (e.g., add/change code)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Thank you for your participation 😊

Best,

Wolfram & Sandro

05-08-2014 – 08:49

Submit your survey.

Thank you for completing this survey.