

Test Report for Delivery Phase

PMoC
14.4.2003
Version 1.0

T-76.115 Test Report

PMoC

Version	Date	Author	Description
1.0	13.4.2003	Jakobsson	Test report for phase DE

Table of Contents

1. Summary	4
2. Variances	4
3. Comprehensiveness assessment	4
4. Summary of results	4
4.1 Unit tests:	4
4.1.1 GML data structure self-test: Component.java	4
4.1.2 Category load self-test: Category.java	5
4.1.3 Component type load self-test: ComponentType.java	5
4.1.4 Department load self-test: Department.java	5
4.1.5 Connection handling self-test: PropertyInfo.java	5
4.2 Integration tests:	6
4.2.1 Graphical tools	6
4.2.1 Saving and loading through IO.	6
4.3 System tests for symbol editing:	6
4.3.1 STC_A1: Creating a new symbol.	6
4.3.2 STC_A2: Open symbol	7
4.3.3 STC_A3: Import graphics.	7
4.3.4 STC_A4: Save symbol	7
4.3.5 STC_A5 Add graphical object to symbol	7
4.3.6 STC_A6: Change graphical attribute	7
4.3.7 STC_A7: Move graphical object	8
4.3.8 STC_A8: Delete graphical object	8
4.3.9 STC_A16: Add terminal	8
4.3.10 STC_A17: Scroll symbol	8
4.3.11 STC_A18: Zoom symbol	8
4.3.12 STC_A19: Save component type	9
4.3.13 STC_A21: Import component type	9
4.3.14 STC_A25: Attach symbol to type	9
4.3.15 STC_A26: Make free symbol terminal symbol	9
4.3.16 STC_A27: New connection symbol.	9
4.4 System tests for examining net:	10
4.4.1 STC_B1: Open net	10
4.4.2 STC_B2: Scroll net	10
4.4.3 STC_B3: Zoom net	10
4.4.4 STC_B4: Expand/collapse tree representation	10

4.4.5 STC_B5: Select root of tree	10
4.4.6 STC_B6: Select visible element types in tree	11
4.5 System tests for net editing:	11
4.5.1 STC_C1: New component	11
4.5.2 STC_C2: Save net	11
4.5.3 STC_C3: Add graphical object to net	11
4.5.4 STC_C4: Move graphical object	11
4.5.5 STC_C5: Delete graphical object	12
4.5.6 STC_C6: Add component to net	12
4.5.7 STC_C7: Move component symbol	12
4.5.8 STC_C8: Delete component	12
4.5.9 STC_C9: Change component property or attribute value	12
4.5.10 STC_C10: Add connection	12
4.5.11 STC_C11: Delete connection.	13
4.5.12 STC_C12: Change connection start/end points	13
4.5.13 STC_C13: Change connection intermediate points.	13
4.5.14 STC_C17: Save component	13
4.6 System tests for browser editing:	14
4.6.1 STC_D1: Create category	14
4.6.2 STC_D2: Change category attribute	14
4.6.3 STC_D3: Delete category	14
4.6.4 STC_D4: Create department	14
4.6.5 STC_D6: Delete department	14
4.4 Summary	15
5. Evaluation	15
6. Summary of activities	15

1. Summary

This document describes the results of the regression tests in the delivery phase. Test cases are built from the user requirements document, the technical specification and the test plan.

During the delivery phase the testing was concentrated on regression tests, and ad-hoc tests for finding bugs in the different graphical tools. As regression tests the unit tests were also run again.

The testing environment is the same as the development environment, described in the project plan.

2. Variances

Testing in the phase went mostly as planned. All the modules of the program were functional at the end of the phase and all of the planned tests could be run, and even a few tests that were not planned were run, as optional functionality was implemented. All priority 1 tests were run. Of priority 2 tests, the symbol editing tests and the corresponding functionality had previously been dropped in agreement with the customer (A9-A15, A22-24).

3. Comprehensiveness assessment

As the project has advanced, the testing has advanced to include system tests on all parts of the program. To cover the different drawing tools and combinations of action, ad-hoc testing was used. Covering all the possible combinations of actions with explicit test cases was not possible. All core functionalities of the user requirement document have gone through testing, so the testing spans the entire system.

4. Summary of results

4.1 Unit tests:

4.1.1 GML data structure self-test: Component.java

Description:

This unit test tests the GML data structure classes by instantiating them a few times, storing data in them (including some attempts at invalid data), reading them back, deleting some of them and checking some automatically updated properties of the data structures.

Test result for GML data structure self-test:

Passed 10/4/2003 by Jan Lönnberg. The test is run by running the main method in Component.java. The output is in the log file. The extra lines starting with "CT"

appear to be the result of debugging code elsewhere in the program, and do not constitute a problem.

4.1.2 Category load self-test: Category.java

Description:

This unit test attempts to load a category into memory. Checks on the correctness of the loaded category are performed manually.

Test result for Category load self-test:

Passed 10/4/2003 by Jan Lönnberg (attributes appear correct when checked by hand). The test is run by running the main method in Category.java. The output is in the log file.

4.1.3 Component type load self-test: ComponentType.java

Description:

This unit test attempts to load a component type into memory. No checks on the correctness of the loaded category are performed.

Test result for Component type load self-test:

Passed 10/4/2003 by Jan Lönnberg (no errors found). The test is run by running the main method in ComponentType.java. The output is in the logfile.

4.1.4 Department load self-test: Department.java

Description:

This unit test attempts to load a department and its contents into memory. Checks on the correctness of the loaded data are performed manually.

Test result for Department load self-test:

Passed 10/4/2003 by Jan Lönnberg (no apparent errors found). The test is run by running the main method in Department.java. The output is in the logfile.

4.1.5 Connection handling self-test: PropertyInfo.java

Description:

This unit test tests the connection handling extensions of the GML classes by loading a component type, creating some terminal properties, connecting the terminals, checking the resulting terminals' types and checking that the connection can be traversed backwards.

Test result for connection handling self-test:

Passed 10/4/2003 by Jan Lönnberg. The test is run by running the main method in PropertyInfo.java. The output is in the log file.

4.2 Integration tests:

4.2.1 Graphical tools

Description: The function of the drawing, selection and zooming tools is tested together to test the adding and removing of the eventlisteners and selectboxes.

Execution: The test is run by ProConfEditor.java.

Test Result:

Passed 9.4 2003 by Tomas Backas.

4.2.1 Saving and loading through IO.

Description: Saving and loading both gml data and svg data through the IO-package.

Execution: The test is run by ProConfEditor.java.

Test Result:

Passed 10.4 2003 by Markus Jakobsson. However, a bug was encountered that corrupts svg files for symbols. As a symbol is opened, the svg file gets manipulated so that defs tags are inserted into the file. The bug does not affect functionality in any way, and the format is corrected when saving the symbol. The bug has been logged in burana with the id number #3754.

4.3 System tests for symbol editing:

4.3.1 STC_A1: Creating a new symbol.

Description:

The test aims to create a new free symbol as described in the requirements document, use case A1. The user requests a new symbol from the tree view. The new symbol is created in the tree. The symbol is created as a free symbol that later can be attached to a component type or set to a terminal. A new symbol editor window displaying an empty symbol is opened. The test integrates both the tree and symboleditor.

Execution:

The test can be run by ProConfEditor.java:

- 1) Rightclick on a category in the tree and choose “create new symbol”

Test Result:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.2 STC_A2: Open symbol

Description and execution:

This test case corresponds to use case A2.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.3 STC_A3: Import graphics.

Description and execution:

This test case corresponds to use case A3.

Test Results:

Failed.

Test run 10.4 2003 by Björn Forss. Import of external SVG works partly. Some external svg-files can be imported to our program. The program seems to have problems especially with svg-files containing scripts. Imported svg seems to somewhat mess up the function of our program. Burana bug 3159.

4.3.4 STC_A4: Save symbol

Description and execution:

This test case corresponds to use case A4.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss. The symbol-files written by the IO-system are correct.

4.3.5 STC_A5 Add graphical object to symbol

Description and execution:

This test case corresponds to use case A5.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.6 STC_A6: Change graphical attribute

Description and execution:

This test case corresponds to use case A6.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.7 STC_A7: Move graphical object

Description and execution:

This test case corresponds to use case A7.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss. Graphical objects can be moved in symbol editor.

4.3.8 STC_A8: Delete graphical object

Description and execution:

This test case corresponds to use case A8.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss. Graphical objects can be deleted one and one and in groups.

4.3.9 STC_A16: Add terminal

Description and execution:

This test case corresponds to use case A16.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.10 STC_A17: Scroll symbol

Description and execution:

This test case corresponds to use case A17. The symbol is scrolled with the pan tool and by pressing the shift key while moving the mouse on the canvas.

Test Results:

Test passed 10.4 2003 by Tomas Backas.

4.3.11 STC_A18: Zoom symbol

Description and execution:

This test case corresponds to use case A18.

Test Results:

Passed 10.4 2003 by Tomas Backas

4.3.12 STC_A19: Save component type

Description and execution:

This test case corresponds to use case A19.

Test Results:

Test passed 10.4 2003. Implemented as part of A21.

4.3.13 STC_A21: Import component type

Description and execution:

This test case corresponds to use case A21.

Test Results:

Test passed 10.4 2003 by Tomas Backas.

4.3.14 STC_A25: Attach symbol to type

Description and execution:

This test case corresponds to use case A25.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss.

4.3.15 STC_A26: Make free symbol terminal symbol

Description and execution:

This test case corresponds to use case A26.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss. The symbols were changed to terminals on both the graphical- and gml-side.

4.3.16 STC_A27: New connection symbol.

Description and execution:

This test case corresponds to use case A27.

Test Results:

Passed.

Test run 10.4 2003 by Björn Forss

4.4 System tests for examining net:

4.4.1 STC_B1: Open net

Description and execution:

This test case corresponds to use case B1.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.4.2 STC_B2: Scroll net

Description and execution:

This test case corresponds to use case B2.

Test Results:

Test passed 10.4 2003 by Tomas Backas. The net is scrolled with the pan tool and by pressing the shift key while moving the mouse on the canvas.

4.4.3 STC_B3: Zoom net

Description and execution:

This test case corresponds to use case B3.

Test Results:

Passed 10.4 2003 by Tomas Backas

4.4.4 STC_B4: Expand/collapse tree representation

Description:

The user expands or collapses a branch of the tree view.

Test result for B4:

Passed 10.4 2003 by Jan Lönnberg.

4.4.5 STC_B5: Select root of tree

Description:

The user changes the root of the tree view.

Test result for B5:

Failed 10.4 2003 by Jan Lönnberg (not implemented).

4.4.6 STC_B6: Select visible element types in tree

Description:

The user chooses which objects should be visible in the tree.

Test result for B6:

Failed 10.4 2003 by Jan Lönnberg (not implemented).

4.5 System tests for net editing:

4.5.1 STC_C1: New component

Description and execution:

This test case corresponds to use case C1.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.2 STC_C2: Save net

Description and execution:

This test case corresponds to use case C2.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.3 STC_C3: Add graphical object to net

Description and execution:

This test case corresponds to use case C3.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.4 STC_C4: Move graphical object

Description and execution:

This test case corresponds to use case C4.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.5 STC_C5: Delete graphical object

Description and execution:

This test case corresponds to use case C5.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.6 STC_C6: Add component to net

Description and execution:

This test case corresponds to use case C6.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.7 STC_C7: Move component symbol

Description and execution:

This test case corresponds to use case C7.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.8 STC_C8: Delete component

Description and execution:

This test case corresponds to use case C8.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.9 STC_C9: Change component property or attribute value

Description and execution:

This test case corresponds to use case C9.

Test Results:

Passed 10.4 2003 by Kenneth Haglund.

4.5.10 STC_C10: Add connection

Description and execution:

This test case corresponds to use case C10.

Test Results:

Passed.

Test run 10.4 2003 by Markus Jakobsson.

4.5.11 STC_C11: Delete connection.

Description and execution:

This test case corresponds to use case C11.

Test Results:

Passed.

Test run 10.4 2003 by Markus Jakobsson.

4.5.12 STC_C12: Change connection start/end points

Description and execution:

This test case corresponds to use case C12.

Test Results:

Failed.

Test run 10.4 2003 by Markus Jakobsson. End points cannot be changed, the connection has to be deleted and drawn again.

4.5.13 STC_C13: Change connection intermediate points.

Description and execution:

This test case corresponds to use case C13.

Test Results:

Passed.

Test run 10.4 2003 by Markus Jakobsson.

4.5.14 STC_C17: Save component

Description and execution:

This test case corresponds to use case C17.

Test Results:

Passed.

Run 10.4 2003 by Tomas Backas.

4.6 System tests for browser editing:

4.6.1 STC_D1: Create category

Description and execution:

This test case corresponds to use case D1.

Test Results:

Passed.

Run 10.4 2003 by Anders Gebala.

4.6.2 STC_D2: Change category attribute

Description and execution:

This test case corresponds to use case D2.

Test Results:

Failed 10.4 2003 by Anders Gebala. Feature was not implemented.

4.6.3 STC_D3: Delete category

Description and execution:

This test case corresponds to use case D3.

Test Results:

Passed 10.4 2003 by Anders Gebala.

4.6.4 STC_D4: Create department

Description and execution:

This test case corresponds to use case D4.

Test Results:

Failed 10.4.2003 by Anders Gebala. As saving departments was not implemented this features was dropped.

4.6.5 STC_D6: Delete department

Description and execution:

This test case corresponds to use case D6.

Test Results:

Failed 10.4.2003 by Anders Gebala. As saving departments was not implemented this features was dropped.

4.4 Summary

Table 1: Summary of results

	Core functionality	All functionality
Tests run	30	48
Tests passed	30	41
Tests failed	0	7

Table 2: Summary of open bugs in Burana

PR#	State	Priority	Severity	Synopsis
3159	Open	semilow	Normal	All external graphics can't be imported and imported svg behaves odd after the user tries to move it.
3417	Open	semilow	Normal	The length of the name-field at the browser doesn't change when renaming item.
3677	Open	semilow	Minor	grid alignment problem
3244	Open	low	Minor	Character set in text objects too limited
3416	Open	low	Minor	Backspace hides more than one character at text input
3754	Open	low	Minor	Extra defs in some svg files
3258	Analyzing	semilow	Normal	The surrounding SelectBox rectangle gets misaligned with the object it surrounds, when mousemoved on a slow computer?

5. Evaluation

Testing showed that the prototype for the proconf system is finished, and the features that have been agreed on with the customer are working. All the core functionality tests (based on core functionality in the user requirements document) tests passed without remarks. The features that have not been successfully tested are either optional or have been dropped in agreement with the customer. The remaining open bugs do not affect core or even desirable functionalities, and were left open due to the tight time budget.

6. Summary of activities

The regression testing in the last phase was conducted after the last features of the system had been implemented and the last high priority bugs had been fixed. Some

bugs where discovered and immediately fixed. The last minute fixing of the project had broken a few parts of the system, but where quickly fixed after beeing recognized. As there were very few new features to test, testing activities took less time than expected.

Table 3: Planned effort - hours to be spent on testing

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
DE	2	2	3	2	3	7	2	20

Table 4: Actual effort - hours spent on testing

Work	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Designing tests	0	0	0	0	0	1	0	10
Executing tests	1	0	1	1	0,5	1	1	5
Writing test documents	0,5	0	0,5	0,5	0,5	3	0,5	5,5
Total	1,5	0	1,5	1,5	1	5	1,5	12