

# Card Suite Testing Toolbox – product launch case study



© 2010 Tieto Corporation

Jurijs Grigorjevs

Tieto,  
[jurijs.grigorjevs@tieto.com](mailto:jurijs.grigorjevs@tieto.com)

**tieto**

# Content

- Why we did it?
- What is in?
- How we implemented it?
- What are the next steps?



# Motivation 1(2)

- We develop payment card system – **Card Suite**
- For internal use we **created different tools** – we spend years on them (all of them)
- We use and **improve** them for daily tasks (keep them up to date)
- We will help **ourselves** by improving usability and documentation
- Other benefits: in such small project we try **Scrum** practices



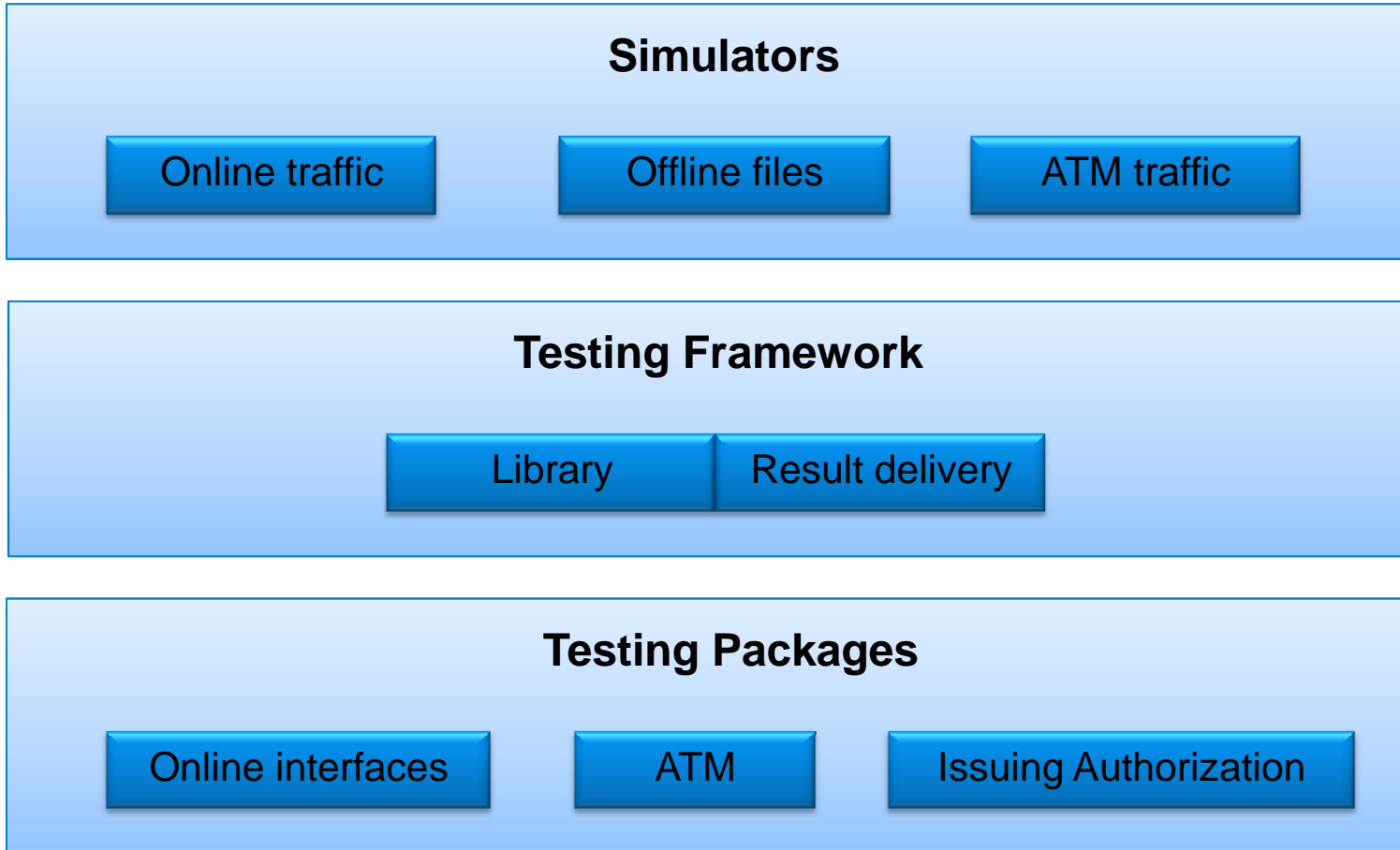
# Motivation 2(2)

## Why customers need our tools?

- Performance testing
- Regression testing during migration projects
- Testing of new functionalities
- Regression testing of specific HW



# Structure of Testing Toolbox



# How we did it?

## Goal

- Improve internal scripts and tools to be used by our customers

## Key points:

- Usability
- Documentation
- Packaging
- Limited budget

## Team:

- Experts without previous knowledge about tools



# Usability

**Only result monitoring has web GUI, other tools are executed from command line.**

- Improve GUI
- Implement --help option for command line tools
- Configuration could be specified in files or in command line during execution
- Provide as minimum parameters as possible
- Support Windows and Linux environments
- Provide examples
- Integrate with existing Card Suite environments



# Documentation

- Use common principles (documentation in code) – Doxygen
- Provide documentation inside packages
- Describe examples and use cases





# Documentation (Doxygen)

TIETO FINANCIAL SOLUTIONS "Card Suite System"

## Simon

1.0.93

External specification

[Main Page](#) [Namespaces](#) [Classes](#)

Card Suite Test Tools : Simon - Developer's Guide

### Configuration

- Environment variable CSHOME must be set to use Simon; CSHOME defines path, where Simon root directory is located.
  - Windows:** CSHOME is set during installation, no action required
  - Linux/Unix:** CSHOME is to be set by user. This can be done in a few ways:
    - Making it a persistent environment variable by appending to *bashrc*, *bash\_profile* or other shell script file
    - Making it a session environment variable by executing following commands:

```
CSHOME=PATH
export CSHOME
```



where PATH is a path, where Simon is installed.
    - Executing Simon with explicit CSHOME definition:

```
CSHOME=PATH simon OPTIONS
```

where PATH is a path, where Simon is installed and OPTIONS are command line options for Simon.
- (OPTIONAL) Add *bin* directory to PATH environment variable to be able to call Simon from any directory.  
**Note:** All examples in following sections assume this step done. If it's not, prepend Simon calls with path to Simon executable.
- Install license  
License should be located in *cfg/license.txt* file. If Simon is installed into Card Suit, append received license to *cfg/license.txt* file.
- Configure  
Edit configuration file *cfg/simon.conf*, though default configuration is acceptable to start with.

Now you can proceed to the [Usage](#).

©2010 Tieto. All rights reserved.

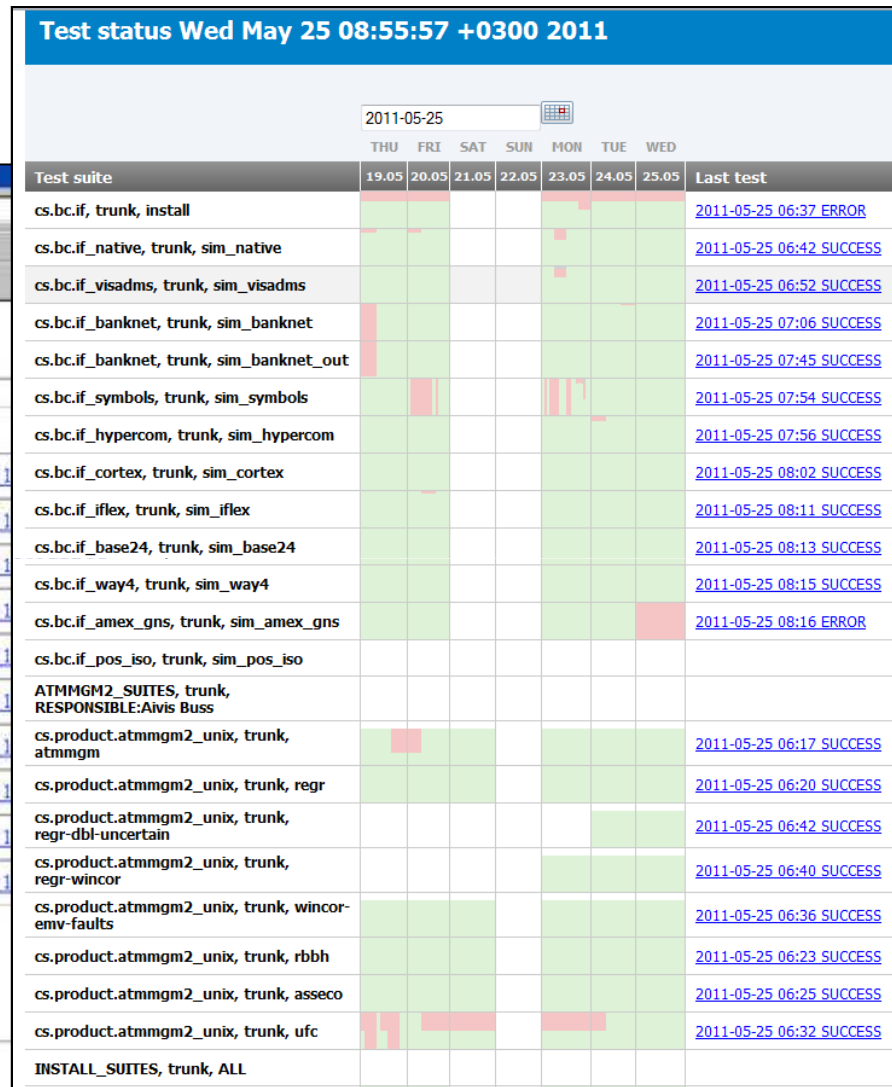
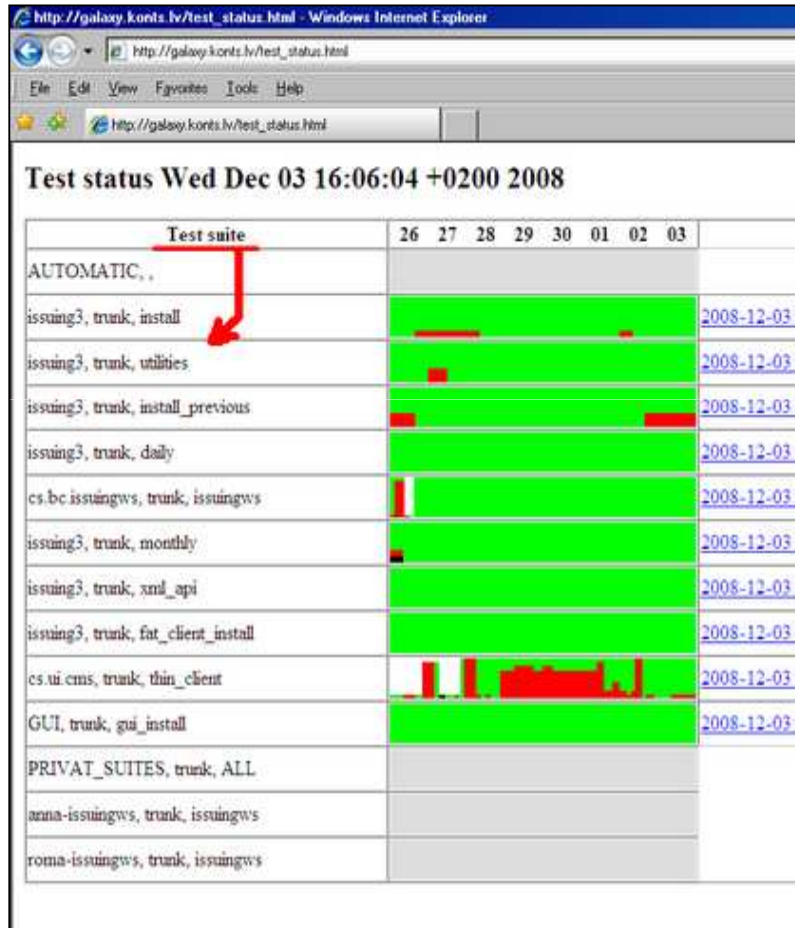


# Packaging

- Create packages - testing packages should be the same as Card Suite components (configuration management, dependence on platforms and 3<sup>rd</sup> party applications)
- Support installation on Linux and Windows
- One step installation
- Licensing (do not licence each protocol, but all together)
- Provide everything what We have!



# Improvements



# Improvements

2011-05-23 Test suite: cs.bc.if\_visadms, trunk, sim\_visadms  
 Tested on: autotest@galaxy.kontis.lv

◀ Back to test status

	Started	Finished	Elapsed	Passed	Failed	Error	Message	log	out	err
sim_visadms	06:54:07	07:02:31	00:08:24	38	0	0	VISADMS SIMON test suite completed			98
sim_visadms	12:23:40	12:30:39	00:06:59	26	9	3	VISADMS SIMON test suite completed	12	7	
▶ outgoing	12:23:40	12:23:47	00:00:07		1		out.1100_20_6011_refund_rec - diff of response to ROUTING_SWITCH failed!			1
▶ outgoing	12:23:47	12:23:54	00:00:07		1		out.1100_26_6010_pay_credit_rec - diff of response to ROUTING_SWITCH failed!			1
▶ outgoing	12:23:55	12:24:22	00:00:27	1			out.1100_21_6010_pay_credit_rec - successfully processed			1
▶ outgoing	12:24:24	12:24:32	00:00:08		1		out.1100_00_AFT_rec - diff of response to ROUTING_SWITCH failed!			1
▶ outgoing	12:24:32	12:25:03	00:00:31	1			out.1100_26_OC_rec - successfully processed			1
▶ incoming	12:25:03	12:25:11	00:00:08	1			0100_00_status.cor - successfully processed			1
▼ incoming	12:25:11	12:25:18	00:00:07		1		0100_00_5968_59_rec.cor - parse ROUTING_SWITCH log failed!			1
shell	12:25:11	12:25:11	00:00:00	1			prepare message			
shell	12:25:11	12:25:13	00:00:02	1			run Simon			81
shell	12:25:18	12:25:18	00:00:00		1		/home/autotest/te_trm/test/bs_if/test_case/processing/scripts/parse_rsw.sh /home/autotest/RTPS/log/FIELDLOG_110523.log "FLD_011" 427112 > /home/autotest/te_trm/test/bs_if/test_case/processing/work/VISADMS /req_to_rsw.tmp			1
shell	12:25:18	12:25:18	00:00:00	1			Log - [loc_110523.log,0100_00_5968_59_rec.cor].			2
shell	12:25:18	12:25:18	00:00:00	1			Log - [symbols_SYMBOLS_110523.log,0100_00_5968_59_rec.cor].			8
shell	12:25:18	12:25:18	00:00:00	1			Log - [task_drv_110523.log,0100_00_5968_59_rec.cor].			6
shell	12:25:18	12:25:18	00:00:00	1			Log - [tcp_110523.log,0100_00_5968_59_rec.cor].			2
▶ outgoing	12:25:18	12:25:46	00:00:28	1			out.1100_00_5968_4_rec - successfully processed			1

Automated Test Report - Windows Internet Explorer

http://galaxy.kontis.lv/log/2008-12-03/suite

No	Started	Finished	Elapsed	Test Case	Result	Message	log	out	err
1	15:19:23	15:20:23	00:01:00	1_build_server_install	PASS	1_build_server_install.sh			
2	15:20:25	15:20:26	00:00:01	2_create_schema	PASS	2_create_schema.sh			63
3	15:20:28	15:22:49	00:02:21	3_create_data_structures	PASS	3_create_data_structures.sh			
4	15:22:51	15:26:56	00:04:05	5_create_plsql	PASS	5_create_plsql.sh			
5	15:26:59	15:27:00	00:00:01	4_run_upgrade_scripts	FAIL	Upgrade contains serious errors		150	1
6	15:27:02	15:28:22	00:01:20	find_diffs	FAIL	Some structures missing from upgrade or repository		2	53

Wed Dec 03 15:28:22 +0200 2008



# Next steps

## Maintenance

- Packages should be updated in parallel with appropriate protocol interfaces
- Bug fixing
- New features and protocols



# Conclusion

Customers require tools for different testing purposes!

- Make them happy and provide such tools.
- Be happy with improved functionality and usability.

**Do You have such tools for your customer?**



# Jurijs Grigorjevs

Integration testing unit manager  
Tieto Latvia

