

Software Testing and QA- ARC middleware

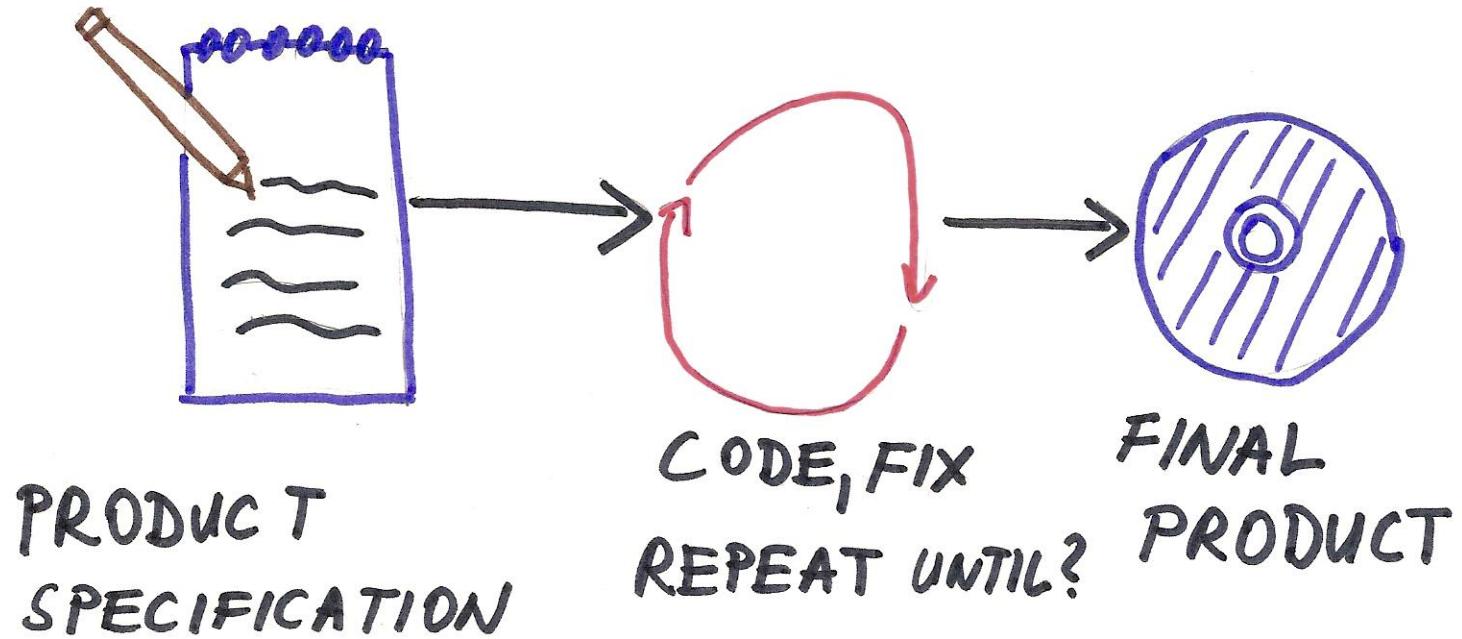


Jozef Černák, jcernak@upjs.sk
Marek Kočan, Martin Savko
P. J. Safarik University in Kosice
Slovak Republic

The software development process

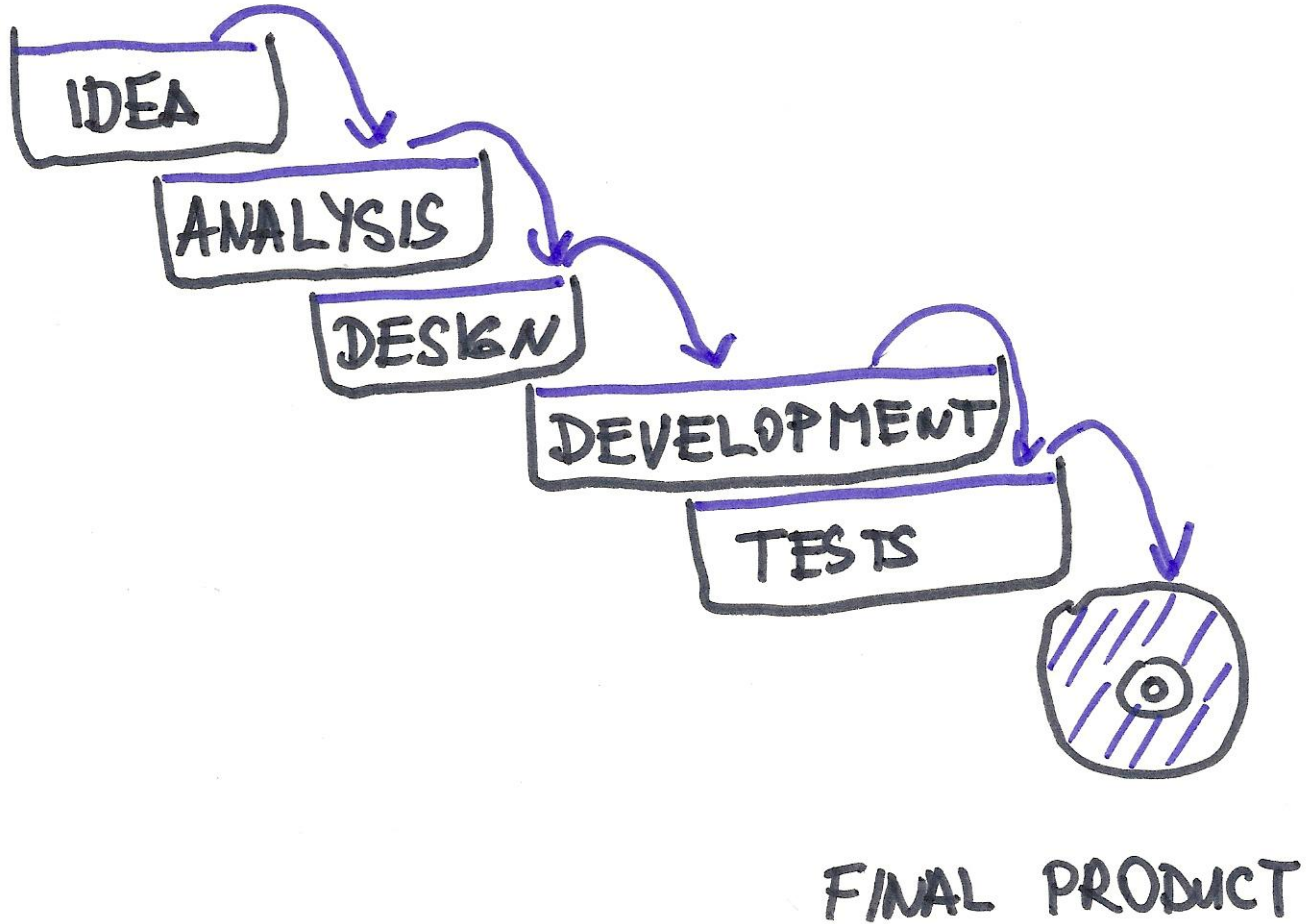
- Basic models
 - **Code and Fix model**
 - **Waterfall model**
 - Spiral model
- Complex models
 - **Combination of basic models**

Code-and-Fix model

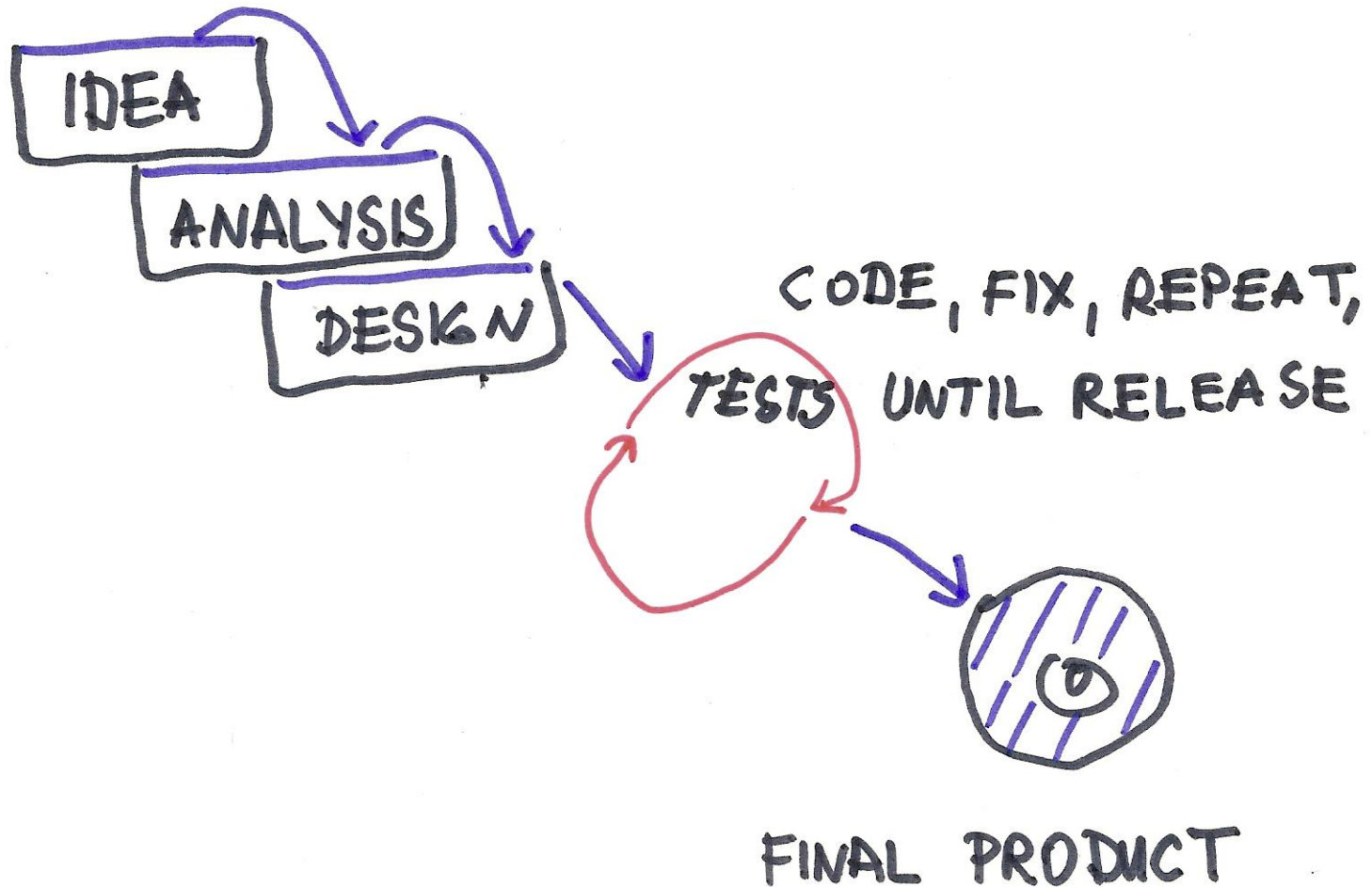


nordugrid middleware –predecessor of ARC (2001-2006)

Waterfall model



Complex model



Testing

- Each stage of development process could be tested
- Ideas, design by model analyzer for example Alloy project <http://alloy.mit.edu/>
- Development
 - unit tests,
 - build tests,
 - functional tests, performance tests, interoperability tests
 - usability tests

Tests in KnowARC

- Testing started at coding stage
- Pieces of software -Unit tests (small number of tests)
- Building binaries from sources was tested for each revision
- Functional tests were performed for any night builds
- Performance tests were performed only two two times during 3.5 years
- Usability test (one at the end of the project)
- Outcome of the Design stage was not be systematically tested



These pages are dedicated to tests of features of ARC1 "nightlies" and subsequent ARC1 subversion ([trunk](#)) revisions. To see the test results choose from the menu above.

Nightlies testing (test metrics)

Services: [AREX](#) [SECURITY](#) [STORAGE](#) [CHARON](#) [HOPI](#) [ECHO](#) [Registration to ISIS](#)

ARC* clients: Jobs submitted to [AREX](#) [ARC0](#) [CREAM CE](#)

Other: Files transfers- [ARCDATA](#) [ARC1 BROKER](#) [ARCRESUB](#) [ARCMIGRATE](#) [ISISTEST](#)

Revision builds

The table below presents build status information of revisions of nordugrid subversion branches on the recent releases of dominant Linux distributions. Each cell contains number of the most recent revision that was already built for respective branch/platform pair. The color of the number (green for success, red for failure and violet if the builds are not performed for specific platform/branch combination) indicates build status. Upon clicking on the numbers one can get to the page with more detailed presentation of results for specific build and also earlier builds.

	Fedora 11, 64bit	Fedora 11, 32bit	CentOS 5, 32bit	Debian 5, 32bit	Ubuntu 8.04, 32bit	SuSE 11.1, 32bit	Gentoo, 64bit
arc1/trunk	18014	18014	16880	17687	18014	16863	18014
arc0/trunk	18032	17441	16554	17377	17496	15641	N/A
arc0/branches/v_0_8	17741	17741	15515	16352	17741	15638	N/A
arc1/branches/components_for_the_0.8	14369	14369	14369	14369	14369	14369	N/A

Archive of past test results



Builds and automated tests of ARC development track

KnowARC

[HOME](#)

(09/08/09)

Revision #	autogen	configure	make	check
14081	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14080	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

(09/07/09)

Revision #	autogen	configure	make	check
14079	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14078	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14077	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14076	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14075	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14074	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

(09/04/09)

Revision #	autogen	configure	make	check
14069	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14068	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Builds and automated tests of ARC development track

KnowARC

[HOME](#) [REVISIONS](#) [BUILDS](#) [AREX](#) [SECURITY](#) [STORAGE](#) [ECHO](#) [HOPI](#) [CHARON](#)

(09/08/09)

Revision #	BUILD	AREX	SECURITY	STORAGE	HOPI	ECHO	CHARON
14081	✓	N/A	N/A	N/A	N/A	N/A	N/A
14080	✓	N/A	N/A	N/A	N/A	N/A	N/A

(09/07/09)

Revision #	BUILD	AREX	SECURITY	STORAGE	HOPI	ECHO	CHARON
14079	✓	N/A	N/A	N/A	N/A	N/A	N/A
14078	✓	N/A	N/A	N/A	N/A	N/A	N/A
14077	✓	N/A	N/A	N/A	N/A	N/A	N/A
14076	✓	N/A	N/A	N/A	N/A	N/A	N/A
14075	✓	N/A	N/A	N/A	N/A	N/A	N/A
14074	✓	N/A	N/A	N/A	N/A	N/A	N/A

(09/04/09)

Revision #	BUILD	AREX	SECURITY	STORAGE	HOPI	ECHO	CHARON
14069	✓	N/A	N/A	N/A	N/A	N/A	N/A
14068	✓	N/A	N/A	N/A	N/A	N/A	N/A



HOME

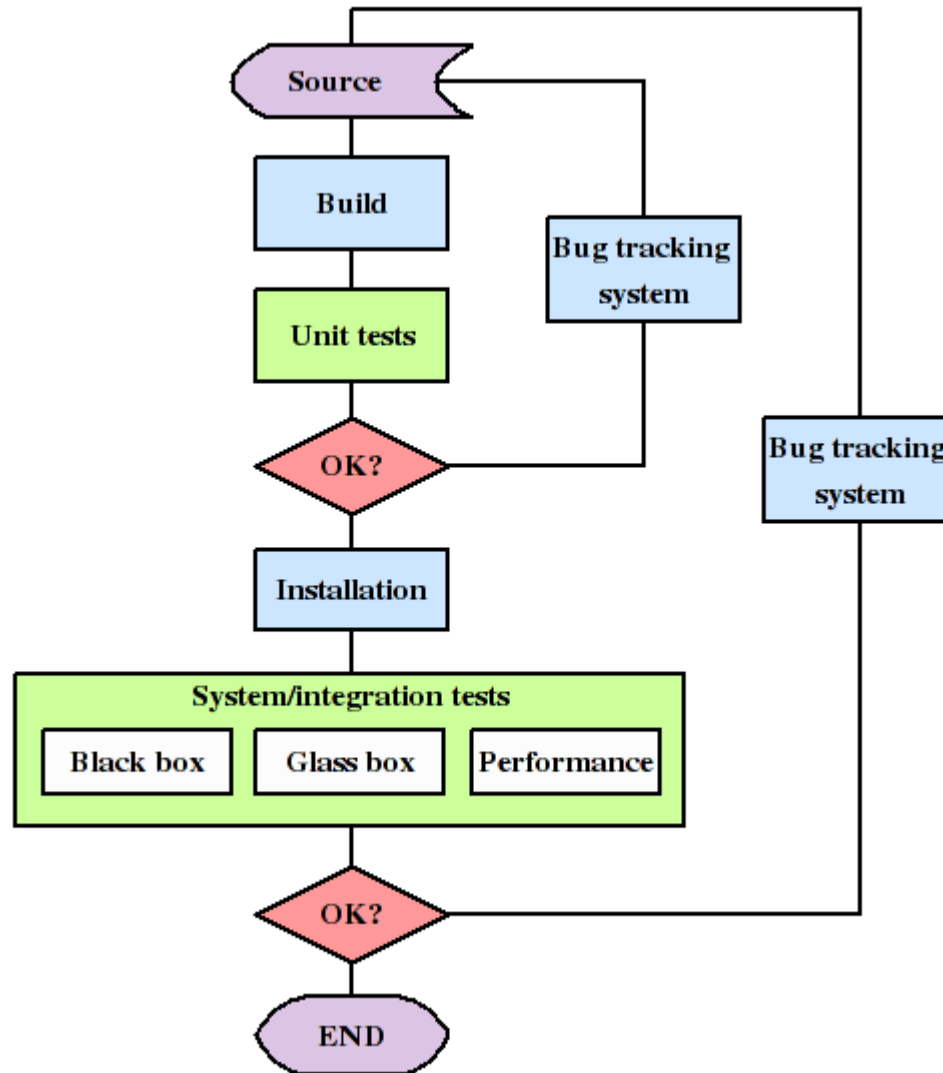
Test results as a ratio between successful and total tests

	OpenSusesource rpm	Fedorasource rpm	Redhatsource rpm	Ubuntusource rpm
2009-02-28	0.0% 0.0%	97.9% 36.6%	97.2% 54.2%	0.0% 36.6%
2009-02-27	0.0% 0.0%	87.3% 36.6%	85.9% 76.1%	0.0% 36.6%
2009-02-26	0.0% 0.0%	76.1% 84.5%	81.0% 92.3%	0.0% 36.6%
2009-02-25	0.0% 0.0%	87.3% 81.7%	76.1% 76.1%	0.0% 36.6%
2009-02-24	0.0% 0.0%	76.1% 87.3%	78.9% 83.8%	0.0% 36.6%
2009-02-23	0.0% 0.0%	81.7% 97.9%	67.6% 96.5%	0.0% 36.6%
2009-02-22	0.0% 0.0%	64.8% 60.6%	59.9% 62.0%	0.0% 19.7%
2009-02-21	0.0% 0.0%	61.3% 59.2%	62.0% 62.7%	0.0% 19.7%
2009-02-20	0.0% 0.0%	31.7% 29.6%	28.9% 27.5%	35.2% 19.7%
2009-02-19	0.0% 0.0%	28.2% 32.4%	28.2% 29.6%	39.4% 19.7%
2009-02-18	0.0% 0.0%	59.2% 53.5%	52.8% 49.3%	62.7% 19.7%
2009-02-17	0.0% 0.0%	53.5% 52.1%	52.1% 57.7%	62.7% 19.7%
2009-02-16	0.0% 0.0%	55.6% 52.8%	52.1% 52.1%	62.7% 19.7%
2009-02-15	0.0% 0.0%	52.8% 7.7%	47.9% 53.5%	63.4% 19.7%
2009-02-14	0.0% 0.0%	57.0% 52.1%	53.5% 53.5%	62.7% 19.7%

QA

- We concentrated to set up procedures for two main outcomes of the project:
 - software,
 - Deliverables.
- We have defined the proces for Relase management http://wiki.nordugrid.org/index.php/Release_management
- Testing (testing plan, metric).
- Analyses of bug status.
- We could not implement the best practices of QA due to limited number of project participants many of them had many roles in the project.

Testing and bugfixing processes



What have we learned?

- The most significant bugs were found during performance tests (memory leak, scalability issues).
- The new features of new releases were not right defined in some cases.
- The functional tests on bigger infrastructure (operational tests) were not realised. Pilot Grid System (infrastructure of several servers 10) does not fulfil initial expectations.
- Testing in permanent stress due to accumulation of delays in previous activities (bug fixing of critical bugs).
- Small numbers of beta-testers
- We were looking for other frameworks, for example internal review of ETIC.
- Our test framework is capable to detect many bugs for many platforms Linux, MAC OS an Windows.

EMI TESTING

- Will all project use the same testing tools?
- What is procedure to propose test cases?
- Interoperability tests?
- Usability tests?
- Performance tests?