THE LINUX FOUNDATION

LSB INFRASTRUCTURE PROGRAM

ISPRAS Report

March 27, 2008I

Abstract

This document presents top-level report of ISP RAS results for the period Feb 28, 2008 - March 27, 2008

Revision History

Date	Version	Description
2007-03-27	0.1	Initial version

Contents

ISPRAS Activities Overview

The current ISPRAS team scope includes the following areas:

1. Development and maturing **LSB informational systems**:
   a. The main **LSB Database** (both schema and data) - 71 tables with over 25 million records in total.
   b. Over 40 **Database Scripts** (including libtodb2 tool fro importing libraries).
   c. Linux developer portal - **LSB Navigator** (http://linux-foundation.org/navigator/).
   d. **LSB Certification System** (http://linux-foundation.org/lsb-cert/).
2. Development and maturing of **automated test *execution* and result analysis frameworks***:*
   a. LSB **ATK Manager** for testing and analyzing *applications*.
   b. LSB **DTK Manager** for testing and analyzing *distributions*.
3. Development and maturing of **automated test *development* frameworks** for different cost & value grades:
   a. **UniTESK** for deep testing.
   b. **T2C** for normal testing.
   c. **Azov** for shallow testing.
4. Development of **new tests** for LSB interfaces in various quality grades:
   a. Deep tests.
   b. Normal tests.
   c. Shallow tests.
5. **Analytical and community collaboration** tasks:
   a. When developing tests we inspect both specifications and Linux implementations (both manually and by tests) - we analyze, additionally annotate and publish all found issues at linuxtesting.org (http://linuxtesting.org/results/impl_reports and http://linuxtesting.org/results/std_reports) and then cooperate with corresponding authors to make specifications and upstream components finally get fixed.
   b. We are actively involved in discussing / problem solving of LSB / Linux issues of various kinds in mailing lists / irc.
   c. We promote and advance LSB and Linux/Open Source in general through participating in Russian government official working groups (to define federal policies), speaking at conferences, publishing papers, organizing events.

Achievements for the period

Detailed list of completed tasks can be extracted from the weekly reports at the http://www.linux-foundation.org/en/ENG:Status page (see ISPRAS section at the bottom). The sections below highlight only most important / higher level things.

LSB Database

1. DB Infrastructure (schema, scripts, general issues):
    a. **2** new tables added to schema and populated (LibraryAttribute and IntStd).
    b. Changed a way of storing community data - files are now more than 2 times smaller, database setup procedure is more than 2 times faster.
2. LSB Database Data
    c. Collected data for **110** new applications (including **80** proprietary) and **2** new distributions.
    d. Completed libGLU inclusion to LSB
    e. Updates for **1500** C++ symbols - access types, static and virtual functions markers.
    f. Collected missing data for ~**50** header files from **3** libraries.
3. LSB Bugzilla Activity:
    a. **11** new bugs identified (mostly in legacy DB data).
    b. Patches for **10** bugs created.
    c. Investigations conducted and suggestions submitted for **8** bugs.

LSB Navigator

1. New 'Solution Support' section added to represent all information (that can be obtained from the database) on symbols required during decision making in one table.
2. New page added to enable deprecated symbol usage analysis.
3. **3** new filters added on application analytics pages.
4. **5** new complex consistency checks added in admin mode to help maintaining DB data consistency.

LSB DTK Manager

1. Released LSB DTK Manager 1.5.6 with **9** bugs fixed (both reported externally and identified by ISPRAS at additional QA cycle).
2. Most efforts are now in developing **DTK-2** (with significant internal architecture redesign):
    a. the first version of MANIFEST based architecture developed;
    b. completely redesigned the Web-UI Custom Tests page (more user-friendly version selection; advanced test suite customization);
    c. command-line UI allows running test suites by their status/version;
    d. automatic manifest generating from template.
3. A tool for comparing multiple journals created.

LSB ATK Manager

1. Released LSB ATK Manager 1.3 with command-line UI implemented and recursive processing of RPM/TAR.GZ archives.
2. Eclipse plug-in: developed support for 3 project types (LSB Executable):
    a. "C" Release;
    b. "C" Debug;
    c. "C++" Release.

LSB Certification System

1. **5** new features/improvements:
    a. Reminder script was improved to distinguish unanswered audit threads and also unanswered threads are now highlighted at the admin status page.
    b. Multiple file attachments are now possible.
    c. Progress diagrams for admin mode added.
    d. Clickable names in audit (admin mode).
    e. 'Status' menu as a Cancel button when uploading files via ATK/DTK Manager.
2. **5** bug fixes.
3. **6** minor (cosmetic) improvements.
4. Internal code reorganization performed.

Deep Testing

1. Tests for **56** interfaces were upgraded to the deep level (**47** from io.fstream.fstream, **5** from process.goto, **4** from process.context subsystem).
2. **182** additional requirements were marked up in formal specifications for util.format, util.conversion, locale.textdomain, io.file, socket.socket, pthread.pthread, process.goto, process.pgroup, fs.fs, fs.meta.meta, util.getopt, util.regex, util.float groups of interfaces.
3. Checks-up for **30** requirements were included in test scenarios.
4. "locale" formal model was developed.
5. Stabilization on IA64 architecture done to prevent tests from crash at this architecture.
6. Accurate quality analysis and coverage measurement conducted.

Normal Tests

T2C Infrastructure / Technology

1. Main functionality for enhanced requirement catalogue and template generation support (except the case of multiple source files) has been implemented. Additionally, special dialogs have been added to facilitate definition and modification of substitutions in the requirements' IDs ("define-blocks").
2. A few problems in the code generator have been fixed including the output of invalid messages that sometimes occurred for uninitiated tests.

Test Development

GTK

1. Most of the tests for gobject library have been reviewed (**314** interfaces, **926** test cases total). Corrections were made.

OpenGL

1. Tests have been developed for **41** more interfaces from libGL: about **1300** test cases added.
2. The stabilization (checking on a variety of platforms) of these new tests as well as of the tests for **75** interfaces written before is now in progress.
3. The image capturing facility used by the tests has been revisited. Some problems have been fixed and now this functionality also works correctly on ppc/ppc64 and s390/s390x architectures.

Now the status of test development for OpenGL is as follows:

Library	Tested interfaces	Number of test cases
libGL	116	2887

X11

1. **51** test cases have been developed so far for **36** interfaces from libX11.
2. To mark up the requirements for these interfaces in the documentation, html version of its **8** chapters has been prepared from various sources.
3. A special "X11 traffic hooking" technique is developed and now is being adopted to develop test scenarios for X11.

Now the status of test development for X11 is as follows:

Library	Tested interfaces	Number of test cases
libX11	36	51

C++

1. About **90** test cases have been developed for **56** interfaces which are mostly methods of exception classes.
2. Special techniques have been adopted to be able to test constructors and destructors of these classes.
3. A few cases of suspicious behavior of std::uncaught_exception() interface have been detected. Further analysis is now performed to decide if there is a problem in the library or in the documentation.

Now the status of test development for libstdcxx is as follows:

Library	Tested interfaces	Number of test cases
libstdcxx	74	198

New Libraries to be Tested (libfreetype, libpng12, libjpeg)

The documentation for **3** libraries (libfreetype, libpng12, libjpeg, 211 untested interfaces total) from LSB Desktop module has been investigated to decide if T2C technology can be used in test development for these libraries. The main concerns here were error-handling facilities used in libpng12 and libjpeg. The analysis of the documentation and the developed sample code showed that T2C tools require no special modification to be applicable for these cases.

Desktop-T2C Test Suite Summary

Now the "Desktop-T2C" test suite developed by ISPRAS (numbers are in total since the beginning) contains nearly **7000** tests for more than **1700** interfaces from the following libraries:

Library	Tested interfaces	Number of test cases
libglib-2.0	828	1938
libgthread-2.0	2	2
libgmodule-2.0	8	28
libatk-2.0	222	574
libfontconfig	160	314
libgdk-pixbuf-2.0	71	373
libgobject-2.0	314	926
libGL	116	2887
libX11	36	51
Total	**1757**	**7093**

Note: C++ tests are in a separate test suite (see corresponding statistics for C++ above).

Shallow Testing

1. Finished framework for shallow-tests integration with T2C backend.
2. Organized nightly build for shallow-tests on **10** machines of **7** different architectures.
3. Stabilized shallow-tests for Qt3 library on all **7** architectures.
4. As a result of Qt3 testing:
    a. **2** classes were withdrawn from LSB;
    b. **1** class was added to LSB;
    c. **140** in-charge interfaces from abstract class were marked as 'untestable';
    d. description of **4** interfaces was admitted as incomplete by Trolltech team.
5. Corrected existing info in LSB DB about Qt4 library, such as parameters' types, returned types, information about constructors and destructors, access level for some interfaces.
6. New library group partitioning was decided and added in Qt4 library.
7. Created source-parser tool and applied it for uploading additional info from Qt4 library (~**9000** interfaces, **2000** types).
8. Prepared infrastructure environment (generator, database, report system) for starting development of shallow-test for Qt4 library.

Community Tasks

1. Lots of messages in the lsb-* lists.
2. Found, analyzed and published **6** confirmed problems in Linux upstream components and specifications (see http://linuxtesting.org/results/impl_reports and http://linuxtesting.org/results/std_reports).
3. Jointly with *Ministry of Telecommunications and IT* developed "*Russian Federal Concept of Using Open Source in Government Organizations*" - this defines and drives open source usage in Russia at government level.
4. Prepared and submitted "*DB-driven Infrastructure for Development of API Standards*" article and talk for "XXI Century High-Technology and Cybernetics". Accepted by program committee.
5. Prepared "*Using Open Source Projects for University Education*" article.