Linux distribution testing and quality assurance: current developments of the EDOS project and open questions

FOSDEM 2007

François Déchelle, EDGE-IT fdechelle@mandriva.com
Stéphane Laurière, EDGE-IT, slauriere@mandriva.com
Paulo Trezentos, Caixa Mágica, Paulo.Trezentos@iscte.pt
Mário Morgado, Caixa Mágica, mario.morgado@caixamagica.pt
Laurent Godard, NUXEO, lgodard@indesko.com
Linux distribution testing

Issues and objectives

- Linux distribution quality assurance is a difficult task
  - complexity (number of packages ~ 10000)
  - in-house QA team or community testing
  - very few automatic testing
- EDOS workpackage 3 goal: to research and experiment solutions for Linux distributions quality assurance
- A framework providing:
  - automatic and manual testing
  - access to tests, test results...
  - real-time view of the testing state of the distribution
EDOS architecture

General architecture

- Upstream src
  - PushRPM
- Package repository nb1
- Package repository nb2
- SPEC file repository
- Test package repository
- Installability tester
- LTP Runner
- Dogtail
- STAF/STAX Runner
- TULIP

- Package metadata
- QA database
- Upstream bug repositories
- RSS feeds

- BuildFarm Manager
  - Upstream tracker
  - ANLA, RpmCheck, ...
  - CEVE
  - Dependencies libraries
  - Problem Generator
  - TestFarm Manager

- EDOS Master Server

- EDOS Portal
  - EDOS Explorer
  - EDOS Visualizer
  - Metrics Dashboard

- EDOS PM
  - Web service / REST API

- Trusted Peers
  - EDOS Dissemination Service
    - iDip
    - KadoP
    - Active XML
    - Azureus

- HTTP or SSH

- i586
- x86-64
- PPC
- Virtual environment

- TULIP server

- Launchos

- Developer
  - Package
  - Tester
  - Observer
Test specification file format

EDOS testing framework

- Separation between
  - test management (running, reporting)
  - test implementation
- Integration of existing test suites
- No consensus on language for writing tests
- Tests are packaged
- Tests have pre-conditions (eg. run as root)
- Tests are run by a harness implemented in shell, Python...

- ... XML schema for test specification
A test specification
EDOS testing framework

<?xml version='1.0' encoding='UTF-8'?>
<test>
  <description
      name="crashme"
      suite="LTP">
    This test runs the LTP crashme test.
  </description>
  <package name="edos-ltp"/>
  <condition>
    <user name="root"/>
  </condition>
  <test-runner>
    <shell script="yes | /usr/lib/ltp/runltp" shortLog="true">
      <arg value="-q"/>
      <arg value="-f /usr/lib/ltp/runtest/crashme"/>
    </shell>
  </test-runner>
</test>
Test runner
EDOS testing framework

- Functions
  - load an XML test specification
  - verify package and conditions
  - run the test
  - report the platform description and the test result

- Written in Python
- Distribution agnostic (package format, platform access)

- Test reporting
  - use test reporting service
  - support parallel test reporters (stdout, XML...)
TULIP framework
EDOS testing framework

- **Testing Upgrades of Linux Images Program**
- Drive upgrade tests of various Linux distributions to ensure both fine grained QA at the package level and testing of the standard update mechanism
- Upgrading a distribution is a risky business, especially when using “unstable” branches
- Both a packaging and a tool (apt, urpmi...) problem
- Need also testing tools to reasonably check that things actually work for end users once upgraded
- TULIP uses QEMU and incremental images
### TULIP session report

**EDOS testing framework**

---

#### TULIP Project Activity - Session report

<table>
<thead>
<tr>
<th>Session Reference</th>
<th>Date</th>
<th>Project Name</th>
<th>Project Id</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>195</td>
<td>28-july</td>
<td>debian</td>
<td>11</td>
<td>daily</td>
</tr>
</tbody>
</table>

#### Detail of installed packages

<table>
<thead>
<tr>
<th>detail_id</th>
<th>log_id</th>
<th>delta_rank</th>
<th>Package Name</th>
<th>Status</th>
<th>version</th>
</tr>
</thead>
<tbody>
<tr>
<td>505</td>
<td>195</td>
<td>1</td>
<td>hicolor-icon-theme</td>
<td>ok</td>
<td>0.8.4</td>
</tr>
<tr>
<td>506</td>
<td>195</td>
<td>1</td>
<td>modutils</td>
<td>ok</td>
<td>2.4.27.0-6</td>
</tr>
<tr>
<td>507</td>
<td>195</td>
<td>1</td>
<td>grub</td>
<td>ok</td>
<td>0.97-11</td>
</tr>
<tr>
<td>508</td>
<td>195</td>
<td>1</td>
<td>nano</td>
<td>ok</td>
<td>1.3.11-3</td>
</tr>
<tr>
<td>509</td>
<td>195</td>
<td>1</td>
<td>mount</td>
<td>ok</td>
<td>2.12r-10</td>
</tr>
<tr>
<td>510</td>
<td>195</td>
<td>1</td>
<td>libsysfs2</td>
<td>ok</td>
<td>2.0.0-7</td>
</tr>
<tr>
<td>511</td>
<td>195</td>
<td>1</td>
<td>libc6-dev</td>
<td>ok</td>
<td>2.3.6-15</td>
</tr>
<tr>
<td>512</td>
<td>195</td>
<td>1</td>
<td>libc6-amd64</td>
<td>ok</td>
<td>2.3.6-15</td>
</tr>
<tr>
<td>513</td>
<td>195</td>
<td>1</td>
<td>gnome-mime-data</td>
<td>ok</td>
<td>2.4.2-2</td>
</tr>
<tr>
<td>514</td>
<td>195</td>
<td>1</td>
<td>bsutils</td>
<td>ok</td>
<td>1:2.12r-10</td>
</tr>
<tr>
<td>515</td>
<td>195</td>
<td>1</td>
<td>libc6</td>
<td>ok</td>
<td>2.3.6-15</td>
</tr>
<tr>
<td>516</td>
<td>195</td>
<td>1</td>
<td>gzip</td>
<td>ok</td>
<td>1.3.5-14</td>
</tr>
<tr>
<td>517</td>
<td>195</td>
<td>1</td>
<td>util-linux</td>
<td>ok</td>
<td>2.12r-10</td>
</tr>
<tr>
<td>518</td>
<td>195</td>
<td>1</td>
<td>locales</td>
<td>ok</td>
<td>2.3.6-15</td>
</tr>
<tr>
<td>519</td>
<td>196</td>
<td>2</td>
<td>debconf</td>
<td>ok</td>
<td>1.5.2</td>
</tr>
<tr>
<td>520</td>
<td>196</td>
<td>2</td>
<td>hicolor-icon-theme</td>
<td>ok</td>
<td>0.8.4</td>
</tr>
<tr>
<td>521</td>
<td>196</td>
<td>2</td>
<td>modutils</td>
<td>ok</td>
<td>2.4.27.0-6</td>
</tr>
<tr>
<td>522</td>
<td>196</td>
<td>2</td>
<td>grub</td>
<td>ok</td>
<td>0.97-11</td>
</tr>
<tr>
<td>523</td>
<td>196</td>
<td>2</td>
<td>libgnomeui-common</td>
<td>ok</td>
<td>2.14.1-2</td>
</tr>
<tr>
<td>524</td>
<td>196</td>
<td>2</td>
<td>console-data</td>
<td>ok</td>
<td>20060900</td>
</tr>
</tbody>
</table>

#### Images Delta Upgrades results

- Successful: 14
- Failure: 0

---

FOSDEM

**Bruxelles, 24 February 2007**

© EDOS Consortium 2004 – 2007
QA portal
EDOS testing framework

- Portal for testers, distribution developers...
- Provide access to test data integrated with other WP data
- Based on Xwiki, a Java implemented wiki
- Services are implemented using Xwiki Groovy scripting facility
  - very fast prototyping
  - very easy update of scripts
Interactive test agent
EDOS testing framework

- Use to guide the user during a manual test
  - interactive applications
  - operations that cannot be automated (e.g., USB key insertion)
- XML specification of manual tests
  - user instruction
  - expected result including images
  - test sequencing
- Implemented using and interfaced with the test runner
<tcdl>
  <languages>
    <language id="en"/>
    <language id="it"/>
  </languages>
  <name>ins_frm03</name>
  <application>OpenOffice.org</application>
  <platform>Solaris - Linux - Windows</platform>
  <description>
    <text>
      Verify OpenOffice.org can insert a frame into a text document.
    </text>
    <text lang="it">
      Verificare che OpenOffice.org sia in grado di inserire un frame in undocumento di testo.
    </text>
  </description>
  <steps>
    <step num="1">
      <instruction>
        <text>
          From the menu select File > Open, the Open File Dialog will be opened.
        </text>
      </instruction>
    </step>
  </steps>
</tcdl>
QA metrics

- Metrics are commonly used for measuring QA level
- What metrics for Linux distributions?
- Code coverage?
  - what is the coverage unit? line of code... -> package?
  - coverage measurement requires instrumentation of what is to be measured... how to instrument a Linux distribution?
- Other metrics to evaluate the testing level of a Linux distribution?
Automatic testing and Linux distributions? Would allow to re-run tests after distribution builds. But may require huge processing power (cf. OO.o full test suite). Automatic testing of some components, namely GUI, is a work in progress. works when designed for one environment (OO.o, Mozilla) generic UI testing (dogtail, LDTP).
Thank you!