Tizen Platform Development Process and Infrastructure

Jong Woo, Chae
Software Center, Samsung Elec.
Tizen Development Process is perceived by users and developers as a black box where magic is happening

- Idea of this presentation is to give more information on how it built, what principles it uses and how people can replicate and customize it for own needs
- We will help you
  - to contribute Tizen
  - to define your own Tizen profile
  - to develop Tizen products
Contents

Part I. Tizen Platform Development Process

Part 2. Customizing Development Process

Q & A
Part I. Tizen Platform Development Process
Brief Tizen Platform Development Process

Code Review → Incremental Build → Continuous Integration

Automated Smoke Test

*) OBS: Open Build Service by OpenSUSE
**) GBS: Git Build System by Tizen

TIZEN DEVELOPERS CONFERENCE 2015 SHENZHEN
Code Review with Tizen Governance

Gerrit projects are grouped and ruled by "Domain"

Project Roles

https://wiki.tizen.org/wiki/Tizen_Governance

Architects
Maintainers
Reviewers
Developers

Domains

Release Engineers
QA Engineers

Write Code
Local Verification (Build, Test)
Review Request
Review
Approve Code Change
Integrate Code Change
Submit to OBS

Developer
Reviewer
Integrator/Maintainer

Gerrit projects are grouped and ruled by “Domain”
Incremental Build

Each gerrit project of Tizen composes a RPM package

When a package changed, OBS triggers build using its dependent packages

- No need to compile other source codes – reduce build time to verify a change
- Packages are stored in and updated to Live repository of OBS

```
<table>
<thead>
<tr>
<th>Code Change</th>
<th>Pkg A</th>
<th>Pkg B</th>
<th>Pkg C</th>
<th>Pkg D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>1.1</td>
<td>2.0</td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>1.2.1</td>
<td>2.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1.8</td>
<td>2.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.2.1</td>
<td>2.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Developers can verify a build locally.

Live Repository

http://download.tizen.org/live/Tizen/Mobile/
Snapshot

Version Control for the daily development

• A certain state of all packages after group of changes are merged
• Snapshot ID - <Tizen Project>-<date>-<index>

http://download.tizen.org/snapshots/tizen/mobile/
Repository Difference and Changelogs

Go back...

Difference between mobile_20150901.2 and mobile_20150902.1

Highlights

Added Packages: 0
Removed Packages: 0
Modified packages: 52
Packages with Rebuilds: 263

Added Packages

Removed Packages

Modified Packages

<table>
<thead>
<tr>
<th>Package</th>
<th>Old Version</th>
<th>New Version</th>
<th>GitPath</th>
<th>Old CommitId</th>
<th>New CommitId</th>
</tr>
</thead>
<tbody>
<tr>
<td>ptk-modules-minfi</td>
<td>0.1.0.51.2</td>
<td>0.1.0.51.1</td>
<td>platform/prop/ptk</td>
<td>6200b6d82418bf3f6ae5c5888c9a2ba279e9c6d2</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>pulsaudio-module-filter</td>
<td>5.0-30.1</td>
<td>5.0-30.1</td>
<td>platform/patch/pulsaudio</td>
<td>3250f726963a037f993448656806868368c3ed1f5</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>pulsaudio-module-rt</td>
<td>5.0-30.1</td>
<td>5.0-30.1</td>
<td>platform/patch/pulsaudio</td>
<td>3250f726963a037f993448656806868368c3ed1f5</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>libmm-utility-dev</td>
<td>0.13-14.3</td>
<td>0.14-15.1</td>
<td>platform/patch/libmm-utility</td>
<td>851499214162909047944f3c8f819b3f05</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>alarm-server</td>
<td>0.4.178-19.6</td>
<td>0.4.178-20.1</td>
<td>platform/patch/alarm manager</td>
<td>51a3d2b6193a952efb08919a9c6078e6d4b18f</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>pulsaudio-module-rop</td>
<td>5.0-29.1</td>
<td>5.0-30.1</td>
<td>platform/patch/pulsaudio</td>
<td>3250f726963a037f993448656806868368c3ed1f5</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>call-media-camera-dev</td>
<td>0.2.3-4.6</td>
<td>0.2.5-5.1</td>
<td>platform/patch/camera</td>
<td>5467819b8f84a8f982e4e7c2666ce07da6666f87</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>tebck-sysmlinks-dlp</td>
<td>0.5.1-3.1</td>
<td>0.6.0-4.1</td>
<td>platform/patch/sysmlinks</td>
<td>3ca87343e61f8f956cb9646e46a5c6e40</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>libmm-utility-tool</td>
<td>0.13-14.3</td>
<td>0.14-15.1</td>
<td>platform/patch/libmm-utility</td>
<td>851499214162909047944f3c8f819b3f05</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>tebck-sysmlinks-sysklog</td>
<td>0.5.1-3.1</td>
<td>0.6.0-4.1</td>
<td>platform/patch/sysmlinks</td>
<td>3ca87343e61f8f956cb9646e46a5c6e40</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>tpt-backend</td>
<td>1.7-33.1</td>
<td>1.8-34.1</td>
<td>platform/patch/tpt</td>
<td>4df740104dd7b4f1c7f3e0b0953d5190d8b4f</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>pulsaudio-realtime-scheduling</td>
<td>5.0-29.1</td>
<td>5.0-30.1</td>
<td>platform/patch/pulsaudio</td>
<td>3250f726963a037f993448656806868368c3ed1f5</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
<tr>
<td>pulsaudio-dac</td>
<td>5.0-29.1</td>
<td>5.0-30.1</td>
<td>platform/patch/pulsaudio</td>
<td>3250f726963a037f993448656806868368c3ed1f5</td>
<td>6107bad19d9a657b5a2362db8389c5f9404</td>
</tr>
</tbody>
</table>

TIZEN DEVELOPERS CONFERENCE 2015 SHENZHEN

9/25
Legacy Release Process

Staging build to verify submitted changes are acceptable

- No build break is accepted by this process
- Problem: changes are built in one place
  - hard to identify which relationship among changes
  - hard to identify which changes cause problems
  - smoke test is triggered after release engineer

- Code Review
- Incremental Build
- Automated Smoke Test
- Continuous Integration

Staging Build

Accept when build is successful

Tizen:2.2:Mobile:build

OBS

Accept Submit

Integration Build

Tizen:2.2:Mobile

Smoke Test

Snapshot

Runtime error detected

Code Review

Integrator

Code Change

Code Change

Code Change

Code Change

Release Engineer
Pre-release Process

Changes are verified as a group
Smoke test is triggered before Release Engineer

Release Engineer accepts the changes when:
- No build break, All binary images are created, No test regression
Smoke Test (Build Verification Test)

When new snapshots are created, smoke test is triggered

- Reboot the device
- Download and flash new binary image
- Launch smoke test cases (verifies major daemons are alive)
  - Can be extended to launch any TCs including TCT (Tizen Compliance Test)

TAV (Tizen Automated Verifier) is designed for no modifications in binary images and fully automated process (will be released soon)

Create Snapshot → Jenkins → Smoke Test → TAV

- Tizen 3.0 Mobile
- 1.1 1.2.1 2.2 3.1
- 1.1 1.2.1 2.1 3.1
- Power Cutter
- Flashing Interface
- Launch Smoke Test

Code Review → Incremental Build → Automated Smoke Test → Continuous Integration
Snapshot and Smoke Test

Easy to find the origin of the error

- In Tizen platform, we have \( \approx 900 \) of packages

1. Error is discovered
2. Same error is repeating until #3
3. Survey changed packages only

- Snapshot Diff with #2
  - Added packages: ...
  - Removed packages: ...
  - Modified packages: ...
  - Packages with rebuild: ...

Release Engineer

- Code Review
- Incremental Build
- Automated Smoke Test
- Continuous Integration
Local Verification (GBS)

GBS (Git Build System) supports you:
- to build one or multiple git repositories with dependent packages
- incremental build using local repo
- to create binary images (using MIC, MIC Image Creator)
- to submit changes to OBS to trigger build
Part I. Tizen Platform Development Process

Tizen Platform Development Process

Developer
1. Write Code
2. Local Verification (GBS) (Build, Test)
3. Review Request
   - Review Comment
   - Not OK
   - OK → Approve Code Change

Reviewer
- Review
- Not OK
- OK

Integrator/Maintainer
1. Submit to OBS
2. Integrate Code Change

Release Engineer
1. Pre-release Build
   - Pre-release Snapshot (Image)
2. Smoke Test
   - OK → Accept Submit
3. Integration Build
   - Snapshot (Image)
4. Tizen:Mobile
5. smoke test
   - QA Test
   - Manual, TCT

Official/Milestone Release
- OK
TIZEN DEVELOPERS CONFERENCE 2015 SHENZHEN
Part 2. Customizing Development Process: Recipes
Recipes with Docker

To support your customization, we prepared Docker VM Images

Recipes

Part 2. Scaling Development Process

GBS with your local git/rpm repository
- Recommended to single developers

GBS local fullbuild with your local git/rpm repository
- You can use Jenkins to operate fullbuild process

GBS local fullbuild with Gerrit server
- Fullbuild process by Jenkins, Code Review by Gerrit

OBS with Build Data
- Build Data of Tizen to reduce initial build time
- You can change Build Data with your own needs
Recipes: For Large Team

Full set of Tizen Infrastructures are ready with Docker (except smoke test)

• In Jenkins, continuous integration processes are implemented
• So, you can choose which process you will operate
Demo: Tizen Infrastructure with Docker

Your Team Grows…

You Need to Scale and Optimize Performance of Your Infrastructure

- For GBS Local fullbuild with Jenkins, you can consider:
  - More Jenkins slave nodes to operate fullbuild
  - tmpfs for BUILDROOT to dramatically reduce build time

- For OBS, you can consider:
  - More build workers
  - powerhost (special build worker with higher configurations) to reduce build time of bottleneck packages
  - to extend backend of OBS server to add more and more workers
    - high-speed network among OBS servers and workers
Conclusion

Tizen development process is designed:
- to provide automated process with less human-intervention
- with many open source tools and services

Future plans
- Enhance performance and add developer-friendly functions to the tools and services of Tizen
- Infrastructure by VM still needs configurations → VM with templates
- Prepare more operation manuals for each recipes

If you need supports:
- Leave JIRA tickets to https://bugs.tizen.org/jira/projects/TINF
- Contact Jong Woo, Chae (jongwoo.chae@samsung.com)
References

- https://wiki.tizen.org/wiki/Tizen_Governance
- https://wiki.tizen.org/wiki/GBS_local_full_build_Docker_image
- https://wiki.tizen.org/wiki/GBS_Performance
- https://wiki.tizen.org/wiki/OBS_Performance
- http://download.tizen.org/docker
Thanks!!!