Development Model for Tizen 3.0

Anas Nashif, Intel

TIZEN™
DEVELOPER CONFERENCE
2013
SAN FRANCISCO

‘PLATFORM’ DEVELOPMENT MODEL

Moving from in-out to out-in development

- Until now, code was released at milestones and moved to Tizen.org as public releases
- Only Tizen release spins and development on top of releases is happening on tizen.org
- Core and Platform development is happening elsewhere!
- No continuity and transparency

Moving from in-out to out-in development

- **Now**, the main area of development and contributions is Tizen.org (out).
- For productisation/commercialization and depending on profile policies and rules, main code tree can be pulled and built anywhere by anyone
- Tizen.org is becoming the primary development hub for Tizen

Tizen.org is now the primary development hub for the Tizen Platform

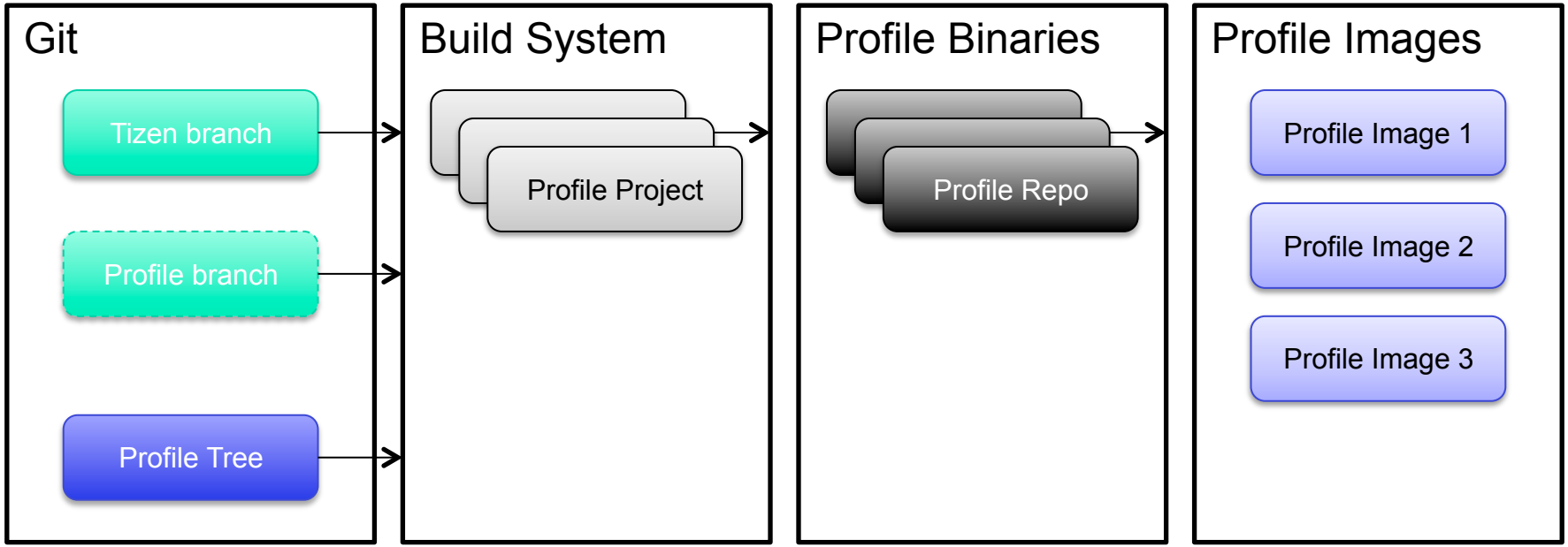
Tizen.org Code Base

- **Keep code close to upstream where possible, accept only upstream-able changes**
- **Latest version does not trump requirements and API compatibility and stability**
- **Goal is to reduce duplication among projects and have 1 component run on different profiles rather than fork and diverge**
  - Same code base but with different build options and enabled features
  - Maintainers have to be Tizen focused rather than profile specific
  - Avoid hardcoding

What makes a profile?

- Majority of code comes from the Tizen branch, the main development code line (platform/*)
- Profile specific projects (profile/*)
- Optional profile specific stabilization branch with changes related to a release (i.e. weekly release)

Code and Output Management



TIZEN 3.0

Tizen 3.0:

- Developed on tizen.org
- Logical evolution from previous Tizen releases
- Configurable and multi-profile support
- Updated toolchain and base system
- Reworked project structure
- Managed innovation through requirements
- Forward looking
- Upstream-able changes

3.0 vs. 2.1

- Modern toolchain
- Refreshed and up to date base OS packages
- Systemd only (remove legacy sysv mode)
- Multiple display system support (Wayland)
- Configurable and supports multiple profiles with one code base
- Strict packaging guidelines
- Changes will undergo more review and scrutiny on Tizen.org, starting with automated checks and all the way to extensive developer, contributor and community review

3.0 Is about Scalability

- **Many profiles**
- **Many devices**
- **Many configurations**
- **Many architectures**

For example:

Smartphone device XYZ on ARM
produced from same platform code as an
IVI device YYY for car ABC running on IA

Project Structure (1/2)

- **project (** ← metadata**)**
  - common
  - ivi
  - mobile
- **platform**
  - upstream (← external)
  - adaptation
  - framework
    - native (← osp)
    - web
  - core
    - api
    - …
  - kernel
- **apps**
  - core
    - preloaded
  - native
    - sample
    - preloaded
  - web
    - sample
    - preloaded
- **sdk**
  - upstream
  - toolchain
  - ide
- **tools**
  - ??
- **scm**
  - builds
  - tools
- **test**
  - tools
  - testsuites
- **doc**
  - platform
  - apps
  - sdk
  - test
- **profile**
  - mobile
    - platform
    - Apps
    - …
  - ivi

Project Structure (2/2)

project

- metadata and information for each profile.

platform

- Platform specific files
- core means “core platform” in tizen 2.0
- Upstream (upstream project, OSS)

apps

Applications

SDK

IDE, toolchain and tools

SCM

- Build script and tools to build platform and SDK

test

- Test tools, test harness, testsuite and tct for each profile

doc

- Documentation for each layer and profiles

profile

- Profile specific git projects and local changes

TIZEN™

DEVELOPER CONFERENCE 2013 SAN FRANCISCO