Tizen, a Wayland-powered operating system

Manuel Bachmann
<manuel.bachmann@open.eurogiciel.org>
Eurogiciel

Open-source development and integration:

- Maintainers for tizen.org (Base, Test, Web Framework, … domains)
- Embedded systems for real-time multimedia:
  - Widi/Miracast stack,
  - Wayland/Weston,
  - Webkit2 browser with HW acceleration.
- Applications: HTML5/CSS3, jquery, jqmobi, Cordova
- Location: Brittany - France
Manuel Bachmann

<manuel.bachmann@open.eurogiciel.org>

- Previous maintainer of the GTK+3 for Win32 packages;
- Author of a GLX wrapper for EGL: EGLX (1);
- Upstream contributor to Weston, EFL, Ozone-Wayland...;
- Maintains Weston and UI domain packages under Tizen.
Summary

• Tizen has a display system-independent build system

• Wayland and its compositors' shells:
  - XDG-Shell enhancements;
  - IVI-Shell and friends (ICO Homescreen...);
  - Tizen API and web framework adaptations;
  - toolkits and frameworks integration.

• The multiuser challenge

• An enforced security model for graphical applications
Tizen, a display system-independent OS

Tizen has a display-system independent build system
Tizen, a display system-independent OS

Tizen has a display system-independent build system

- Build system uses GIT along with OBS/GBS or Yocto;
- Currently provides X86, AMD64 and ARM binary packages;
- X11 and Wayland packages are built in two separate repositories for each architecture:
  - X11 is: Xorg-server, Enlightenment;
  - Wayland is: Weston, tizen-launcher.
- Shared packages are: Crosswalk, AppFW (ail/aul, app-core, slp-pkngmgr, pkngmgr-info...)
- Macros (%with_x, %with_wayland, ...) ensure consistency of the various builds.
Tizen, a display system-independent build system

from Git repositories to X11/Wayland packages
Tizen, a display system-independent build system

What is in which Wayland profile

- **Tizen Common has:**
  - Patched Weston (*);
  - Crosswalk, EFL and Qt apps;

- **Tizen IVI has:**
  - Patched Weston (*);
  - Embedded-oriented optimizations;
  - Crosswalk with Modello Homescreen (web);
  - IVI-Shell with ICO Homescreen (native);
<DEMO TIME>

- 1) Tizen Common : Weston with tz-launcher and apps
- 2) Tizen IVI with Modello Homescreen
- 3) Tizen IVI with ICO Homescreen
Wayland shells, Tizen API and framework integration
Wayland shells, Tizen API and framework integration

XDG-Shell

- Upstream, still work-in-progress (stable version in Weston 1.7.0 ?)
  - used by the Crosswalk web framework and Tizen extensions to provide Tizen API functionalities:
    - `tizen.application.hide()` is mapped to `xdg_surface_set_minimized()`;
    - `tizen.application.launch()`, on an already started application, will resume it and bring it to foreground with `xdg_surface_present()` (Tizen patch);
  - Hope to see `xdg_surface_present()` in the next upstream release (2).
Wayland shells, Tizen API and framework integration

**<DEMO TIME>**

- 1) `xdg_surface_set_minimized()` in a test app
- 2) `xdg_surface_present()` implementation demo
- 3) Tizen Common with test app and hide()/launch()
- 4) Tizen IVI with Modello Homescreen and hide()/launch()
Wayland shells, Tizen API and framework integration

IVI-Shell

- In review process upstream, hope to see it in Weston 1.7.0;
- Provides a Weston shell interface compliant with the GENIVI specification (http://www.genivi.com);
- Present only on Tizen IVI, used by the ICO Homescreen;
- We maintain an external Weston Git repository with patches applied for reviewers (3), and build it as a separate plugin under Tizen;
- Compatibility layer with XDG-Shell, so that standard applications can use it, has been proven possible and is currently being done (4).
<DEMO TIME>

1) Tizen IVI with IVI-Shell and XDG-Shell compat demo
Frameworks and toolkits integration

- Tizen API is implemented under the form of extensions for the Crosswalk web framework, and makes sure correct shell functions are called each time;
- EFL: XDG-Shell and IVI-Shell pushed and integrated upstream\(^{(5)}\);
- Qt: XDG-Shell pushed and integrated upstream by Philippe Coval, IVI-Shell under review\(^{(6)}\);
- Crosswalk/Ozone-Wayland: XDG-Shell and IVI-Shell pushed and integrated upstream\(^{(7)}\).
<DEMO TIME>

- 1) Minimization with EFL, Qt, and Chromium/Ozone-Wayland
The multiuser challenge
The multiuser challenge

- Tizen 3 is switching from a single-user approach to a multiuser one \(^{(8)}\);
The multiuser challenge

- Tizen 3 is switching from a single-user approach to a multiuser one \(^{(8)}\);
- The display part of the implementation will allow the application framework to position a surface on a specific screen \(^{(9)}\):
  - Application framework API calls and policy;
  - possible IVI-shell use? XDG-shell support?
The multiuser challenge

- Tizen 3 is switching from a single-user approach to a multiuser one (8);
- The display part of the implementation will allow the application framework to position a surface on a specific screen (9):
  - Application framework API calls and policy;
  - possible IVI-shell use? XDG-shell support?
- Weston supports static seats configuration(with udev) – consider switching to a more dynamic approach linked to systemd and Tizen Login Manager?
The multiuser challenge

<VIDEO>

- Weston configured for multi-seat with udev
The need for an enforced security model
An enforced security model for graphical applications

- Tizen 3 is switching from a single-user oriented privacy-manager to Cynara.
An enforced security model for graphical applications

- Tizen 3 is switching from a single-user oriented privacy-manager to Cynara;
- When a user installs an application which requests some privileges, and Cynara is informed, how will it display actual notifications and confirmation requests to the end-user?
An enforced security model for graphical applications

- Tizen 3 is switching from a single-user oriented privacy-manager to Cynara;
- When a user installs an application which requests some privileges, and Cynara is informed, how will it display actual notifications and confirmation requests to the end-user?
- A security manager sometimes needs to prevent a surface to do a specific thing without an authorization (going full-screen, or taking a screenshot e.g.). How will this be implemented compositor-side?
An enforced security model for graphical applications

- Tizen 3 is switching from a single-user oriented privacy-manager to Cynara;
- When a user installs an application which requests some privileges, and Cynara is informed, how will it display actual notifications and confirmation requests to the end-user?
- A security manager sometimes needs to prevent a surface to do a specific thing without an authorization (going fullscreen, or taking a screenshot e.g.). How will this be implemented compositor-side?
- How that relates to Wayland Security Module.
The need for an enforced security model

<DEMO>

- UI pop-notification in Tizen
- Requests for privileges (taking a screenshot)
Links

1. EGLX GitHub repository: [https://github.com/Tarnyko/EGLX](https://github.com/Tarnyko/EGLX)
3. weston-ivi-shell upstream adaptation GitHub repository: [https://github.com/Tarnyko/weston-ivi-shell](https://github.com/Tarnyko/weston-ivi-shell)
4. ivi-shell xdg-shell compatibility layer feasibility: [https://www.mail-archive.com/ivi@lists.tizen.org/msg02702.html](https://www.mail-archive.com/ivi@lists.tizen.org/msg02702.html)
5. EFL upstream wayland shells: [http://git.enlightenment.org/core/efl.git/commit/?id=87f02170e659678d7a2f000e6850bd3a29962756](http://git.enlightenment.org/core/efl.git/commit/?id=87f02170e659678d7a2f000e6850bd3a29962756)
   - [https://git.enlightenment.org/core/efl.git/commit/?id=50287ab731d4d87170238b365203e830edc038d5](https://git.enlightenment.org/core/efl.git/commit/?id=50287ab731d4d87170238b365203e830edc038d5)
• (6) : Qt upstream wayland shells :
  https://bugreports.qt-project.org/browse/QTBUG-38633 - https://bugreports.qt-project.org/browse/QTBUG-41172

• (7) : Ozone-Wayland upstream wayland shells :
  https://github.com/01org/ozone-wayland/commit/5f8a34c613ba826c7994c81d03f87df19f48881d -
  https://github.com/01org/ozone-wayland/commit/a034a018b6ec317ec5559dcce6efec9

• (8) : Tizen multi-user architecture :
  https://wiki.tizen.org/wiki/Multi-user_Architecture

• (9) : Tizen multi-user display management :
  https://wiki.tizen.org/wiki/Multi-user_DisplayManagement
Q&A
That's all folks!