

Tizen 3.0 Multi-User Features



Baptiste Durand
Software Engineer
Eurogiciel

<baptiste.durand@open.eurogiciel.org>

TIZEN™
**DEVELOPER
CONFERENCE**
2014
SAN FRANCISCO

Agenda

- Tizen Multi User System and **associated requirements**
- What has changed ? What are the **upcoming updates** ?
- Status / RoadMap

Tizen Multi User System



Multi User System



Tizen Multi User System

- **Categories of Multi User System**

- Multi Seat System

- The system can be used at the same time by several users and they could share the same display.**

- Single Seat System

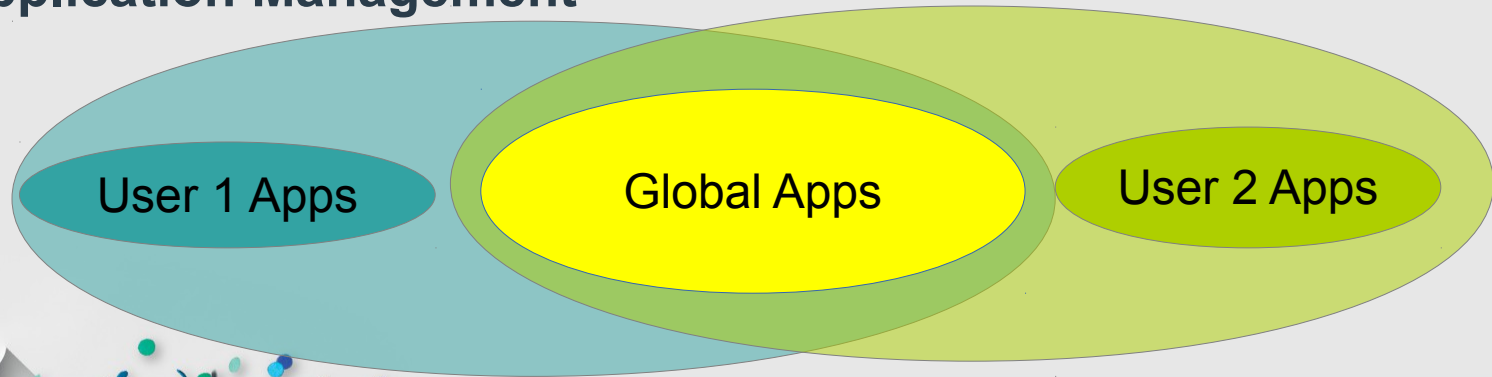
- The system can be used by one user at the given time**

Multi User System : Key Requirements

- **User Management**

- Needs user account
- Each user has its Environment / data / applications
- “Administrator” concept

- **Application Management**



Multi User System : Key Requirements

- **Security Management**

- User can only read its own data and system data
- Security must protect the user data, the system data
- Security must control resources access per User

- **Resource / Service Management**

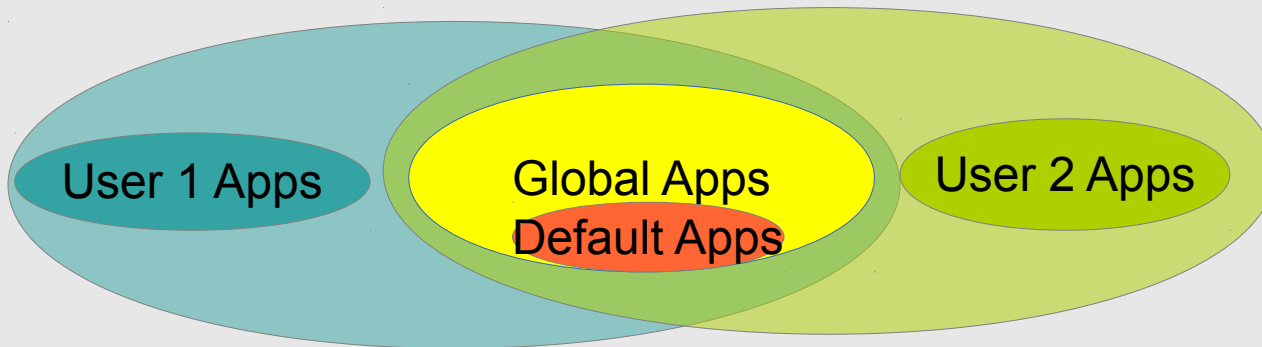
- Shared Access
- Exclusive Access
- Service Concept

Multi Seat System : IVI Needs



- **Guest user**

- Is logged in by default
- Starts Default Applications



Multi Seat System : IVI Needs




- **Resources Management**

- Users can share the same display (e.g. passengers and driver)
- Users can also exchange seats (and so displays)

Single Seat System

- **Similar to profile system (shared phone TV)**
- **No specific resource management**
- **Security Management**





What has changed? Upcoming Updates

Multi User Related changes

- **Main Challenge**
 - Keep the project consistent during migration process
- **Tizen Platform Configuration**
- **Application Framework**
- **Resources Services**
- **Service API**
- **User Management**

Upcoming updates : Tizen Platform Configuration

- **Package source code compliance**
 - The source code was not compliant with Multi User architecture
 - Hard-coded path
 - Hard-coded uids / gids / Username related to user App

Upcoming updates : Tizen Platform Configuration

- Package source code compliance

```
filesystem/filesystem_instance.cc
[...]
```

```
namespace {
const unsigned kDefaultFileMode = 0755;
const char kDefaultPath[] = "/opt/usr/media";
const char kPathSounds[] = "/opt/usr/media/Sounds";
const char kPathImages[] = "/opt/usr/media/Images";
const char kPathVideos[] = "/opt/usr/media/Videos";
const char kPathDownloads[] = "/opt/usr/media/Downloads";
[...]
```

```
AddInternalStorage(kLocationMusic, kPathSounds);
AddInternalStorage(kLocationImages, kPathImages);
AddInternalStorage(kLocationVideos, kPathVideos);
AddInternalStorage(kLocationDownloads, kPathDownloads);
AddInternalStorage(kLocationDocuments, kPathDocuments);
[...]
```



Upcoming updates : Tizen Platform Configuration

- **Tizen Platform Configuration**

- Goal : Having a unique package that defines how the environment variables are made
- Based around on 1 configuration file (meta file) and 1 wrapper library
- Easy to use (as it is bash readable)

Upcoming updates : Tizen Platform Configuration

- Extract of meta file

```
##### Base dirs #####  
# Tizen 3 (single user):  
#TZ_USER_NAME=app  
#TZ_USER_HOME=$TZ_SYS_HOME/app  
#TZ_USER_SHARE=/opt/usr/share  
#TZ_USER_DATA=/opt/usr/data  
#TZ_USER_ETC=/opt/usr/etc  
#TZ_USER_LIVE=/opt/usr/live  
  
# Tizen 3 (multi user):  
TZ_USER_NAME=$USER  
TZ_USER_HOME=$HOME  
TZ_USER_SHARE=$TZ_USER_HOME/share  
TZ_USER_DATA=$TZ_USER_HOME/data  
TZ_USER_ETC=$TZ_USER_HOME/etc  
TZ_USER_LIVE=$TZ_USER_HOME/live  
[...]
```


Upcoming updates : Tizen Platform Configuration

- Example of usage

```
filesystem/filesystem_instance.cc
[...]
```

```
namespace {
const unsigned kDefaultFileMode = 0755;
const char kDefaultPath[] = "/opt/usr/media";
const char kPathSounds[] = "/opt/usr/media/Sounds";
const char kPathImages[] = "/opt/usr/media/Images";
const char kPathVideos[] = "/opt/usr/media/Videos";
const char kPathDownloads[] = "/opt/usr/media/Downloads";
[...]
```

```
AddInternalStorage(kLocationMusic, kPathSounds);
AddInternalStorage(kLocationImages, kPathImages);
AddInternalStorage(kLocationVideos, kPathVideos);
AddInternalStorage(kLocationDownloads, kPathDownloads);
AddInternalStorage(kLocationDocuments, kPathDocuments);
[...]
```



Upcoming updates : Tizen Platform Configuration

- Example of usage

```
filesystem/filesystem_instance.cc
```

```
[...]
```

```
namespace {
```

```
- const unsigned kDefaultFileMode = 0755;  
- const char kDefaultPath[] = "/opt/usr/media";  
- const char kPathSounds[] = "/opt/usr/media/Sounds";  
- const char kPathImages[] = "/opt/usr/media/Images";  
- const char kPathVideos[] = "/opt/usr/media/Videos";  
- const char kPathDownloads[] = "/opt/usr/media/Downloads";
```

```
[...]
```

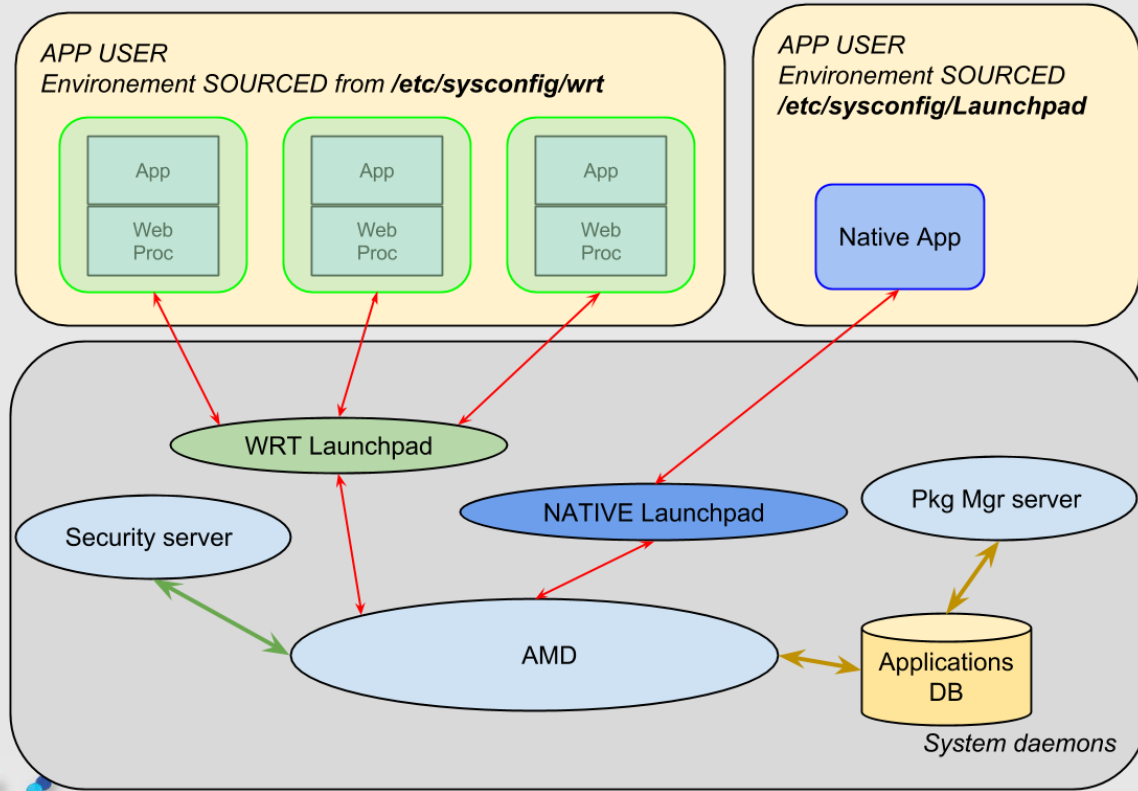
```
+ AddInternalStorage(kLocationMusic, tzplatform_getenv(TZ_USER_SOUNDS));  
+ AddInternalStorage(kLocationImages, tzplatform_getenv(TZ_USER_IMAGES));  
+ AddInternalStorage(kLocationVideos, tzplatform_getenv(TZ_USER_VIDEOS));  
+ AddInternalStorage(kLocationDownloads, tzplatform_getenv(TZ_USER_DOWNLOADS));  
+ AddInternalStorage(kLocationDocuments, tzplatform_getenv(TZ_USER_DOCUMENTS));
```

```
[...]
```



Upcoming updates : Application Framework

Tizen 2.x Model

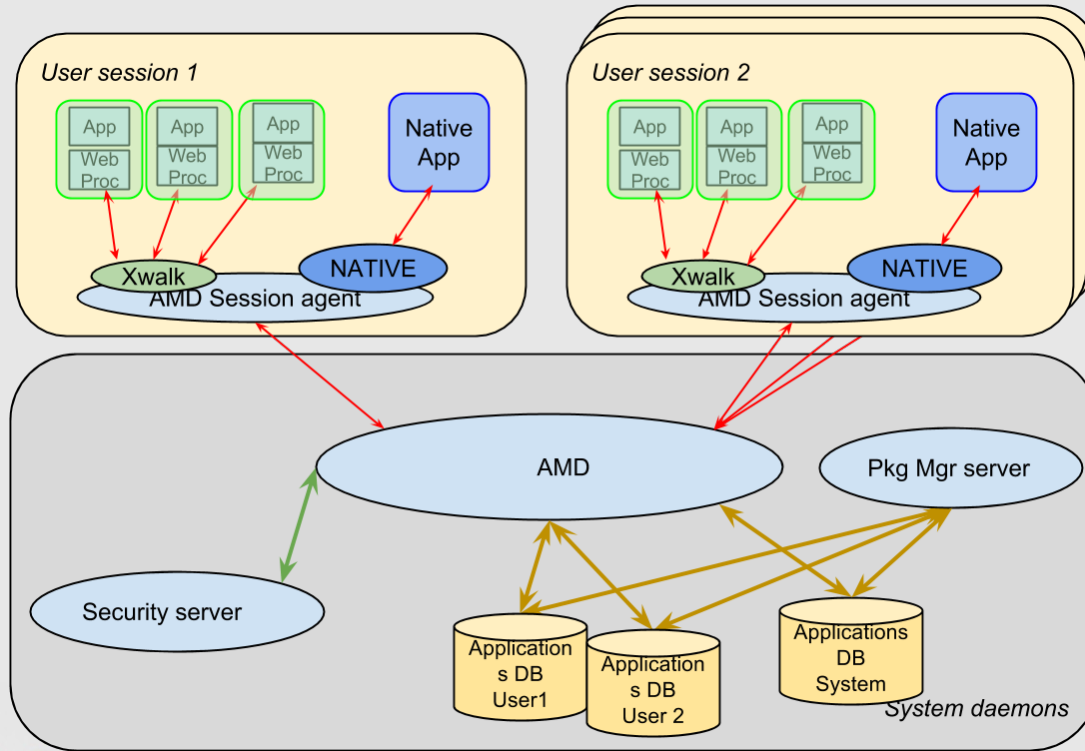


Upcoming updates : Application Framework

- On tizen 2, All apps were launched in a static 'sourced' Environment with the ID App
- 'User' concept needs to be introduced

Upcoming updates : Application Framework

Tizen 3.0 Model



Upcoming updates : Application Framework

- **Application are launched in the User Environment provided by systemd session.**
- **Each User can access to :**
 - A set of databases corresponding to System/common Application
 - Another corresponding to it own databases.
 - Application Databases look like the Tizen 2.0 ones
 - Core API should be extended

Upcoming updates : Resources Services

- **Resources not related to a Tizen framework should be shared:**
 - Display (IVI Requirement)
 - Audio
- **Resources related to a framework should be managed:**
 - Service API Daemon?
 - ...

Upcoming updates : Resources Services

Weston

- **Initial Case**

- Officially 1 compositor per user session
- User cannot switch of display without restarting the weston service

- **What's done**

- 1 compositor for the whole system
- Weston daemon is responsible to allocate visual resources to a user
- Launched as “display” user (system uid)

Upcoming updates : Service API

State PROPOSAL

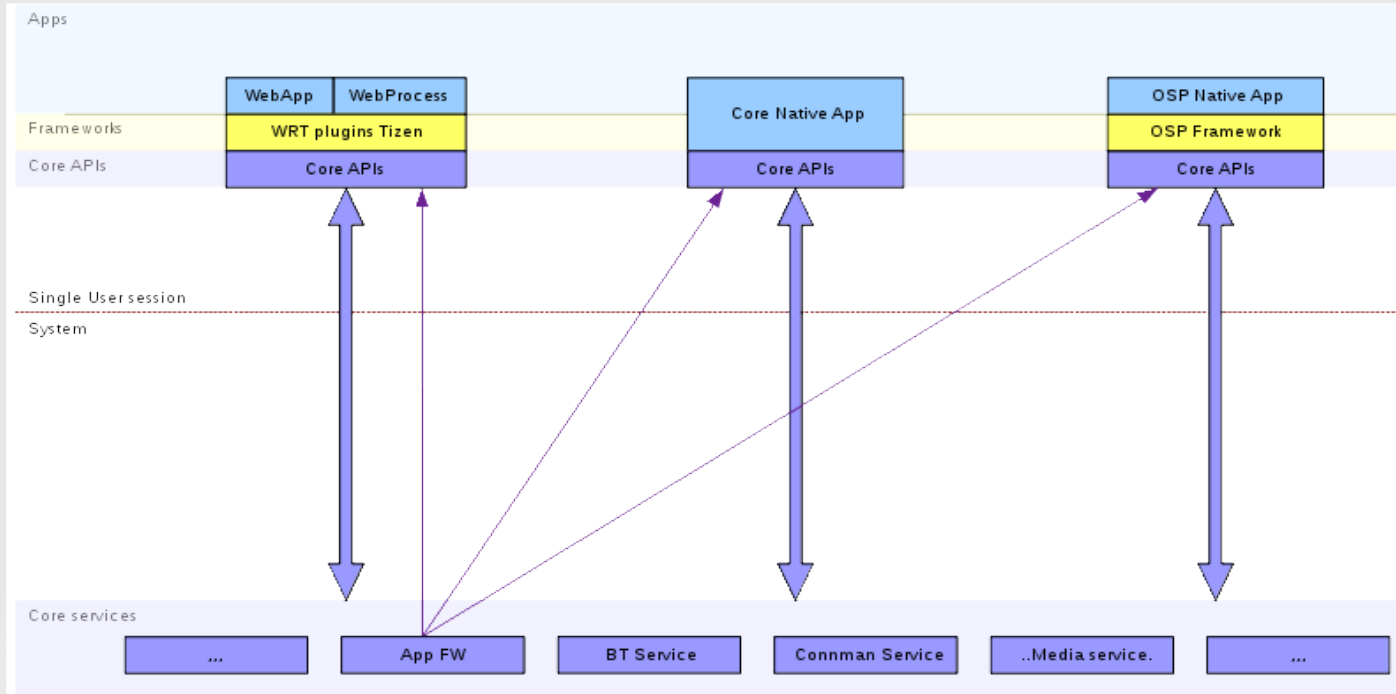
Other investigations are being also tested :

- Containers
- Smack containers launcher

Upcoming updates : Service API

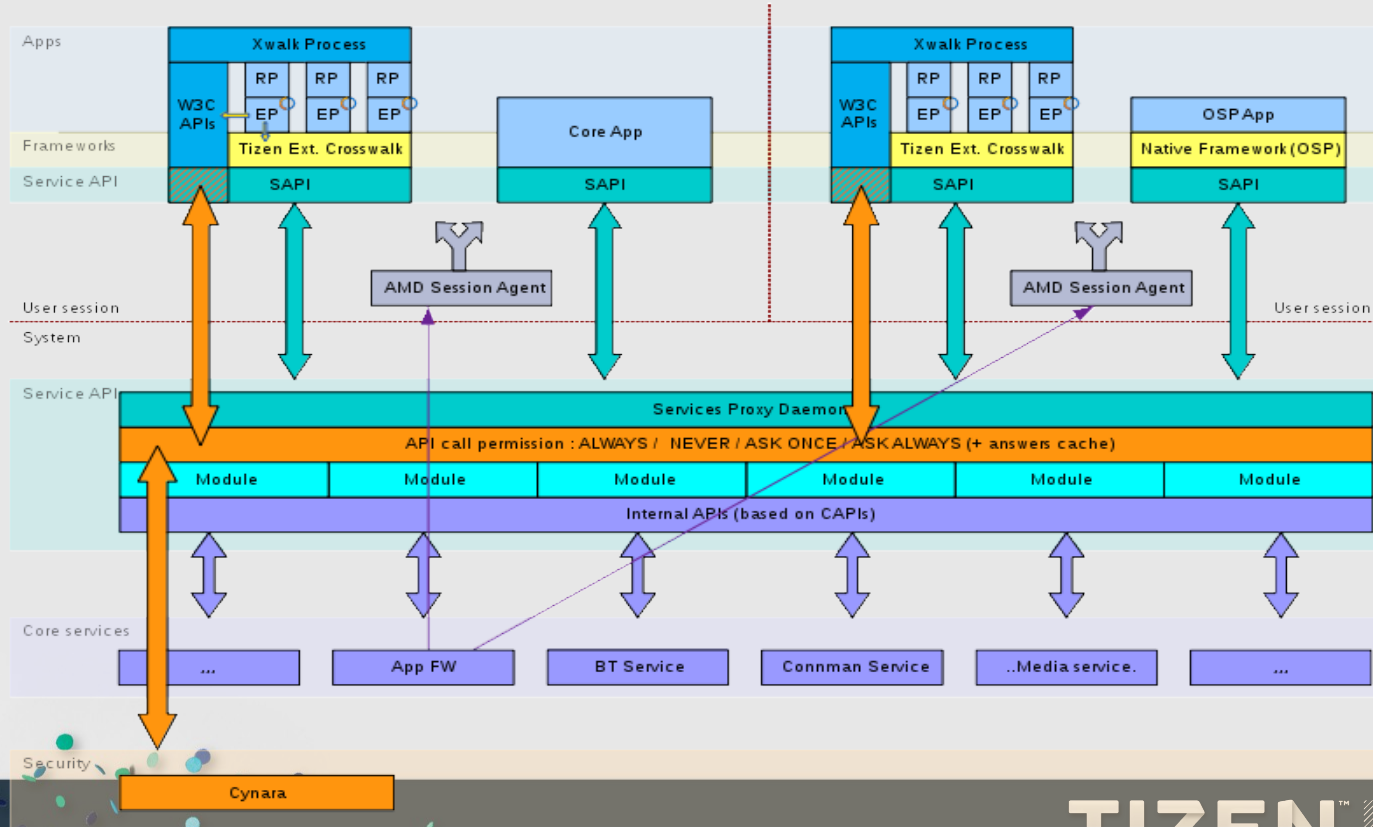
- Framework / CORE API in Tizen2.0

Model in
Tizen 2.X



Upcoming updates : Service API

Model in
Tizen 3



Upcoming updates : Service API

- **Framework / CORE API Tizen3.0**

- New services API (SAPI) is introduced as a drop-in replacement for Core APIs
- SAPI will be as close as possible as actual Core APIs
- Calls are sent to the Services Daemon (local RPC)
- The Services Daemon gets the app. credentials: uid, git, smack label and checks if the call is allowed (using Cynara)
- The call is then executed using the actual Core APIs and the result sent back to client.
- Extra features could be implemented inside the SAPI Daemon (e.g. dynamic resources management)

Upcoming updates : TLM / GUMD

- **Gumd**
 - Manages users accounts
 - DBUS API
- **Tizen Login Manager**
 - Opens user sessions
 - Provides : DBUS API
 - Support : GUI less / auto login
- **Should be adapted to include admin notion**

A decorative graphic on the right side of the slide. It features a white triangular shape pointing downwards, which contains a stylized city skyline in grey and tan. Below the skyline are blue wavy lines representing water. To the right of the triangle is a large yellow circle representing the sun. Scattered throughout the scene are various colored circles and small shapes in shades of blue, green, and dark blue.

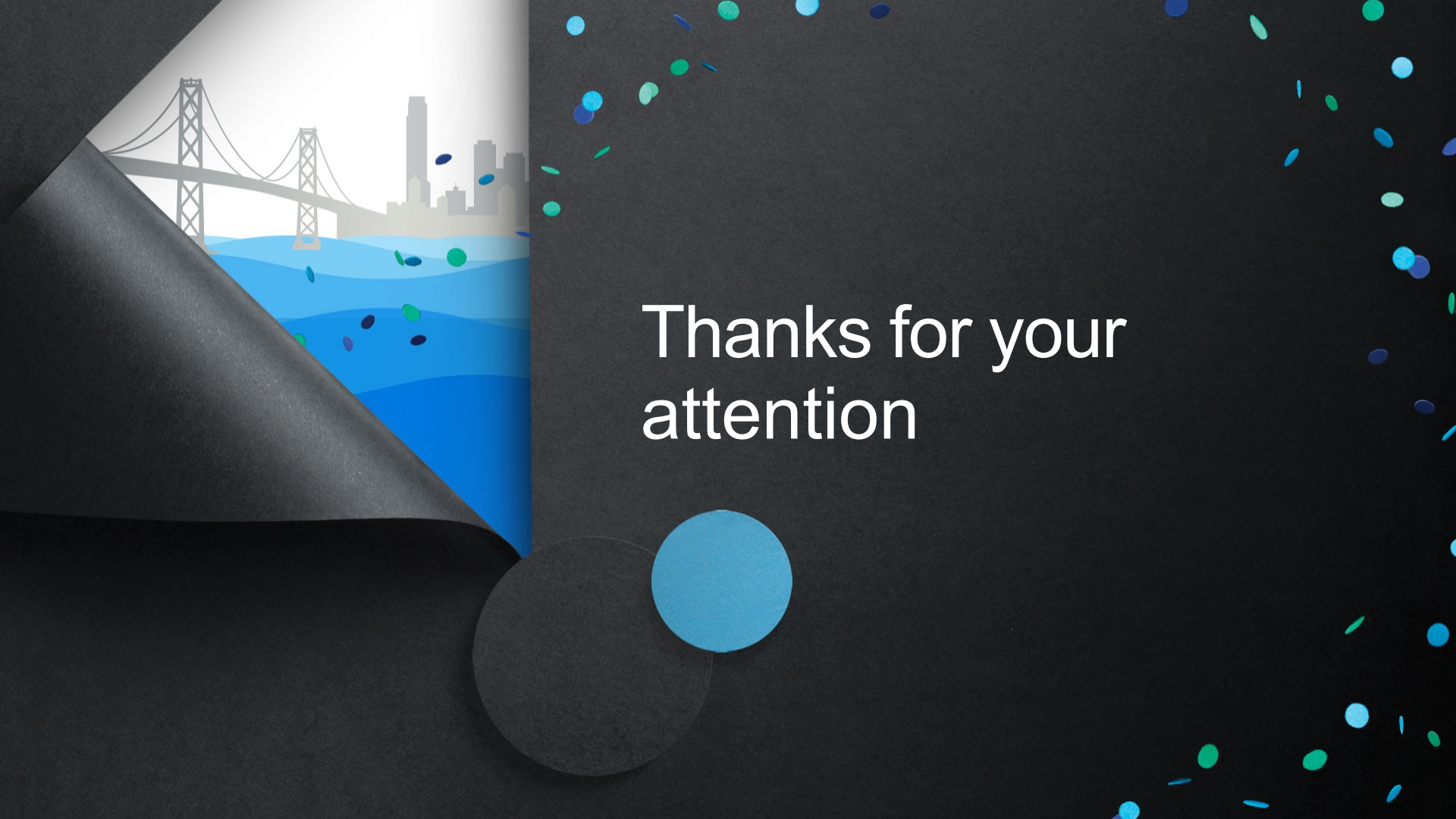
RoadMap and Status

Status and estimated deliveries

	Status	Comments	Estimate delivery
Tizen Configuration Platform	Fully Integrated since start of the year	Platform Meta should be adopted by Profile	X
Application Frwk	Implemented	Some API extension are missing Crosswalk Need to be integrated	End of this month
Ressources Management	In investigation	Note : Weston Server Fully integrated	Background Work
Service API	Architecture review	Architecture proposal is ready	2014-Q3

Questions?

- Links : https://wiki.tizen.org/wiki/Multi-user_Architecture



Thanks for your
attention