Application FW Update

May 11, 2017 | Birmingham UK, GENIVI AMM

Gururaja
Architect, Bosch

Gunnar
Architect, GENIVI Alliance

Application FW

Application FW

- Application framework consists of a set of software components
  - Which enables interaction of Apps with the underlying GENIVI platform
  - App Download and Install management
  - Defines broadly the App structure
  - App life cycle management
  - Data Management
  - Defines security model and access mechanism
  - Last User context handling

GENIVI®

Application FW

- GENIVI Reference Architecture works with two different types of Applications
  - Managed Apps
  - Native Applications
- Application can be a UI based or without a UI

GENIVI®

Native Applications

- Developed under OEM supervision or otherwise vetted
- “Trusted Application” principle for user data separation
- Security : Follow principle of least possible privilege to reduce damage in case of exploit
- Applications by nature are part of System SW update

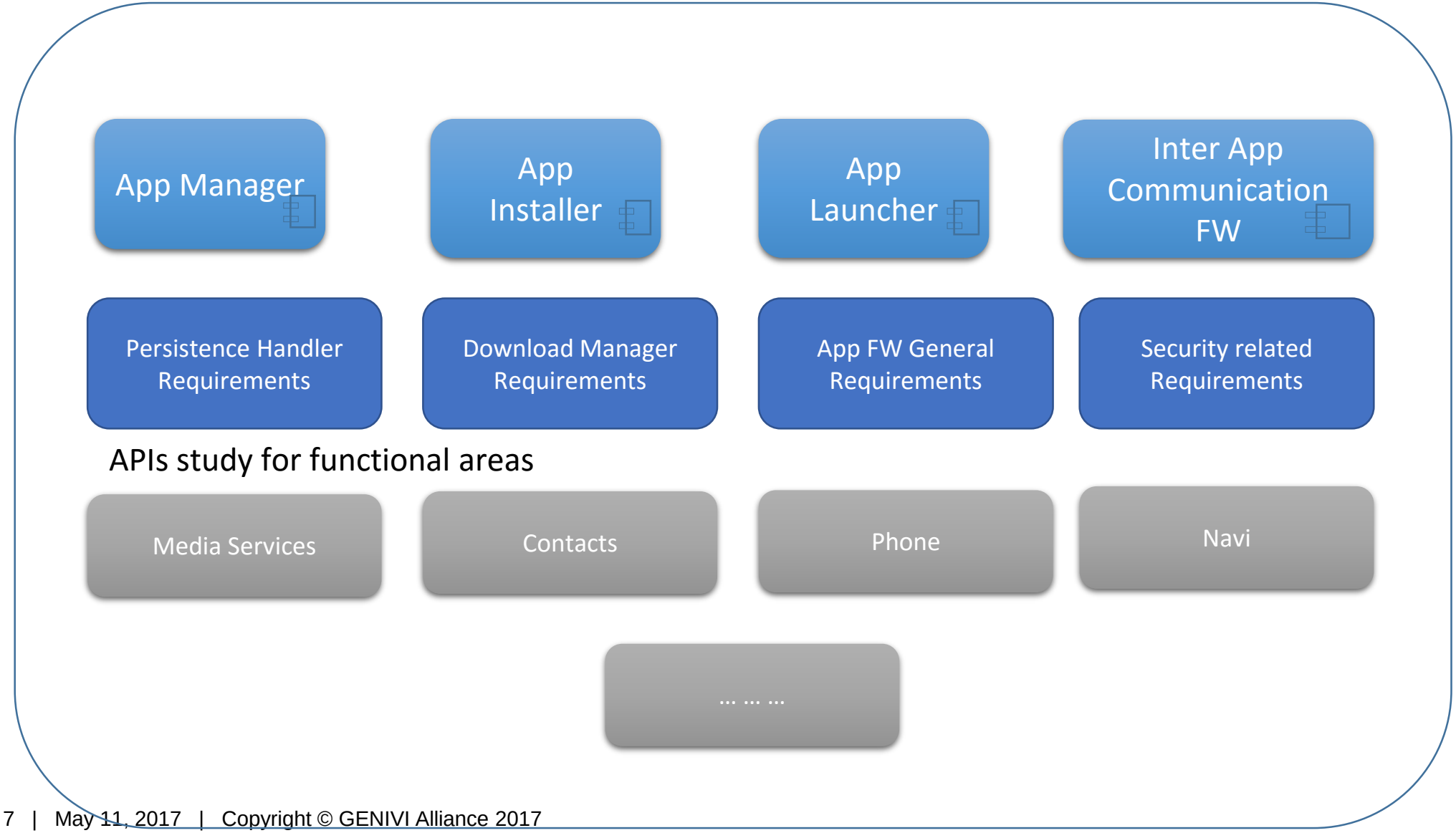
GENIVI®

Managed Apps

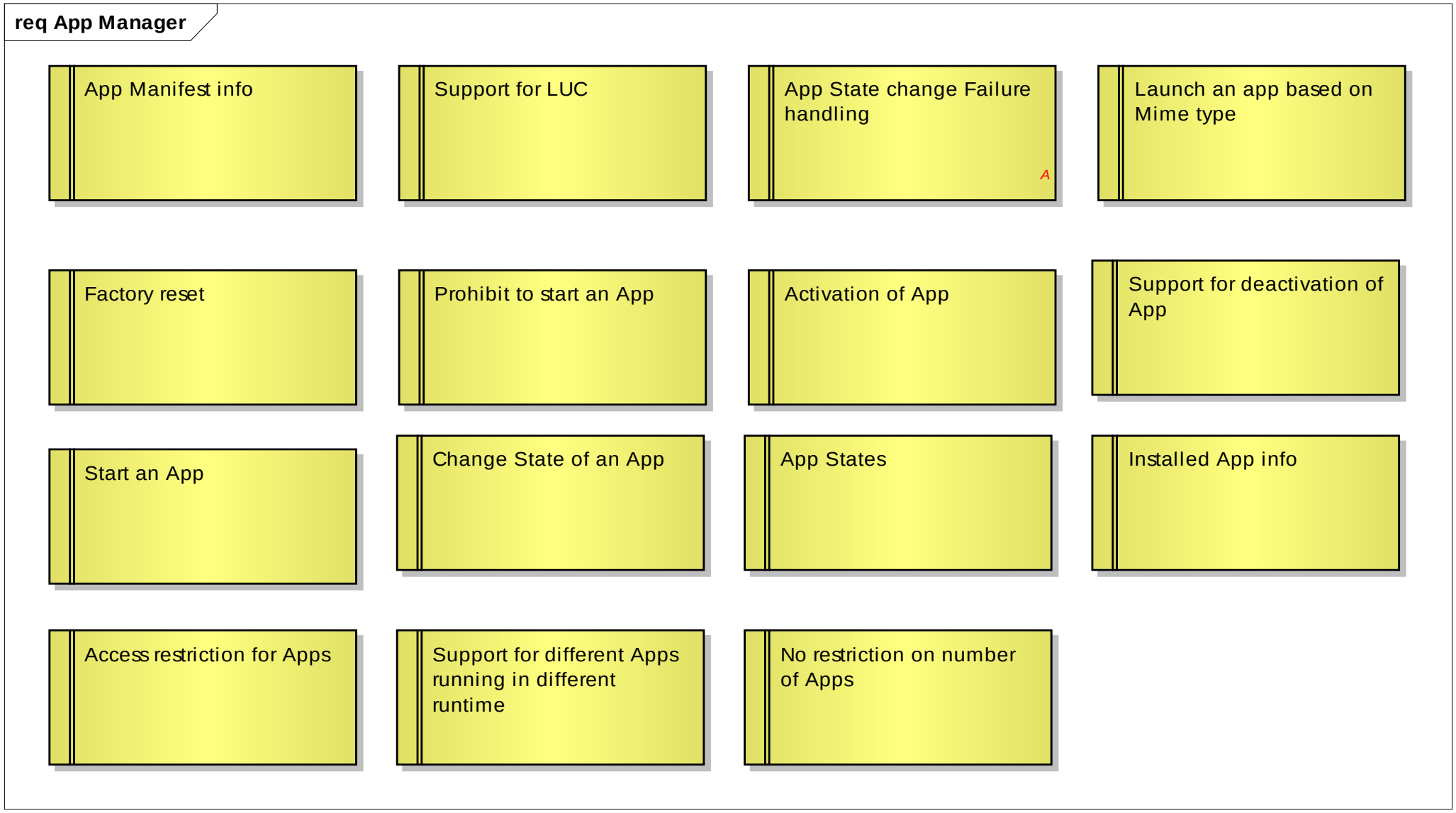
- Downloadable apps
- Third party developed Apps
- Provide full sandboxing
- May reuse some already existing foreign frameworks
- Extended/alternate lifecycle strategy
- Extended/alternate user management strategy
- Extended/alternate persistence management strategy

GENIVI®

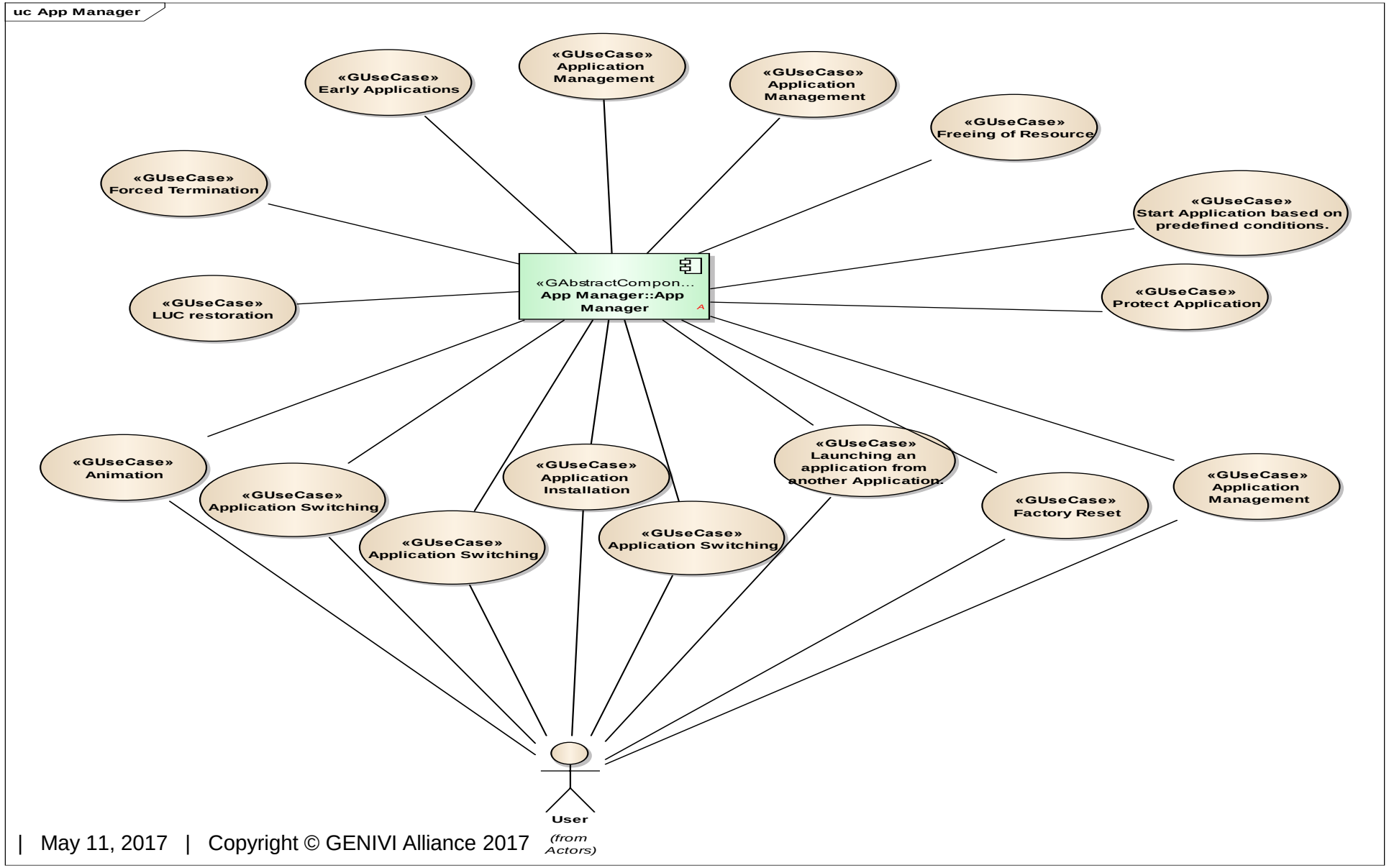
Application FW



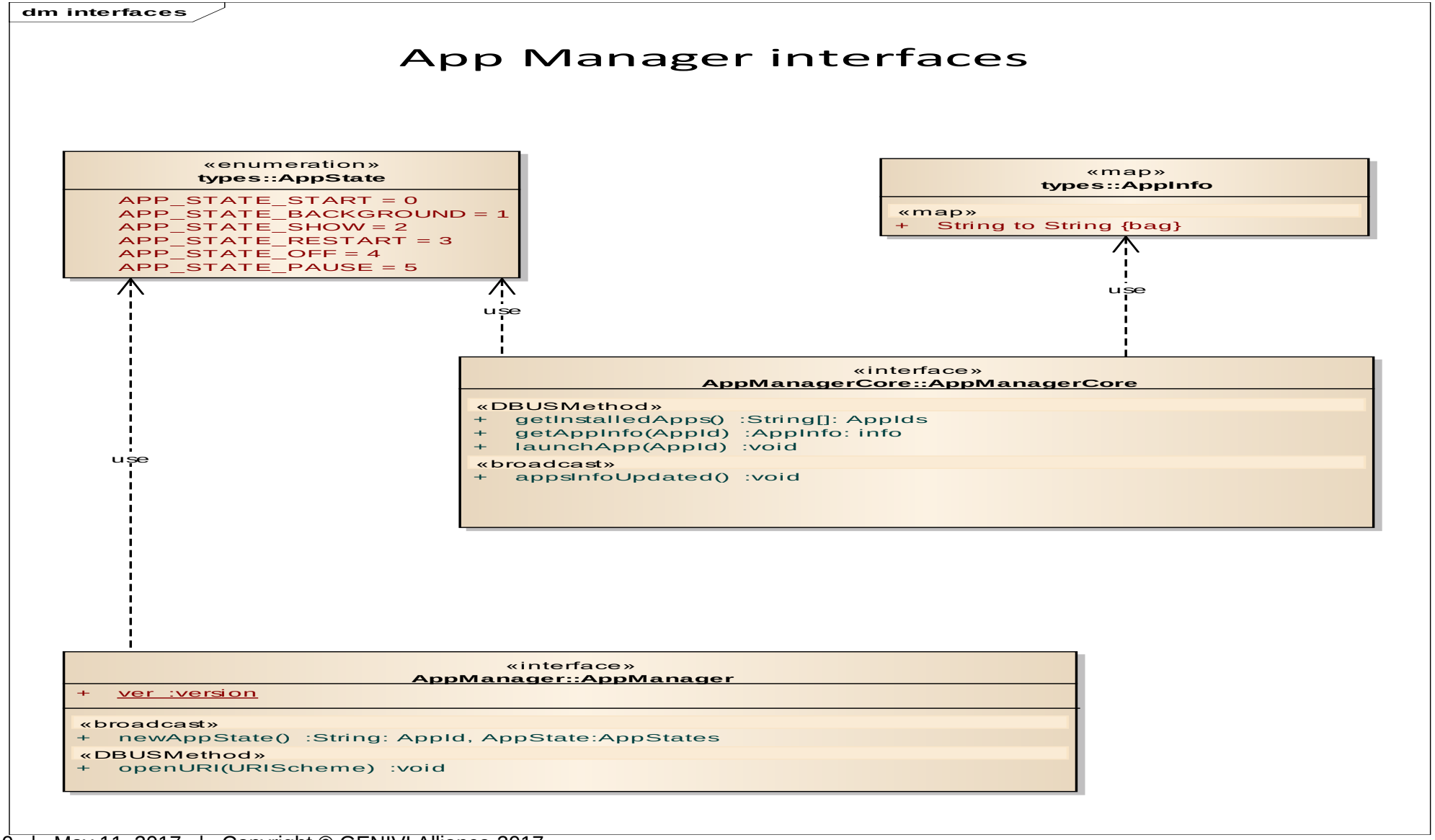
App Manager [P2-AC] – Requirements



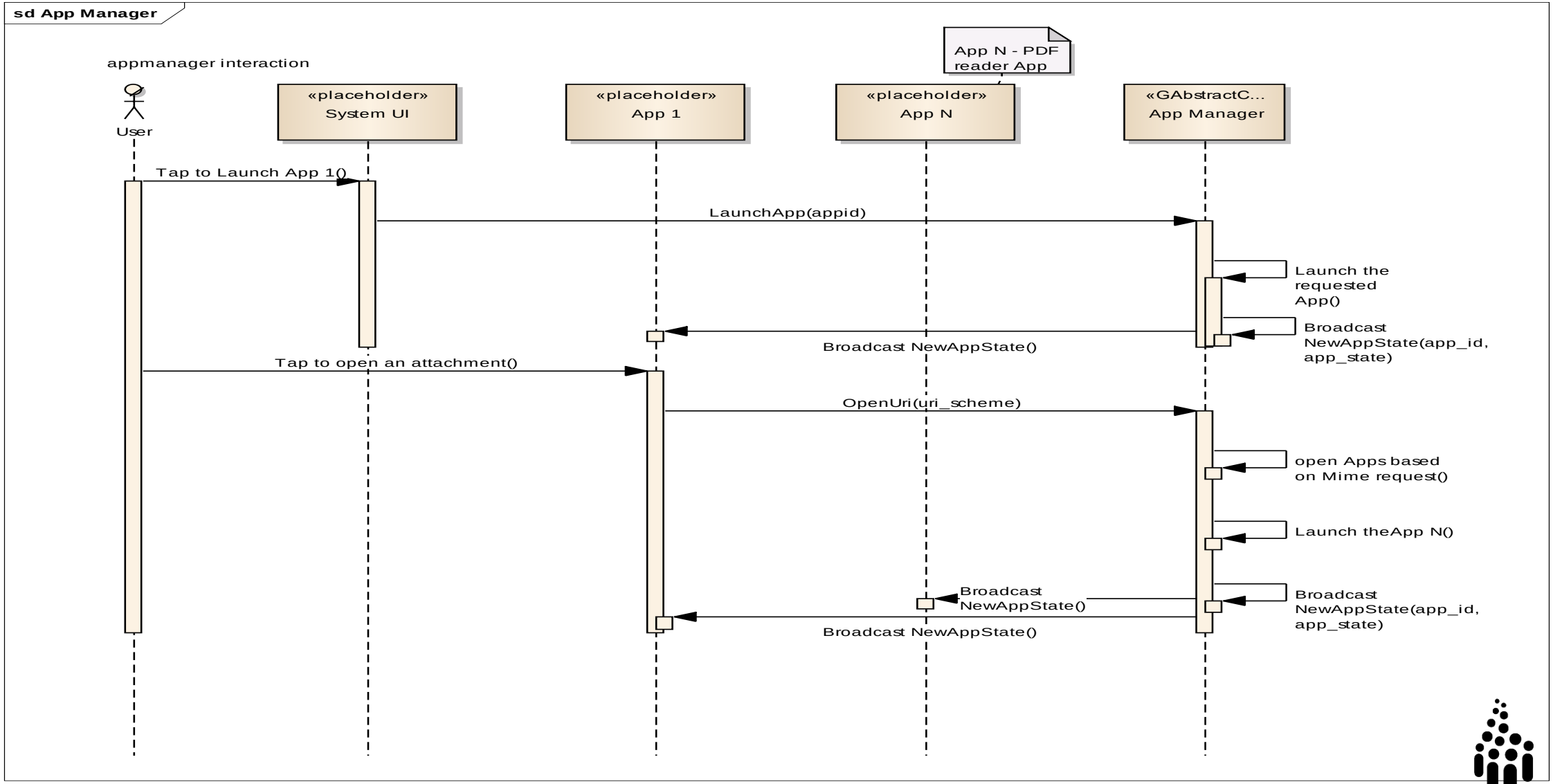
App Manager [P2-AC] – Use Cases



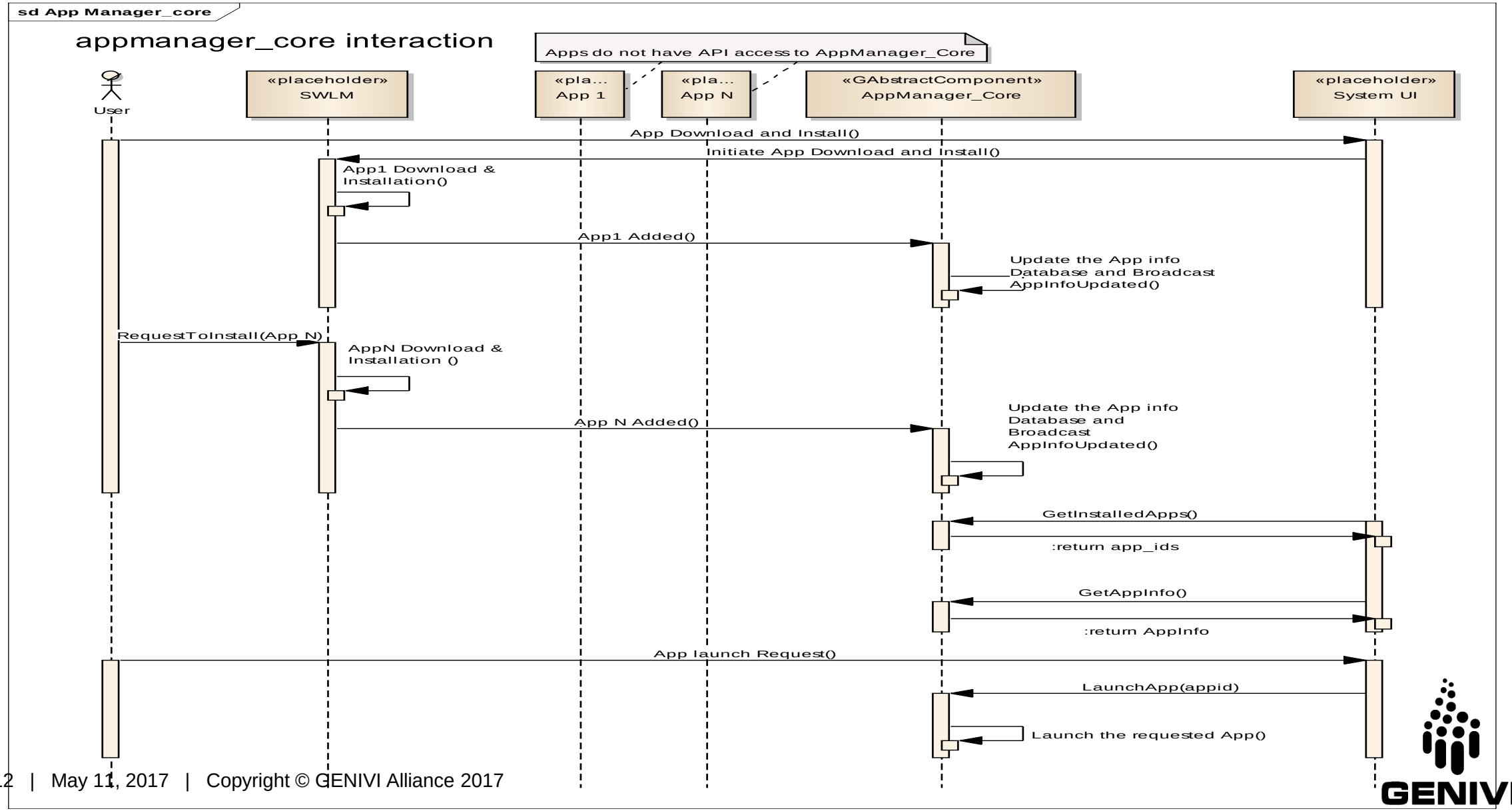
App Manager [P2-AC] – Interfaces



App Manager [P2-AC] – Use case Realization



App Manager Core [P2-AC] – Use case Realization



App Installer – [P2-PC] -> Orion Release

- Support install and un install of apps (from apps store / vendor specific way)
- Define structure for App Meta Data bundle
  - Unique ID, Icon, App Info etc.
- Provision for defining the vendor specific metadata
- Block those Apps without the defined metadata

GENIVI®

App Installer – [P2-PC] -> Orion Release

- Must not limit the number of Apps for installation
- Provision for distinguish between User installable and pre installed app
- Support for uninstall request while app is being run
- Allow for newer version of App installation
- Upgrading apps installed from removable storage

GENIVI®

App Launcher – [P2-PC] -> Orion Release

- Launch the Application based on User request / Entry point
- Upon App startup, Notification to other app must not be visible
- Launch via URI, Document Launching
- If App Launcher is a compositor ? (Under discussion)
  - Allow allocating of right window
  - Show / Hide of the windows
  - Signal show and hide status to Apps
  - Saving last know window contents
  - Restoring last known window content

GENIVI®

Persistence Handler– [In discussion SI-EG]

- Storing Apps private Date
- Ensure that only that program which is part of a specific app can read/write the app's private data
- Shall not allow different user to access/modify other user’s data
- Shall not allow for an app to know whether another app's private data exists
- Shall not allow for an app to know whether another app's per-app data exists
- Shall provide a mechanism for app programs to store the per-app data. This may be a programmatic API
- Shall provide a mechanism by which an app program's per-app data can all be deleted by another system component … … … …

GENIVI®

Download Manager – [In discussion SI-EG]

- App FW shall allow for multiple downloads of Apps
- It shall queue the downloads
  - if system has a limit on the maximum number of concurrent downloads
  - If system has a bandwidth usage limit
- Pending downloads must be saved periodically, so that they can be resumed automatically on next startup
- The list of pending downloads, their progress and states must be treated as private data
- Different app download queue confidentiality
- Different user download queue confidentiality
- … …. …

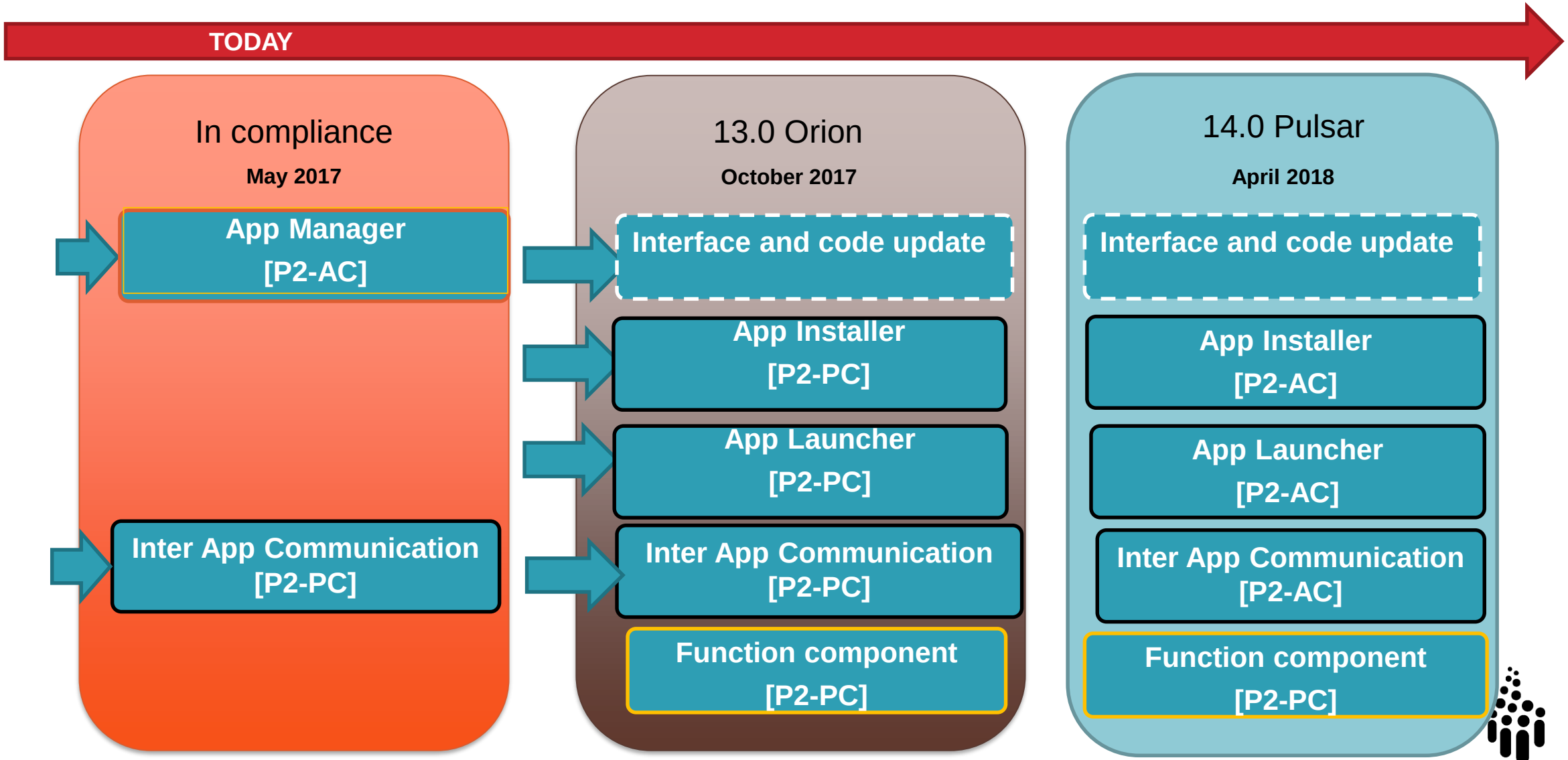
GENIVI®

Security Requirements

- *App integrity*: a malicious or compromised app must not be able to modify the executables and static data of other apps.
- *App confidentiality*: in general, a malicious or compromised app must not be able to list the other apps that are running on the system or the other apps that are installed, either by their bundle names, by their entry points
- *System integrity*: a malicious or compromised app app must not be able to violate the integrity of the system as a whole (for example by modifying the executables or static data of the system, or by altering the system's idea of what is a trusted app source
- *Resource limits*: A malicious, compromised or buggy app might use more than its fair share of system resources, including CPU cycles, RAM, storage (flash) or network bandwidth

GENIVI®

App FW roadmap



App FW contributors

BOSCH
Invented for life

COLLABORA

PELAGICORE
Experience Change

GENIVI®

Application FW Weekly Telco

on every Tuesday @ 13:00 CET

Join WebEx meeting

Meeting number (access code): 805 300 498

Flatpak:App Packaging and Installation - Gunnar

Thank you!

Visit GENIVI at http://www.genivi.org or http://projects.genivi.org

Contact us: help@genivi.org