TIZEN™ DEVELOPER CONFERENCE
MAY 7–9, 2012

An Overview of Tizen Application Core Framework

Youngjoo Park

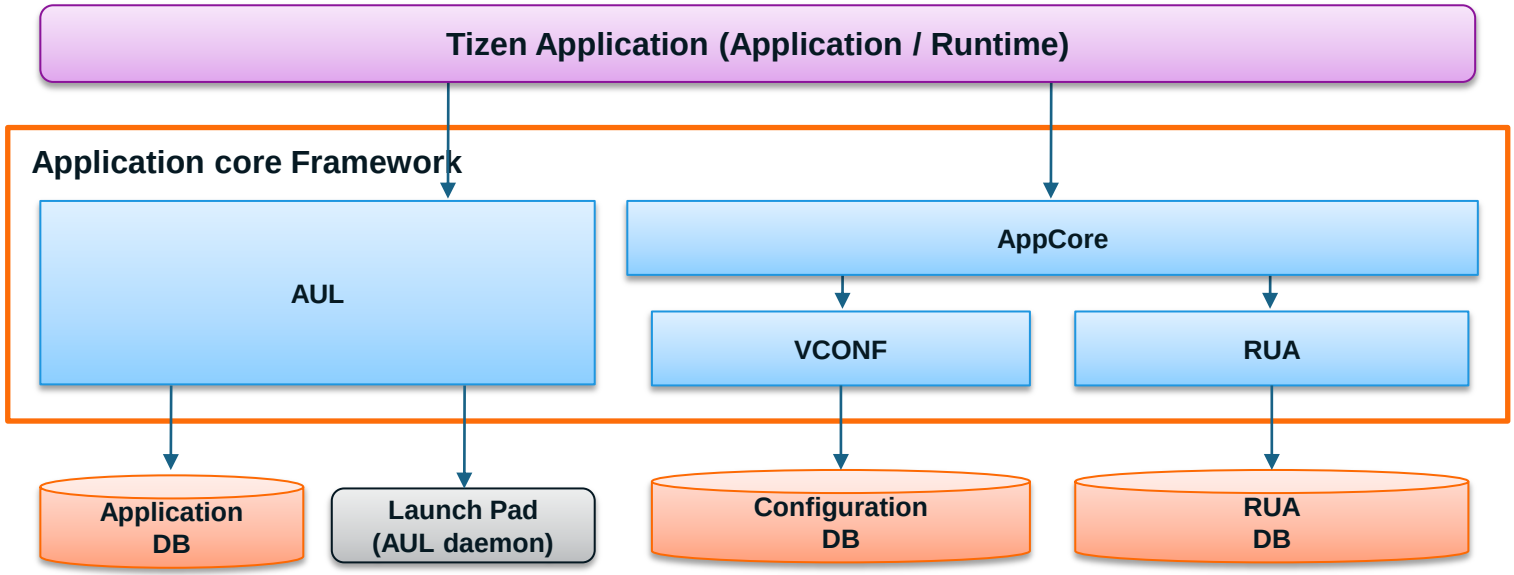
Contents

- Introduction
- Architecture
- Application Core
- Application Utility Library
- Application Service
- Package Manager
- Application Information Library
- Application Use History

Introduction

- **Application core framework provides infrastructure in which each application works seamlessly in the platform.**
  - Application life cycle management
  - Application launch service
  - System event handlers
  - Application launch history
  - Application configuration data management
  - Application installation / uninstallation

Framework Overview



*RUA : recently used application
*AUL : application utility library

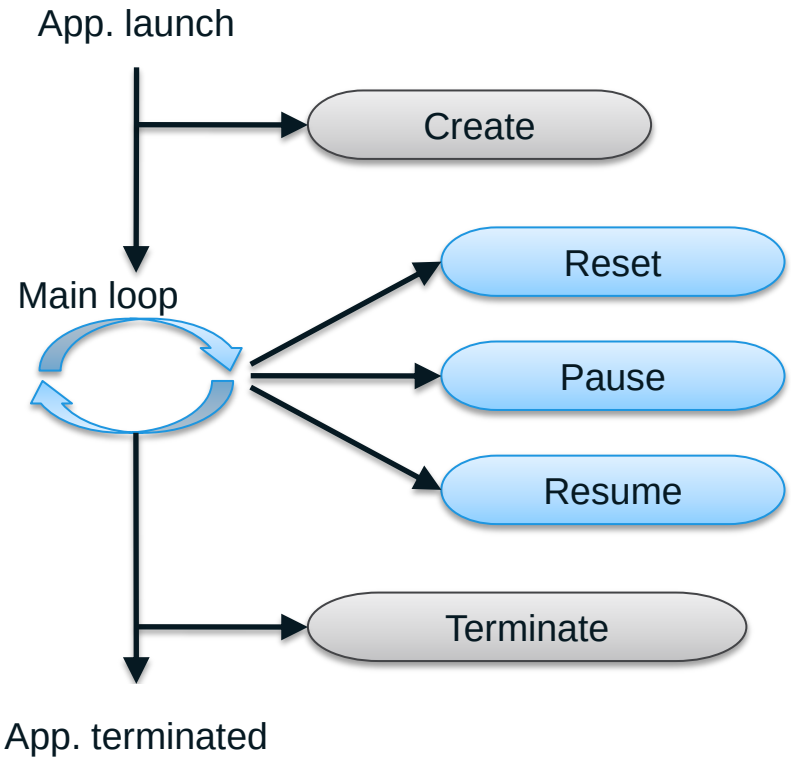
Database | Daemon | Library

Application Core: Appcore

- **Appcore is the application core handling various important events each of which application should be ware of.**

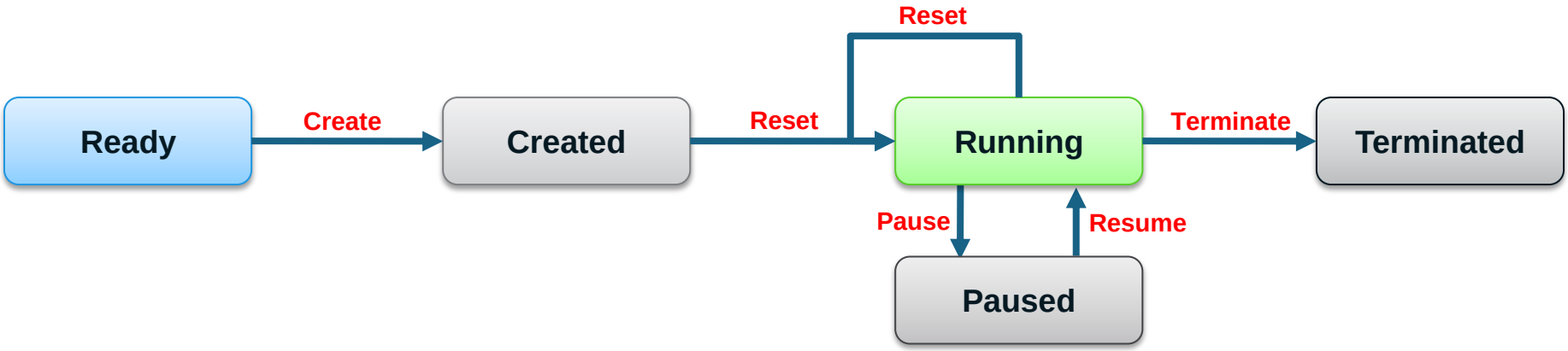
- **Using appcore, developers can**
  - Manage application life-cycle
    - Create, Reset, Pause, Resume, Terminate
  - Handle System Events
    - Low Memory
    - Low Battery
    - Screen orientation Change
    - Language & Region Change

Appcore Internal Callbacks



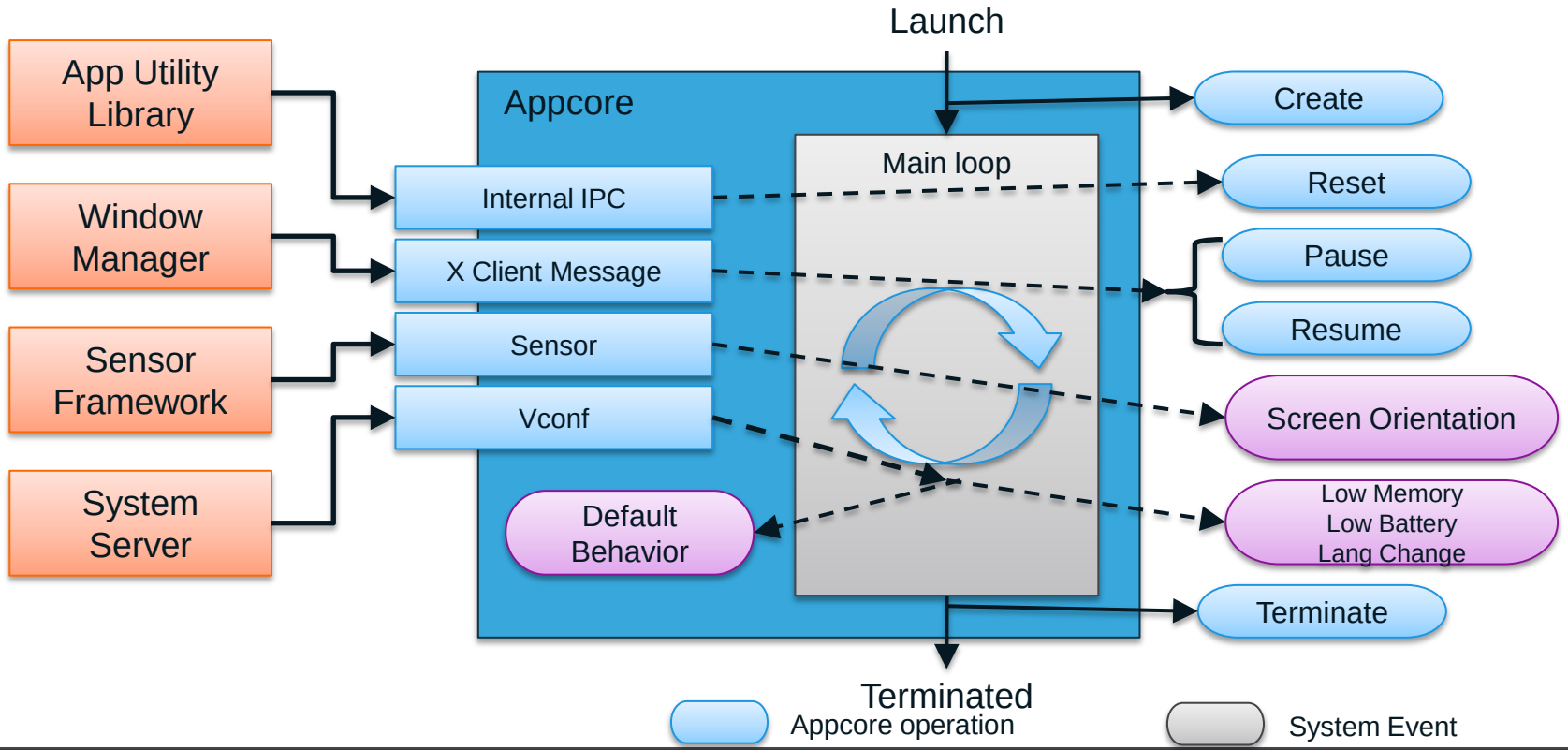
Operation	Description
CREATE	Called once before the main loop. Initialize the application such as window creation, data structure allocation, and etc.
RESET	Called at the launching and every re-launch request
PAUSE	Called when the window of the application becomes invisible. Recommend to suspend the application's tasks
RESUME	Called when the window of application becomes visible again. Resume the paused actions.
TERMINATE	Called once after the main loop. Release the resources.

Application Life Cycle: States and Transitions



States	Description
READY	The application is launched.
CREATED	The application initialized its state.
RUNNING	The application is running in the foreground and is receiving input events.
PAUSED	The application is in the background.
TERMINATED	The application was terminated.

Application Overall Structure



Application Core Framework

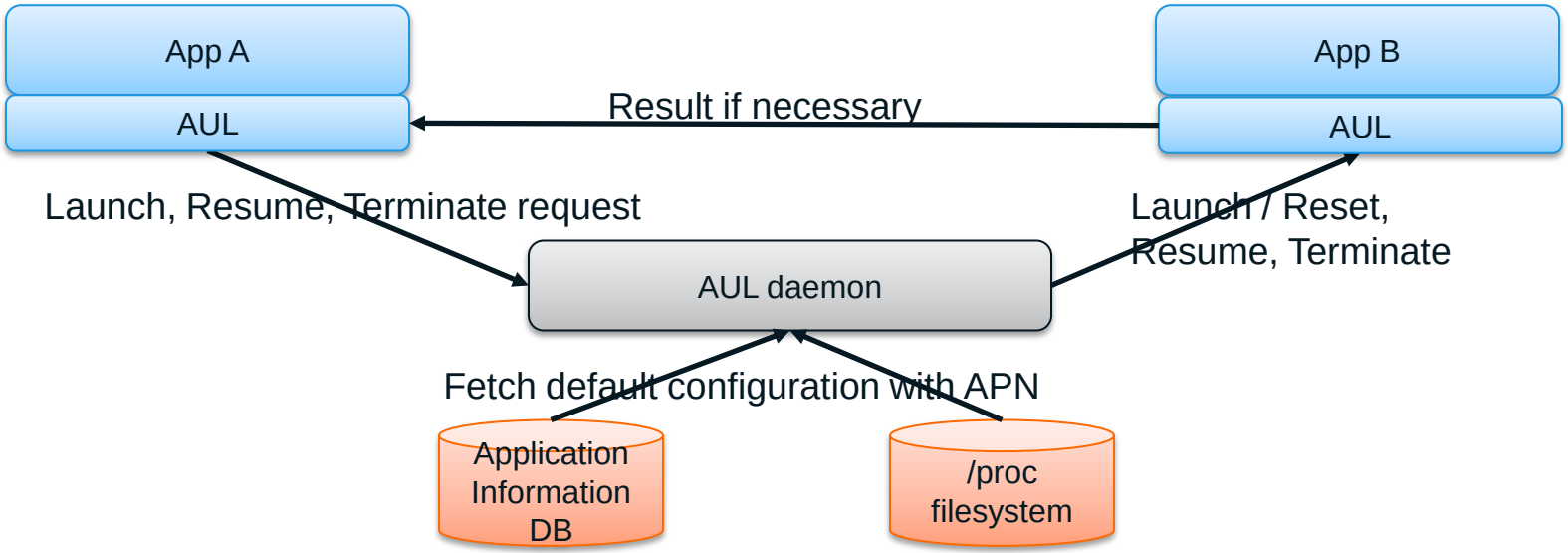
How to launch an application ?

Application Utility Library: AUL

- **Application Utility Library (AUL) provides the following features**
  - Launching/terminating applications
  - Providing running application information

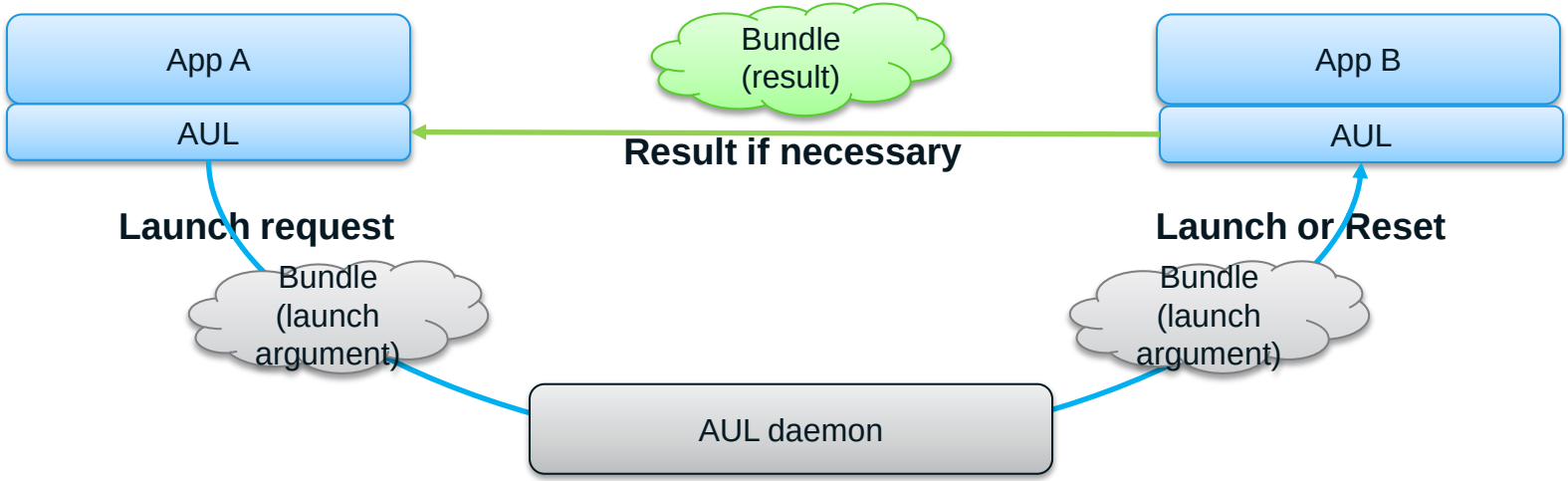
- **AUL consists of the following sub-components:**
  - AUL library: Sending/receiving requests for launching and terminating
  - AUL daemon (a.k.a. launch pad): Handling the requests

AUL Overall Architecture



Application Data Exchange

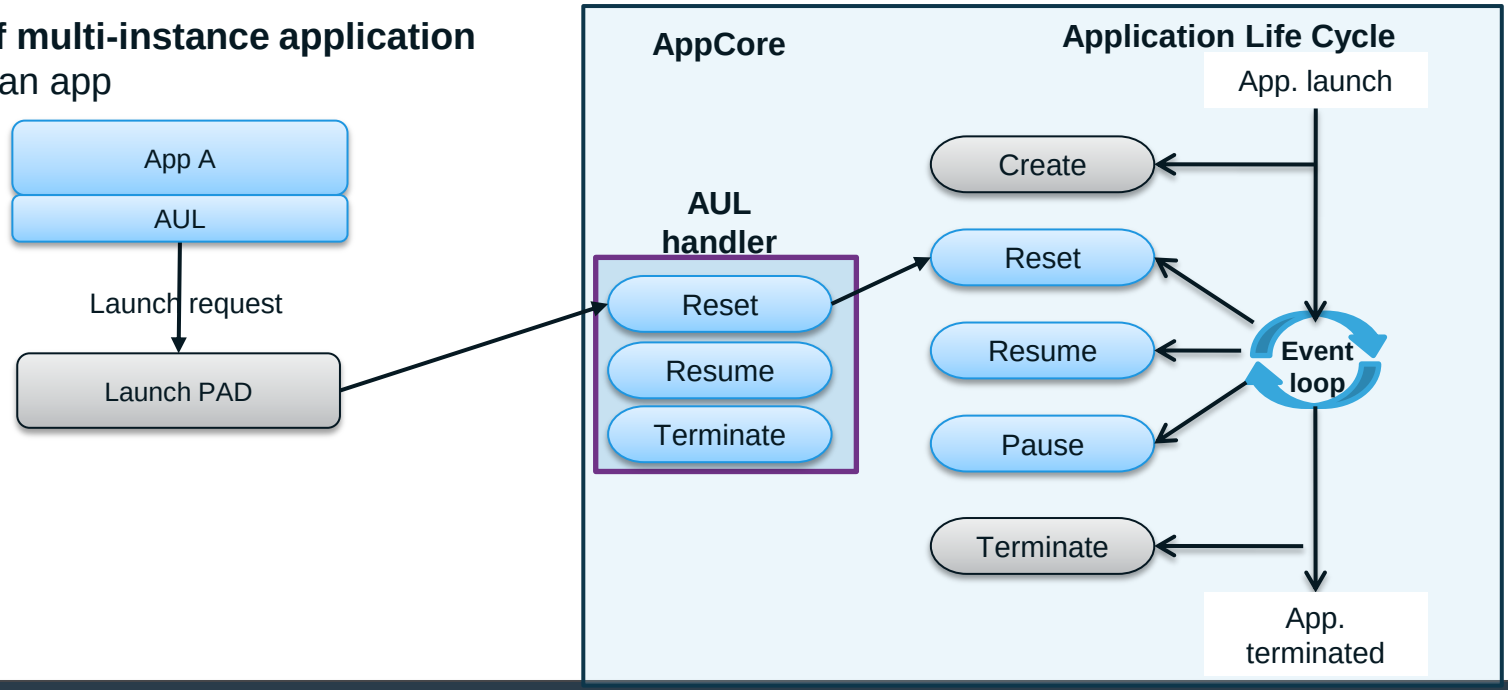
- The actual Application Data Exchange (ADE) occurs as an argument between the caller and callee, using a bundle.
- Bundle is a type of dictionary abstract data, in which information is stored as key-value pairs.
- Bundle contains information regarding the state the app should prepare.



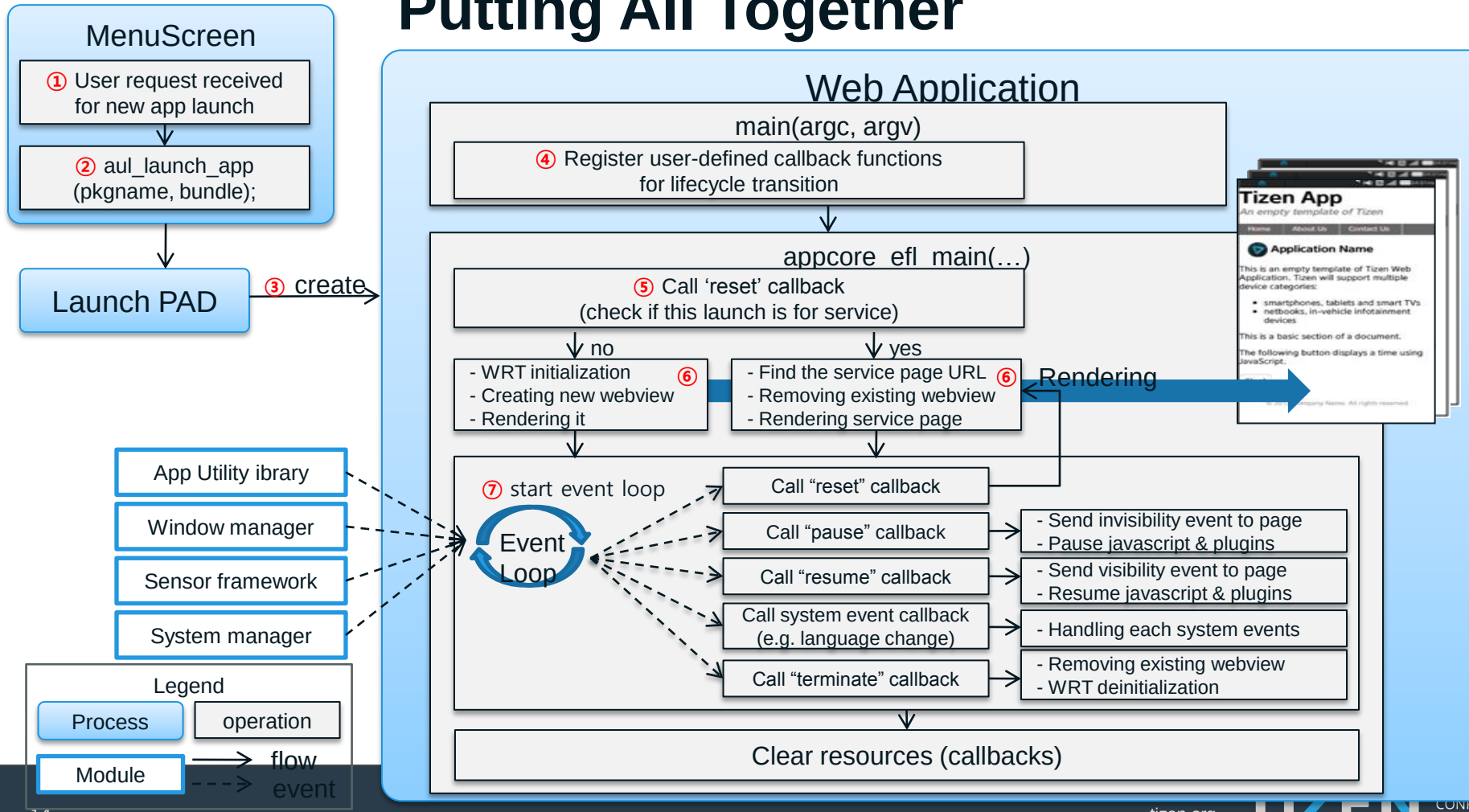
AUL: Launch

In case of single-instance application
- If app is not running, launch application
- If app is already running, send reset event to the running application

In case of multi-instance application
- launch an app



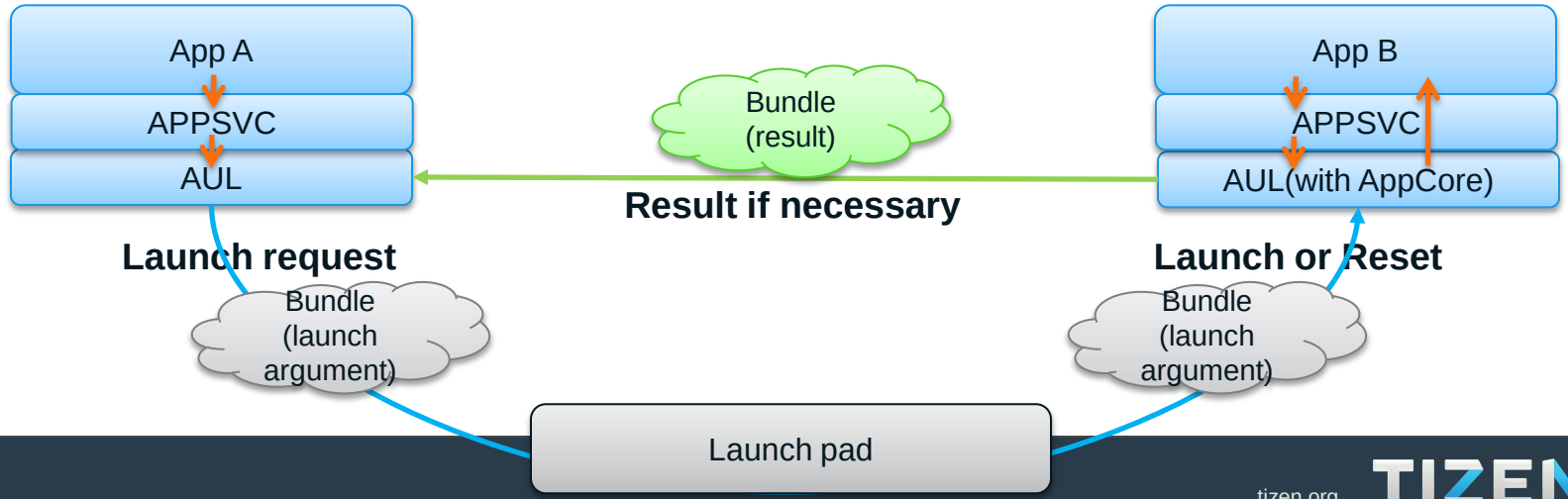
Putting All Together



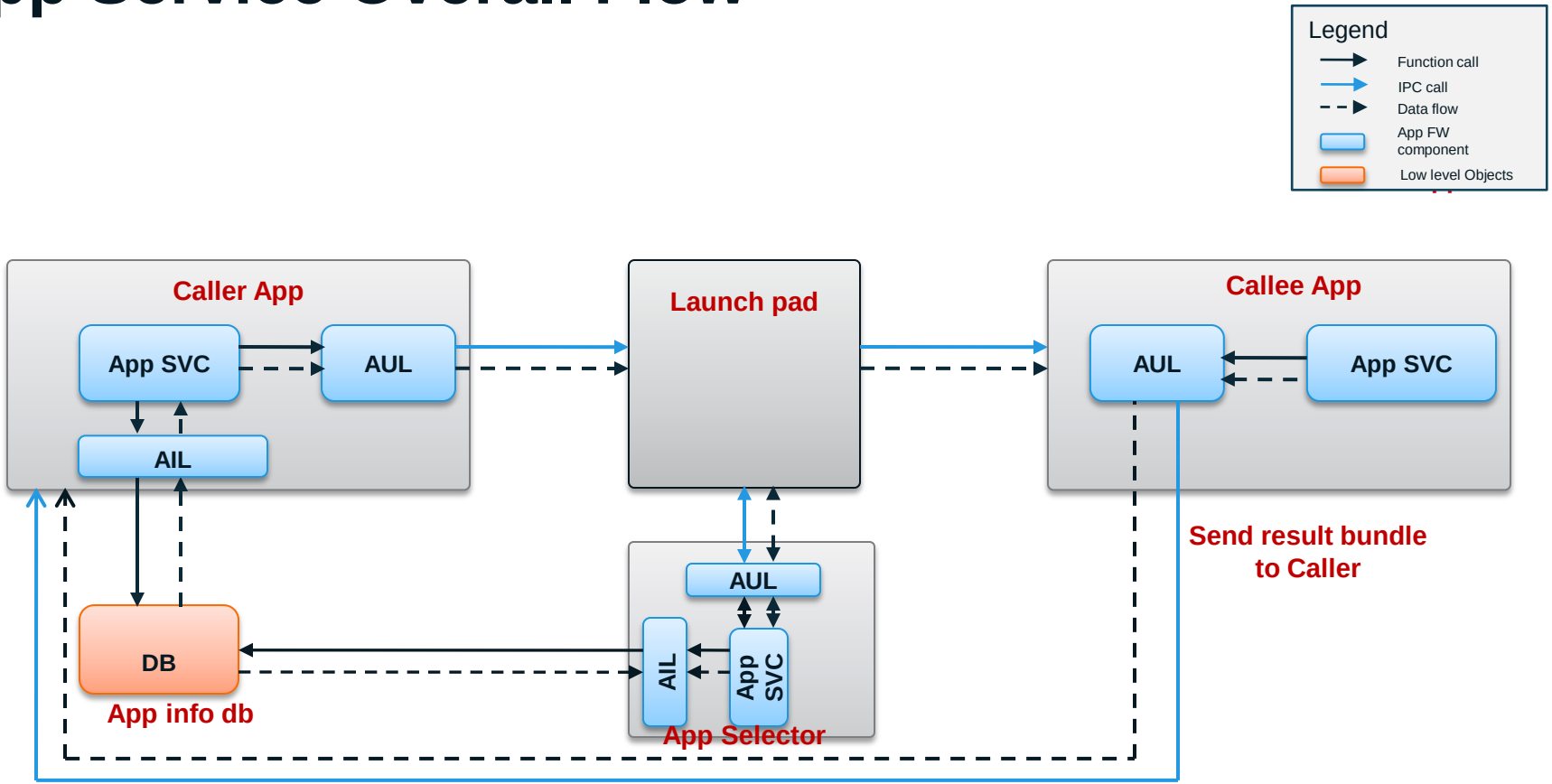
High Level Application Launch Service: App Service

App Service

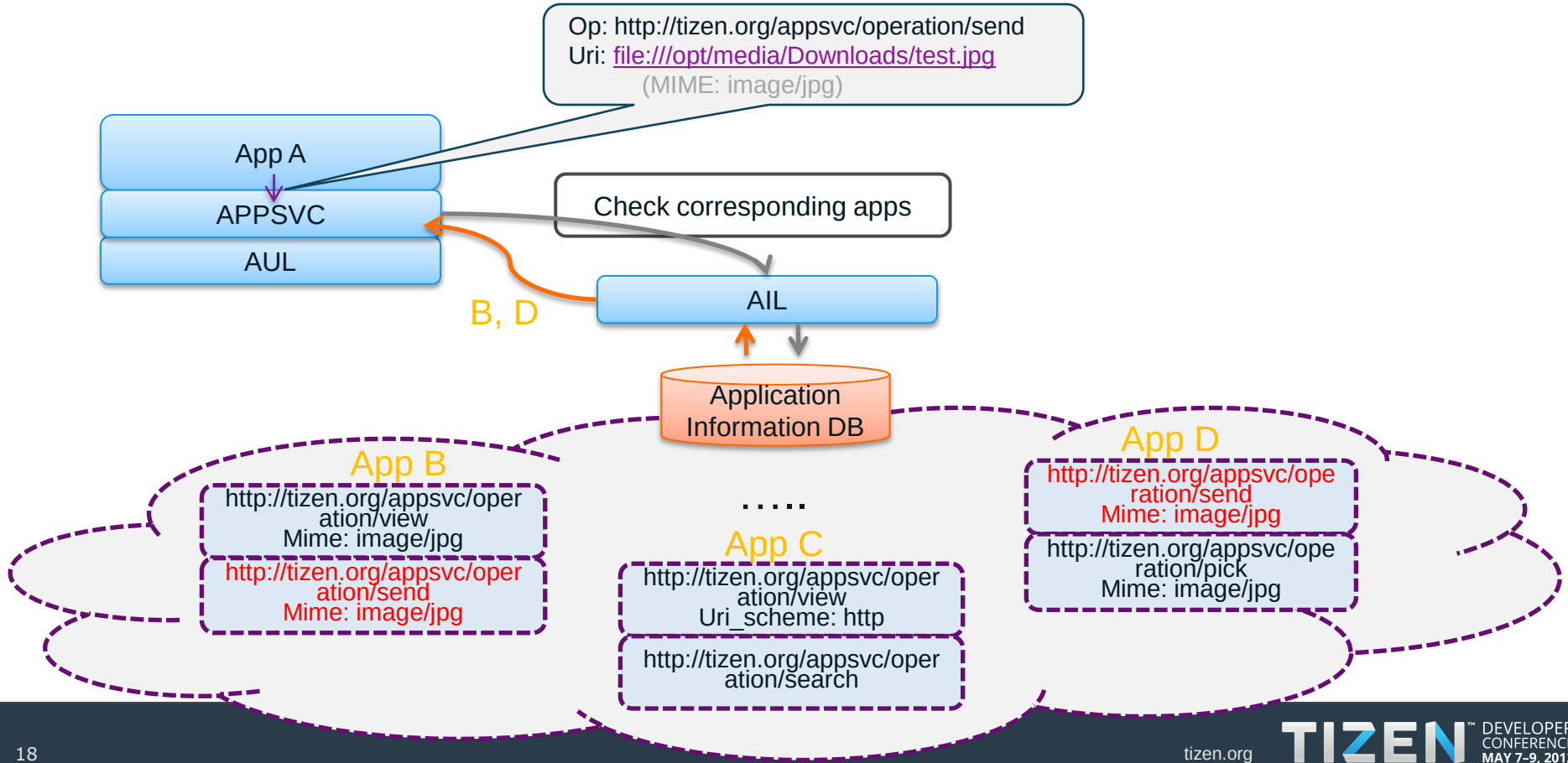
- **App service exposes general service terms, such as view, create, call, and so forth, to developers in case of launching an application with a specific feature**
  - More desirable asking image view service without knowing what image viewer apps are available.
- **Each service can be determined by given operation, URI and MIME type.**
  - Operation: expected action for the request (e.g., view, edit, call, send..)
  - URI: URI information for requested operation (e.g., http://... file://...)
  - MIME type: MIME type information for requested operation (e.g., image/jpeg)
  - Data: Extra data to launched service application(contained in a bundle packet)



App Service Overall Flow



App Service Request



Operation List of App service

Standard Operation Type (Followed Web Intents Format)	Related Data Key	description
http://tizen.org/appsvc/operation/default	N/A	Do default action of each application. It is used to launch application explicitly
http://tizen.org/appsvc/operation/edit	N/A	Provide explicit editable access to the give data URI should contain the path to edit
http://tizen.org/appsvc/operation/view	N/A	Display the specific data. Proper action should be performed by requested data. URI should contain the path to perform
http://tizen.org/appsvc/operation/pick	http://tizen.org/appsvc/data/selected	Pick items and return the path what was selected The key of return value should be set to "http://tizen.org/appsvc/data/selected"
http://tizen.org/appsvc/operation/create_content	http://tizen.org/appsvc/data/selected	Create content and return the path what was created The key of return value should be set to "http://tizen.org/appsvc/data/selected"
http://tizen.org/appsvc/operation/call	N/A	Make a phone call to a specific phone number URI should composite with "tel:" scheme (spec RFC 3966)
http://tizen.org/appsvc/operation/dial	N/A	Dial a number as a specific phone number. Dialer UI should be shown with given specific phone number URI should composite with "tel:" scheme (spec RFC 3966) or set to NULL. If URI is NULL then dialer UI except phone number should be shown
http://tizen.org/appsvc/operation/send	N/A	Deliver some data to someone else URI should contain the path to deliver
http://tizen.org/appsvc/operation/send_text	http://tizen.org/appsvc/data/text	Deliver some text data to someone else
http://tizen.org/appsvc/operation/search	http://tize.org/appsvc/data/keyword	Perform a search The searching keywords should contain to data with key "http://tizen.org/appsvc/data/keyword"

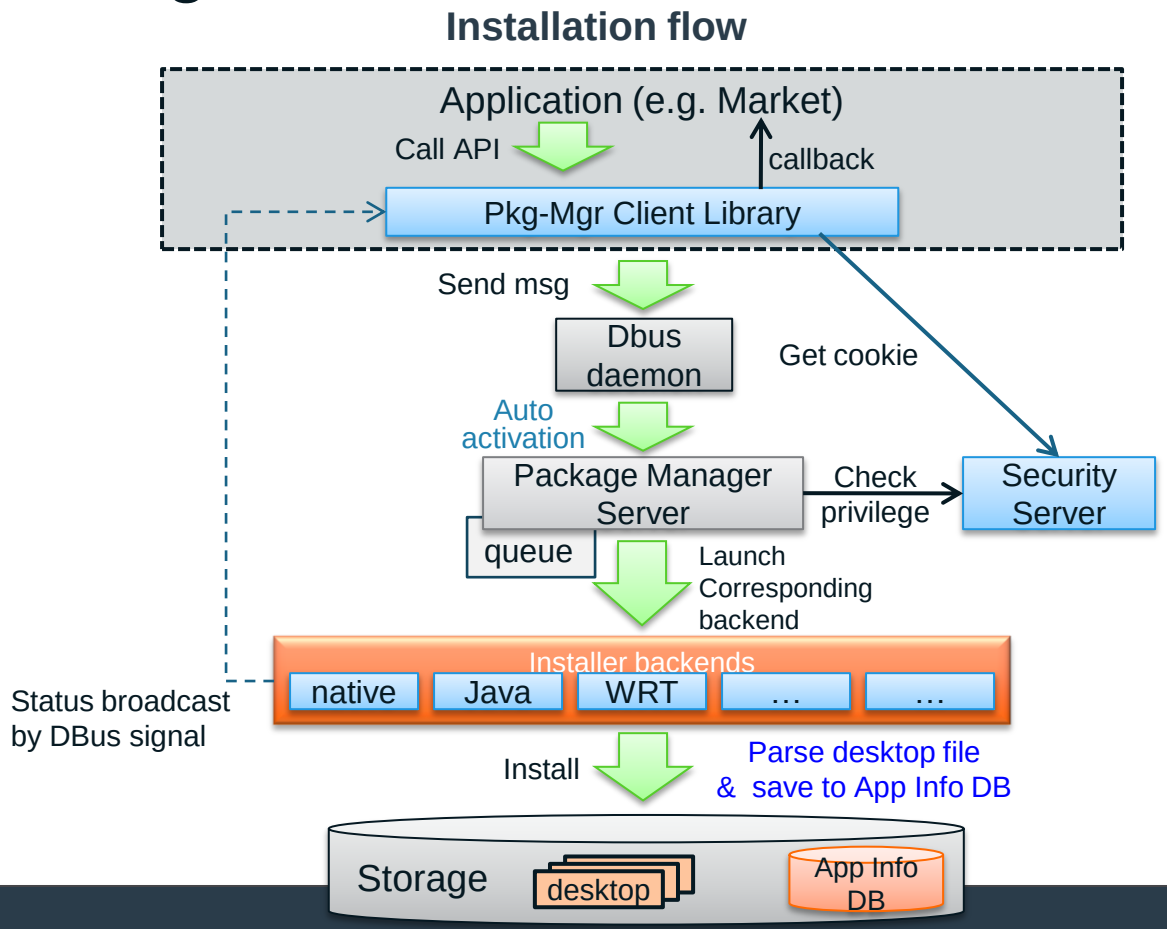
Application Installation

Package Manager

- **Package manager is responsible for installing, upgrading and uninstalling of applications and storing their information.**

- **Expandable structure to support various types of applications**
  - Designated installation modules can be added to the manager
    - Web app, native app, java app, and so forth

Package Manager

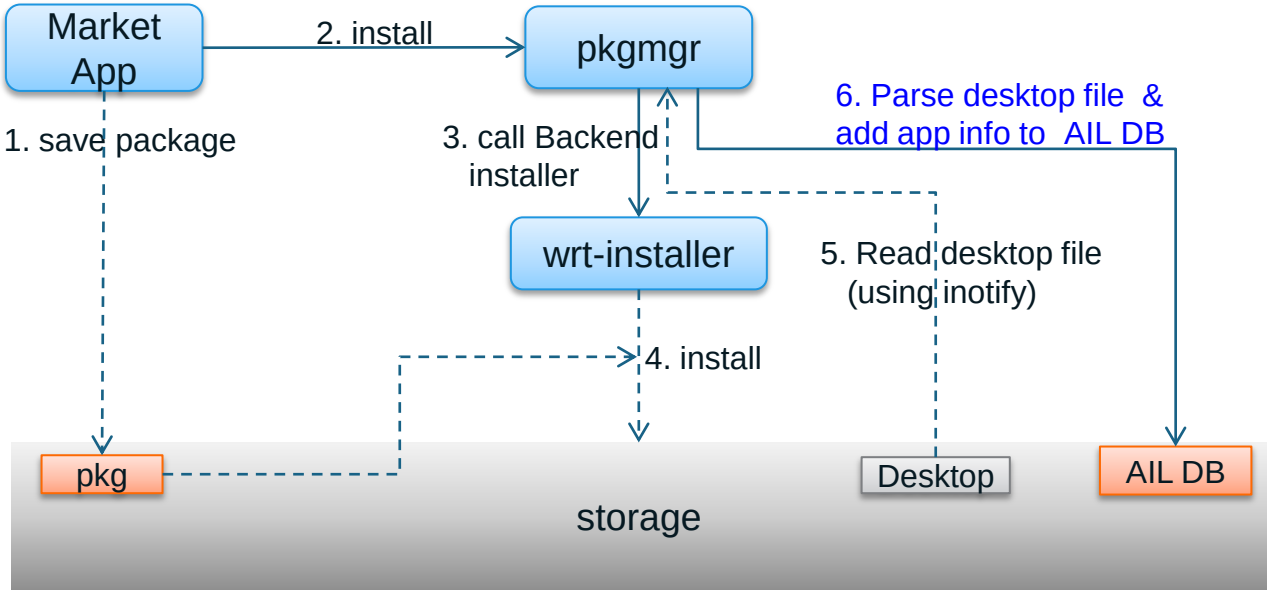


Application Information Management

Application Information Library: AIL

- **AIL is the library providing functionalities to execute application information-related tasks.**
- **AIL provides the following features**
  - Adding, updating and removing application information.
  - Managing application's information – application name, type, icon path, exec path, etc.
  - Retrieving an application list which meets a given filter.

AIL: Adding Application Information



Application Use History

- **Recently Used Application (RUA) logs application use history.**
- **When an application is launched, launch pad updates the history and task manager can either get or clear the history.**
- **RUA also provides below information regarding launching:**
  - package name
  - launch time
  - application path
  - application launch argument

Questions?