

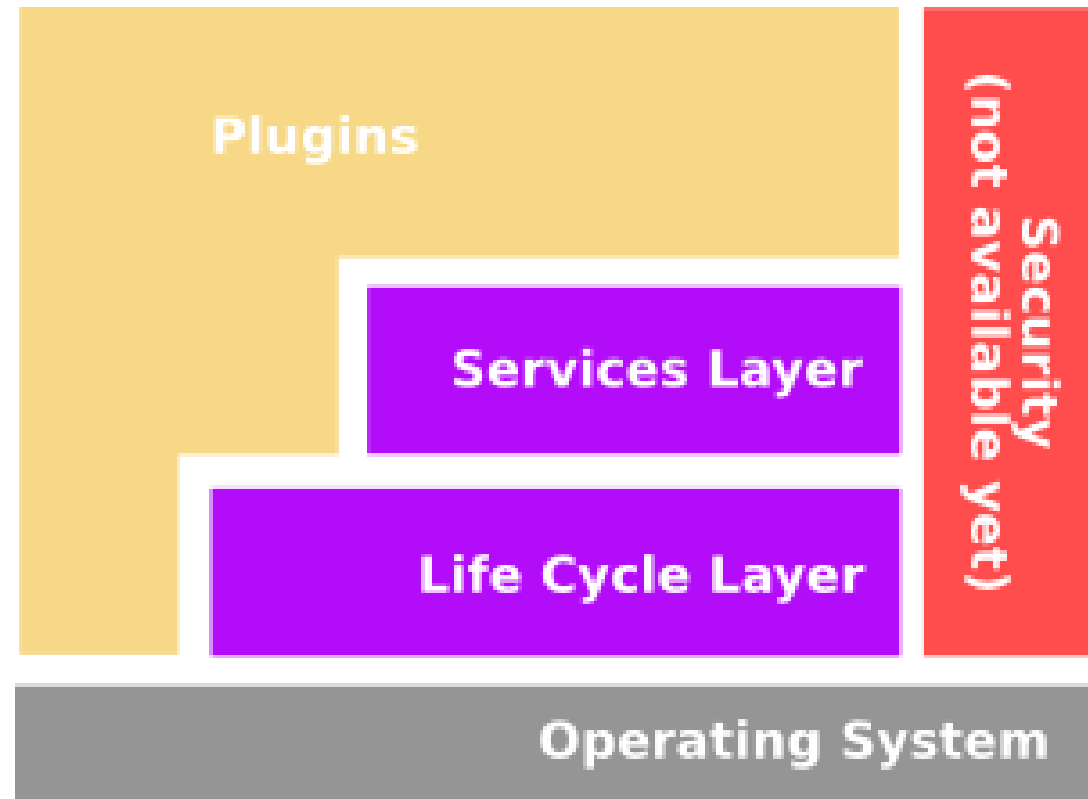
•14/11/11

# CTK Plugin Framework

Sascha Zelzer

# Plugin Framework

- Dynamic Plugin Framework (based on OSGi)
- Enables service oriented architectures
- CTK provides basic plugins for distributed/large-scale applications



# OSGi

- The OSGi Alliance is a non-profit corporation founded in March 1999.
- More than 35 companies from various areas
- Roots in embedded systems
- The OSGi specification is at Release 4 with numerous implementations in Java
- Specification for the core framework and a compendium of service interfaces

# Layers

- Plugins

Plugins are the CTK components created by the developers.

- Services Layer

Connects plugins in a dynamic way by offering a publish-find-bind model for C++ objects.

- Life Cycle Layer

The API to install, start, stop, update, and uninstall plugins.

- Security

Handles security aspects (not available yet)

# Specifications

- OSGi Core Specifications are small
- OSGi Service Compendium defines many optional services:

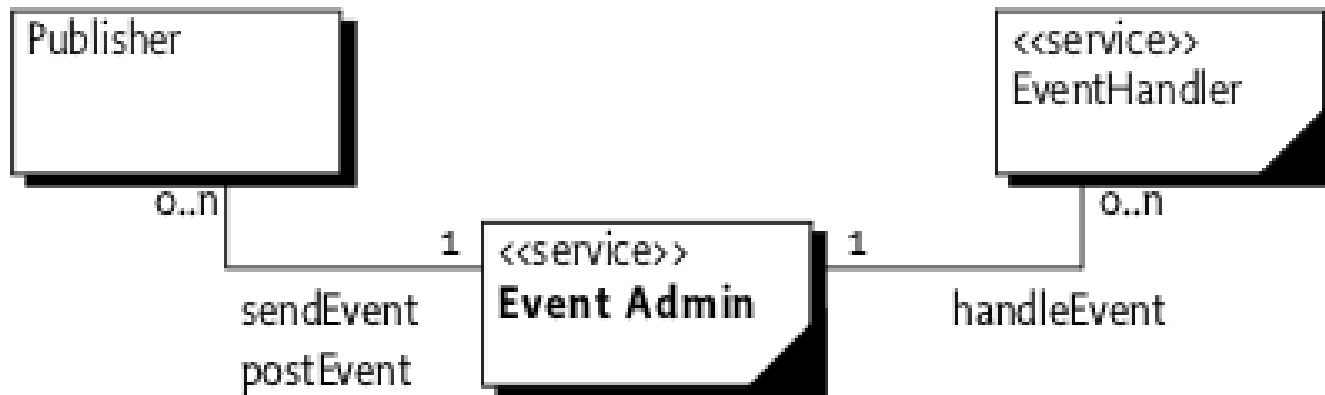
▶ 13 Remote Services	17
▶ 101 Log Service Specification	31
▶ 102 Http Service Specification	43
▶ 103 Device Access Specification	61
▶ 104 Configuration Admin Service Specification	91
▶ 105 Metatype Service Specification	137
▶ 106 Preferences Service Specification	161
▶ 107 User Admin Service Specification	181
▶ 108 Wire Admin Service Specification	205
▶ 109 IO Connector Service Specification	249
▶ 110 Initial Provisioning	259
▶ 111 UPnP™ Device Service Specification	281
▶ 112 Declarative Services Specification	309
▶ 113 Event Admin Service Specification	355
▶ 114 Deployment Admin Specification	375
▶ 115 Auto Configuration Specification	433
▶ 116 Application Admin Specification	441

## Implemented OSGi specifications in CTK

- Log Service Specification  
Provides a general purpose message logger.
- Metatype Service Specification  
Provides a unified way to describe metadata about services.
- Configuration Admin Service Specification  
Allows to set the configuration information of deployed plugins.
- Event Admin Service Specification  
Inter-plugin communication mechanism based on a event publish and subscribe model.

## Event Admin

- Event publisher: sends events related to a specific topic
- Event handler: expresses interest in one or more topics



## Features

- Synchronous or asynchronous event delivery
- Event from different threads are sent in parallel
- Event handler blacklisting

# Using the Plugin Framework

Programming using the CTK Plugin Framework means:

1. Create plug-ins
2. Use services
3. Provide services
4. Deploy plug-ins in a CTK-based environment

## What do we get?

- Stronger encapsulation & loose coupling
- Live updates
- Exchangeable software modules

## What does it cost?

- Code for tracking services: they can come and go as they want
- Little overhead for the plug-in management



## General Benefits

- Reduced Complexity
- Reuse
- Real World
- Easy Deployment
- Dynamic Updates
- Adaptive
- Transparency
- Versioning
- Simple
- Lazy
- Humble
- Non Intrusive

## Integration is easy

- The framework is easy to start and to embed
- Clear separation between inside and outside world

# Use Case – DICOM Application Hosting

