Design of Portable Waveform SW Applications

Daniele Olmisani – SELEX Elsag Spa
daniele.olmisani@selexelsag.com

WInnForum Webinar Series:
The Software Communications Architecture
16/11/2011

SELEX ELSAG
Secure Networking Solutions
A Finmeccanica Company

Tutorial Overview

SELEX ELSAG
A Finmeccanica Company

- This tutorial provides an overview on how the **distributed application design concepts** may be applied in order to enable a better portability of design and implementation of WF Applications.

- The WF Application Portability is a Design achievement

- Application Frameworks (e.g. JTRS SCA specification) provide the definition needed in order to enable the portability of Waveform applications among several Radio Platforms.

Tutorial Workflow

SELEX ELSAG
A Finmeccanica Company

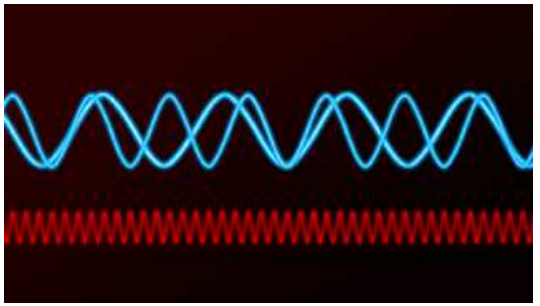
- Tutorial Overview
- Radio WFs vs. WF Applications
- WF Application as Distributed Application
- WF Applications and Radio Platforms
- Radio Platforms and Application Frameworks
- WF Application Features
- WF Design Patterns
  - Architectural Patterns
  - Interaction Patterns
- The WF Design: putting all together
- Highlights: WF Application Components
- Conclusions

Radio WF vs. WF Application

- **Radio Waveform**

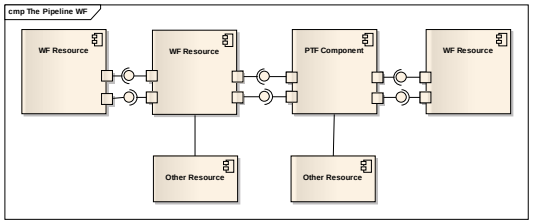
  In this case the term **Waveform** means the shape and form of a signal such as a wave moving in a physical medium.

  The modulation of the waveform make possible the transmission of information by the means of the physical medium.



- **Waveform Application**

  In this case, the term **Waveform** means a SW program that uses the processing and RF capabilities of a radio equipment in order to implement a Radio Waveform



WF Application as Distributed Application

SELEX ELSAG
A Finmeccanica Company

- **Radio Platforms**

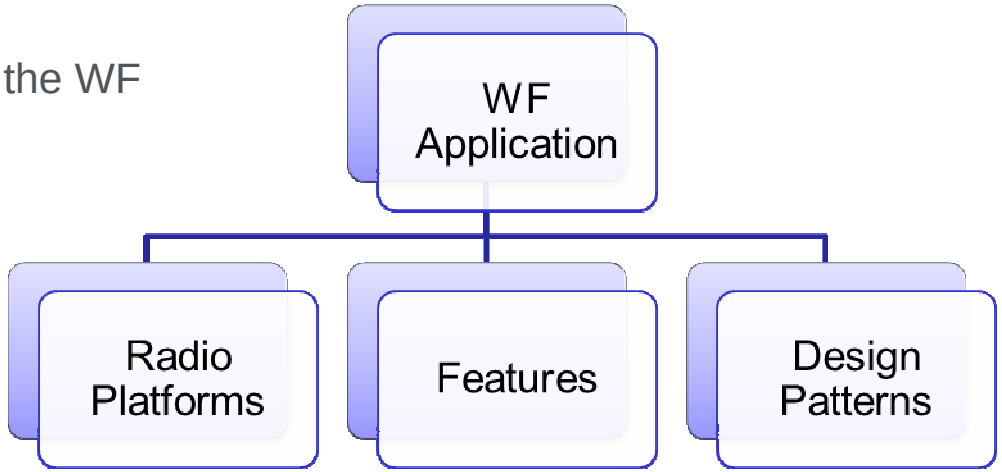
  The Radio Platforms provide reusable HW and SW capabilities used by the WF application in order to implement a specific Radio WF

- **Design Patterns**

  A Design Pattern is a general reusable solution to a commonly occurring problem within a given context

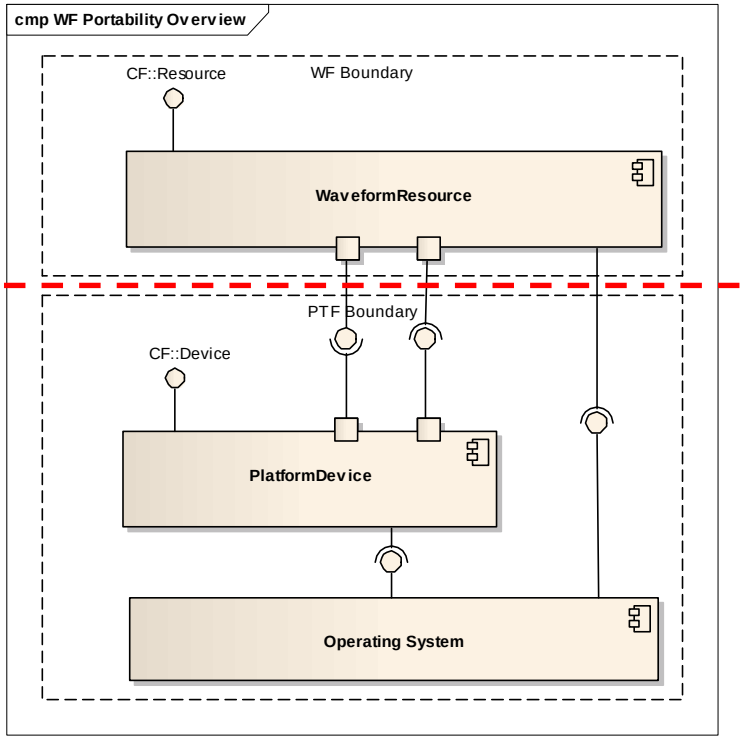
- **WF Features**

  The set of capabilities implemented by the WF Application



WF Applications and Radio Platforms

The **WF Portability** concept is based on the separation of the **WF Applications** from the **Radio Platforms**



Radio Platform

- provides HW and SW capabilities
- allows the access by the means of standard APIs
- allows the management of WFs by the adoption of an Application Framework
- defines a set of features identified in profiles

The JTRS SCA is an example of specification that provides a model of Radio Platforms (Application Framework, Middleware and APIs)

Radio Platforms and the Application Framework

SELEX ELSAG
A Finmeccanica Company

The Application Framework (such as a Web/SOA Application Container) is used in order to implement the standard structure of a WF Application. It may be composed by the following elements:

- **Control Entities**

  The Control Entities are specialized components that allow the deployment, connection and instantiation of the PTF capabilities and WF applications

- **Connectivity Middleware**

  The Connectivity Middleware allows the communication among SW components co-located or distributed on different processing elements (e.g. GPPs, DSPs, FPGAs). It may be very simple (e.g. defines only generic application profile) or very complex (e.g. identifies a transport protocol, data serialization mechanisms, addressing and related services)

- **Standard Capabilities and APIs**

  They provide a standardized access to well defined basic functionalities of the platform (i.e. application management and data processing)

Radio Platforms and Connectivity Middleware

SELEX ELSAG
A Finmeccanica Company

- The Middleware is the Architectural layers that provides a link between separate software applications or separated part of the same application (e.g. components executed on different processing elements, such as GPPs, DSPs and FPGAs)

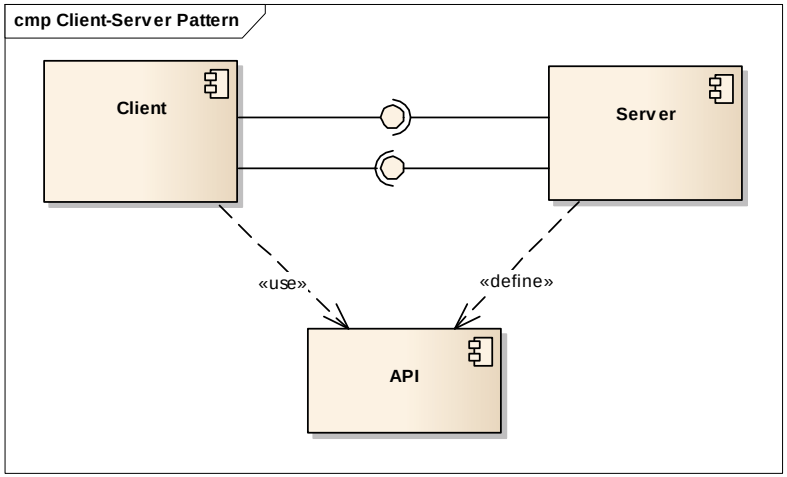
- There are different Middleware solutions, but usually in every solution is at least needed that:
  - a service Consumer is able to identify the requested service Provider (**addressing capabilities**)
  - The format of the information transmitted between Consumer and Provider is independent from the specific implementation of the two end points (**marshaling/demarshaling rules**)
  - the communication between Consumer and Provider is independent form the specific deployment of the two end points (**deployment abstraction**)

Radio Platforms and Common APIs

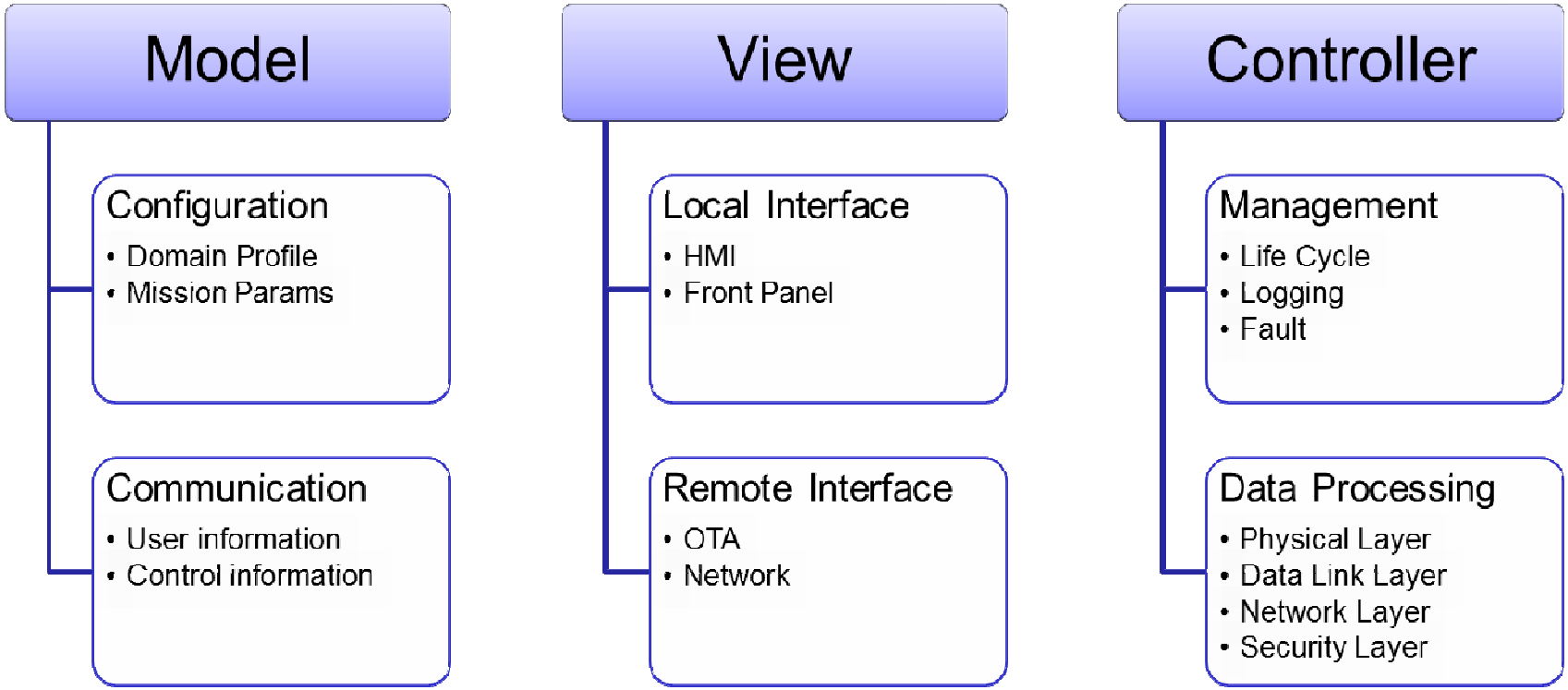
SELEX ELSAG
A Finmeccanica Company

- The Radio Platform provides different capabilities that may be used by the WF Application in order to implement specific Radio WF Features

- The definition standard Radio Platforms APIs allows
  - a clear allocation of WF Features implementation between WF Application and WF Application
  - the re-usage of the high-level design and components decomposition of the WF Application



WF Application Features



WF Design Patterns

SELEX ELSAG
A Finmeccanica Company

WF Architectural Design Patterns

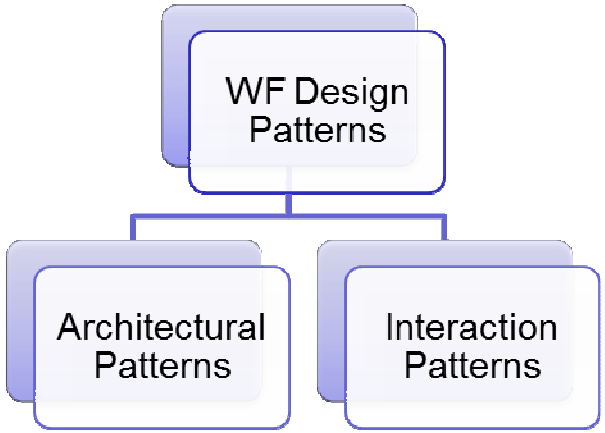
Design Patterns related to the architecture of the whole WF application in relation with:

- the platform architecture
- the application framework
- internal WF components

WF Interaction Design Patterns

Design Patterns related to the interactions of the single WF components with:

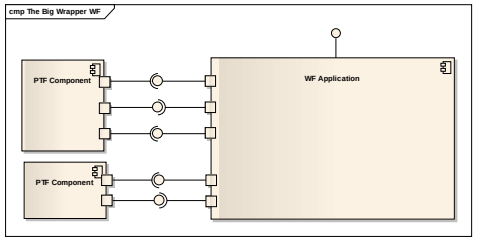
- component internal processing logic
- other WF components
- other platform components

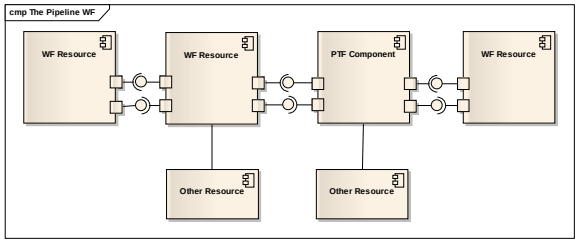


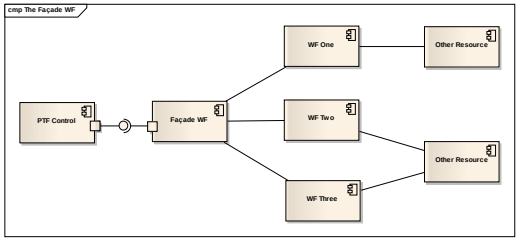
WF Architectural Design Patterns

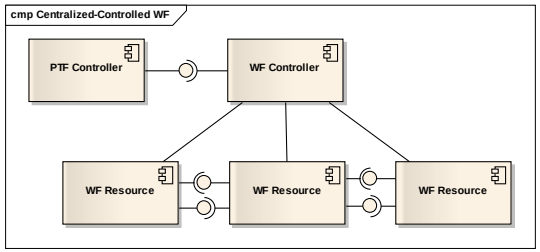
SELEX ELSAG
A Finmeccanica Company

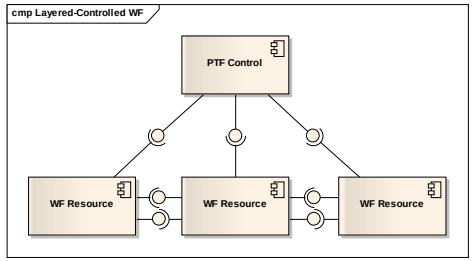
1. The Big Wrapper WF
2. The Pipeline WF
3. The Façade WF
4. The Centralized-Controlled WF
5. The Layered-Controlled WF











Pattern: The Big Wrapper WF

SELEX ELSAG
A Finmeccanica Company

Description

The WF is designed as an unique SW component that incorporates:

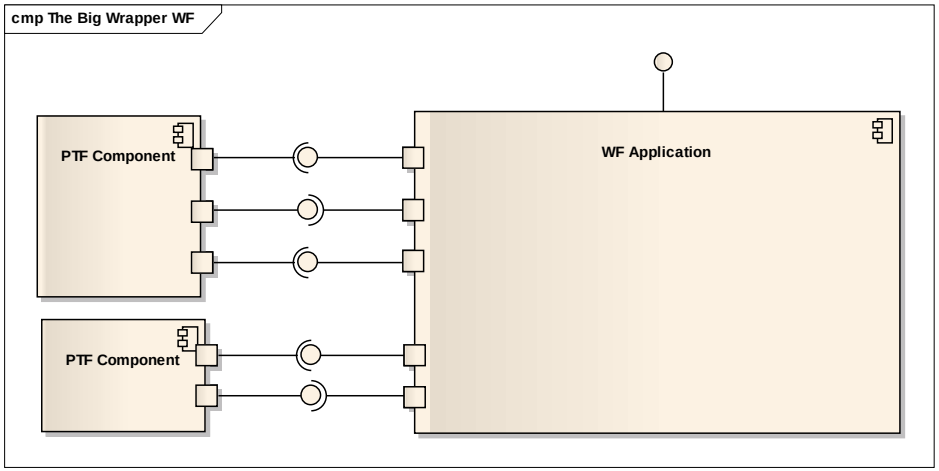
- *all the WF processing logic,*
- *the WF internal interactions*
- *and the interactions with the platform components*

Pros

- Useful in porting legacy implementation of Radio WF: the whole application is wrapped by the WF container logic that interacts with the Application Framework and the other Platform capabilities.

Cons

- Poor modularity: limitation on portability and extensibilty of the WF applilcation



Pattern: The Pipeline WF

SELEX ELSAG
A Finmeccanica Company

Description

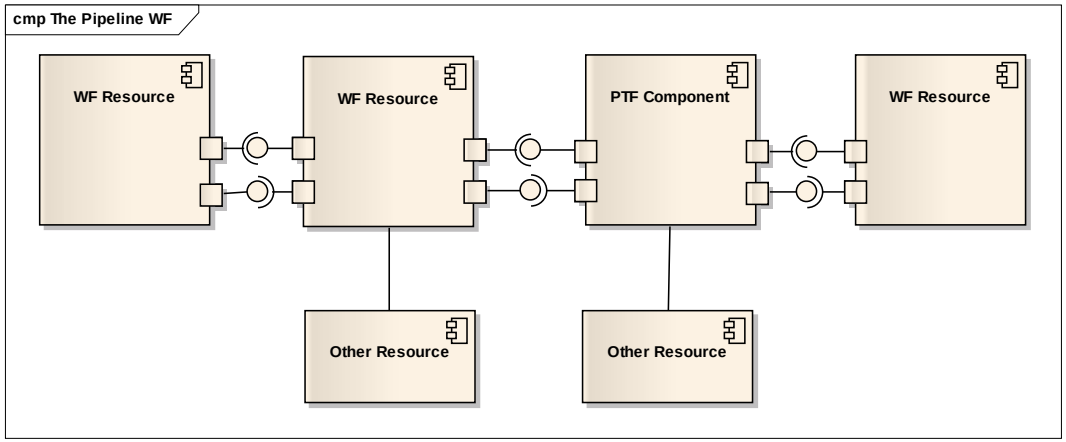
The WF is designed as a sequence of processing components; each component (WF or PTF component) exchanges information only with the adjacent components.

Pros

- Increased portability: each component is self contained with well defined interfaces
- Increased Modularity with improved reuse and extensibility

Cons

- Heavy infrastructure: the improved modularity may have impacts on performances and size of the binary code



Pattern: The Façade WF

Description

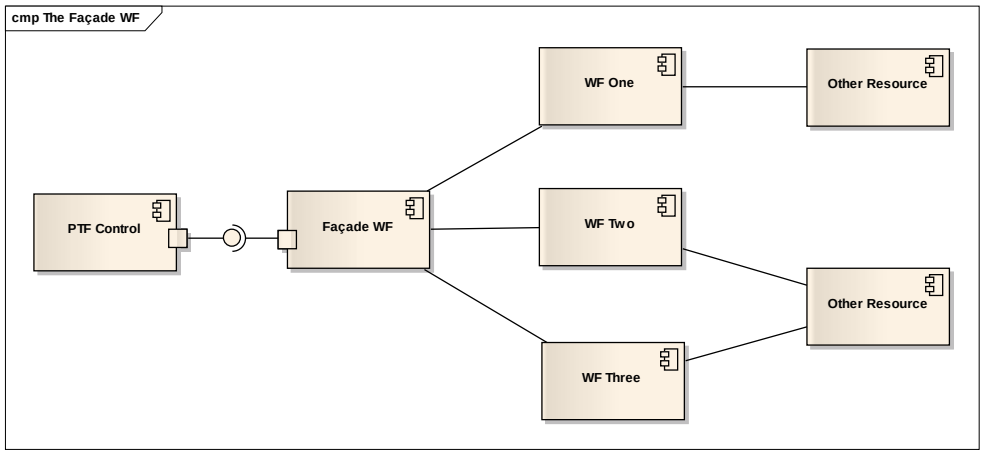
The WF is composed of several sub-applications; each application implements a specific Radio WF; the façade application is in charge of the coordination and configuration of the other applications.

Pros

- Reuse of existing applications

Cons

- The control of each sub-waveform shall be adapted to the needs of the other applications
- The PTF resources should be shared among different applications



Pattern: The Centralized-Controlled WF

SELEX ELSAG
A Finmeccanica Company

Description

The WF control is centralized in a specialized component (e.g. the AssemblyController in JTRS SCA specification). This control component is in charge of the configuration and coordination of the rest of the application:

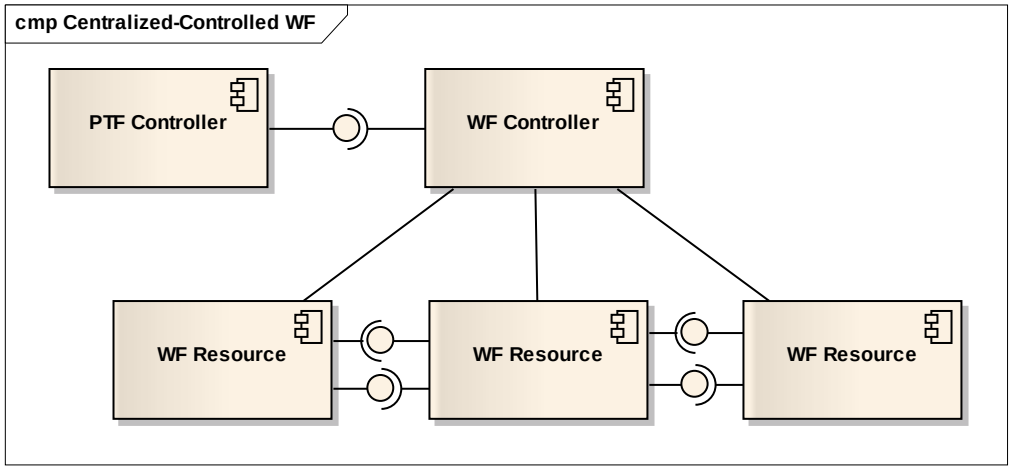
- *The PTF Controller communicates only with the WF Controller which configures and delegates processing operations to the other WF components*
- *The WF components communicate with the PTF by the means of the WF Controller component in order to notify configuration changes and results of processing operations*

Pros

- Well defined global state of the WF Application
- Possibility to implement a centralized logic guard on the WF control

Cons

- Less modularity: each processing componet depends on the WF Controller



Pattern: The Layered-Contolled WF

SELEX ELSAG
A Finmeccanica Company

Description

The WF control is distributed among the WF components. Each component is in charge of its proper configuration and coordintation with the platform and the rest of the application:

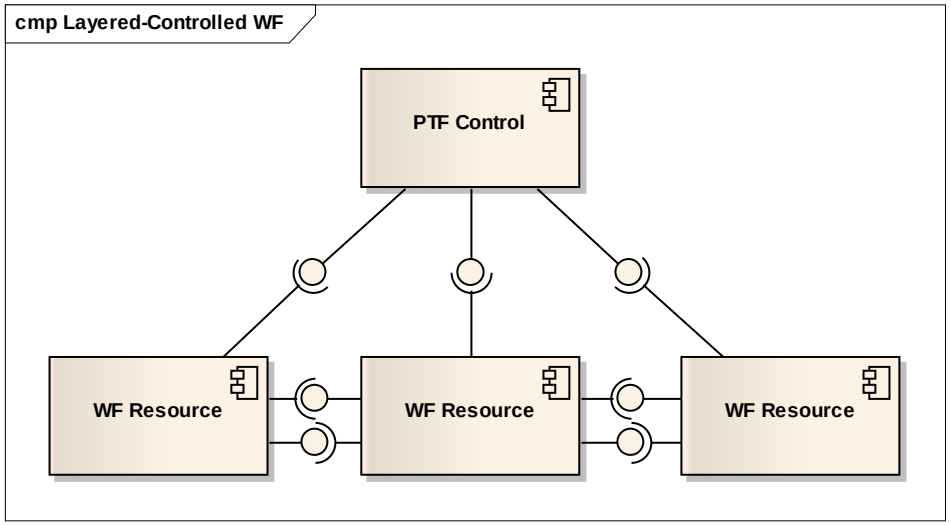
- *notifies the PTF control on changes of configuration parameters and results of processing operations*
- *when needed, requests changes of configuration directly to the related WF component*

Pros

- Increased modularity: each WF component is self-contained

Cons

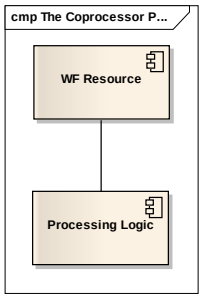
- The global state of the WF Application is distributed among several components

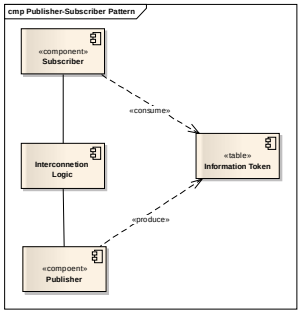


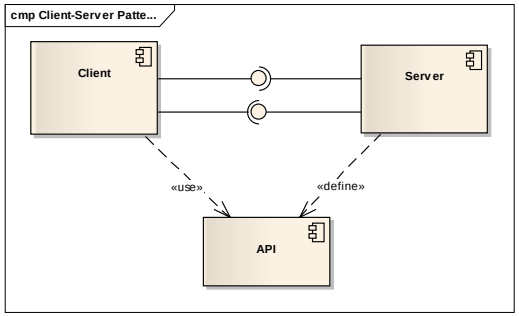
WF Interaction Design Patterns

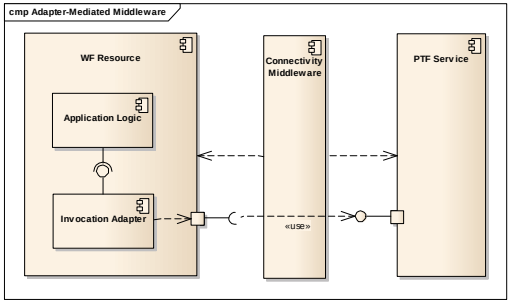
SELEX ELSAG
A Finmeccanica Company

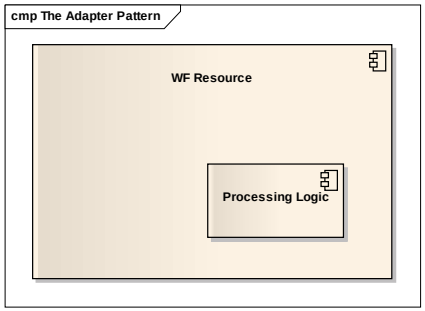
1. The Client-Server Pattern
2. The Publisher Subscriber Pattern
3. The Adapter-Mediated Middleware Pattern
4. The Coprocessor Pattern
5. The Embedded-blog Pattern











Pattern: The Client-Server

SELEX ELSAG
A Finmeccanica Company

Description

The communication between two different components (the client and the server) is performed by the means of a well defined API (e.g. the Application Framework API).

The client and server componets are designed as they are incorporated in the same SW application.

Usually the communication is end-to-end

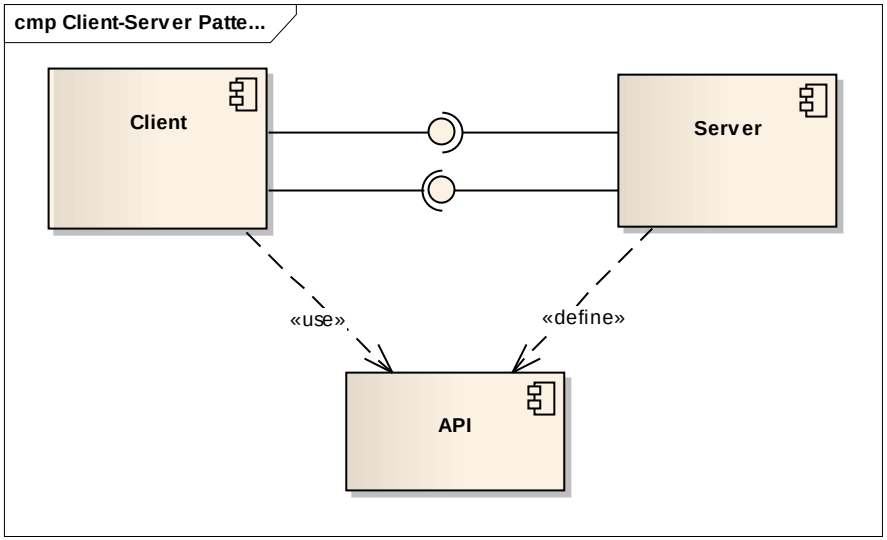
The client and server componets can be co-located or distributed on different processing elements (by the means of the Connectivity Middleware)

Pros

- Clear interfaces: the usage of domain-specific APIs allows the

Cons

- Less modularity: the client-server components are strictly related and shares the same communication needs



Pattern: The Publisher-Subscriber

SELEX ELSAG
A Finmeccanica Company

Description

The communication among several components (usually one publisher and one or more subscriber) is performed by the exchange of data messages.

Data messages are exchanged by the means of a specialized API (e.g. property configuration interface) or a logical bus (e.g. CORBA Event Channel)

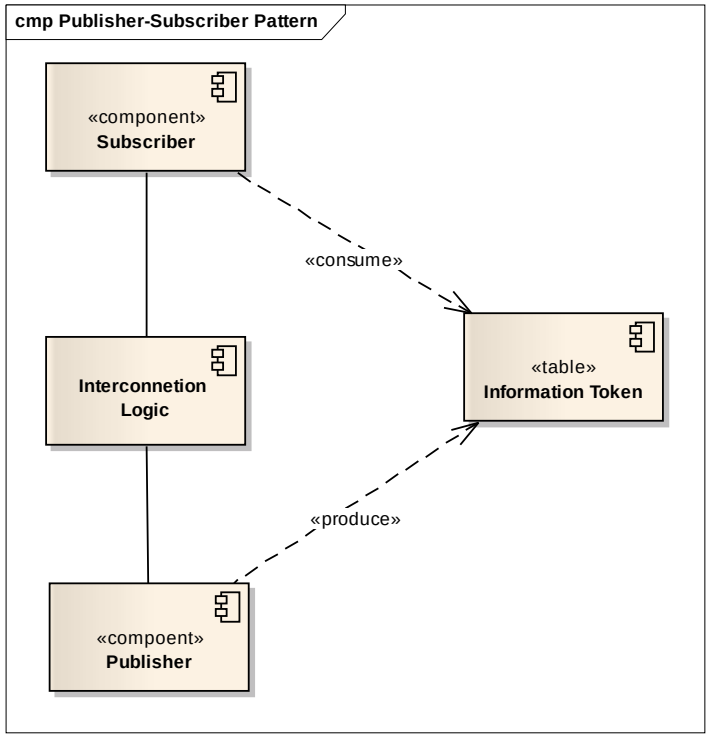
The publisher and subscriber components are sharing only the structure of the message

Pros

- Increased modularity: the publisher and subscriber logic are decoupled by a standard communication means

Cons

- Message-oriented communication: some infrastructure needed (message queues, mutex, ...) in order to support the application communication constraints



Pattern: The Adapter-Mediated Middleware

SELEX ELSAG
A Finmeccanica Company

Description

The communication between a WF component and the Connectivity Middleware is mediated by specific adapter logic. In this way the core logic of the component is well separated from the specific middleware issues.

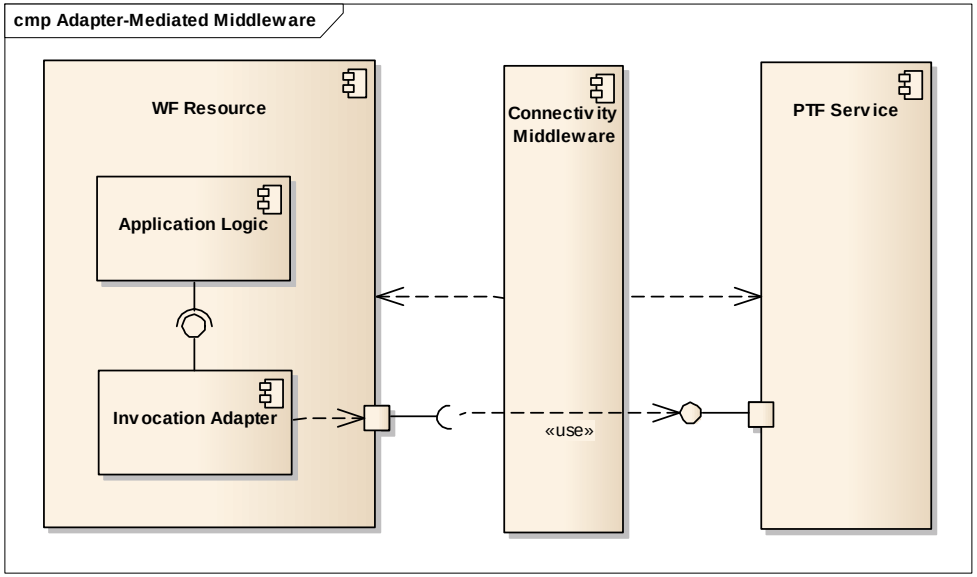
The Invocation Adapter logic should be internal to the component (see the figure) or external. In the last case the same adaptation logic may be shared among several components (e.g. SOA-based application containers)

Pros

- Increased portability: the component may be easily adapted to different Connectivity Middlewares or ported in legacy environments

Cons

- An adaptation API is needed in order to allo the communication between Application Logic and Invocation Logic
- The adding of the adaptation layer may result in a loss of performances



Pattern: The Coprocessor

SELEX ELSAG
A Finmeccanica Company

Description

The communication between a WF component and a supporting Processing Logic is cooperative.

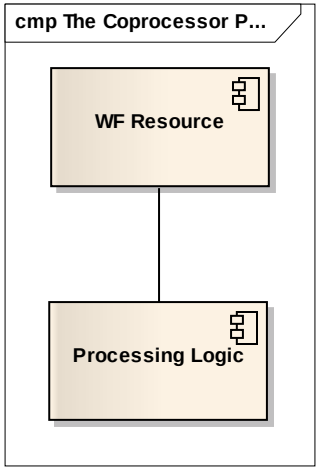
The two components are deployed independently and the inter-communication shall be performed by the means of a standard interface (e.g. using a POSIX interface)

Pros

- Increased modularity: the supporting component should be reused in other applications

Cons

- Increased complexity of WF dependency on exteranal/supporting modules
- Increased modularity may have impacts on the performances



Pattern: The Embedded-blob

SELEX ELSAG
A Finmeccanica Company

Description

There is no separation between the WF component and supporting Processing Logic

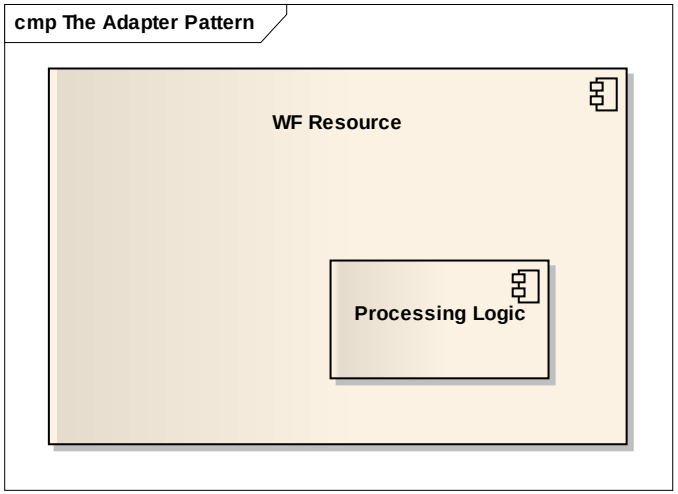
The supporting Processing Logic is deeply incorporate in the WF component. The communication is internal and the the two modules are strictly related.

Pros

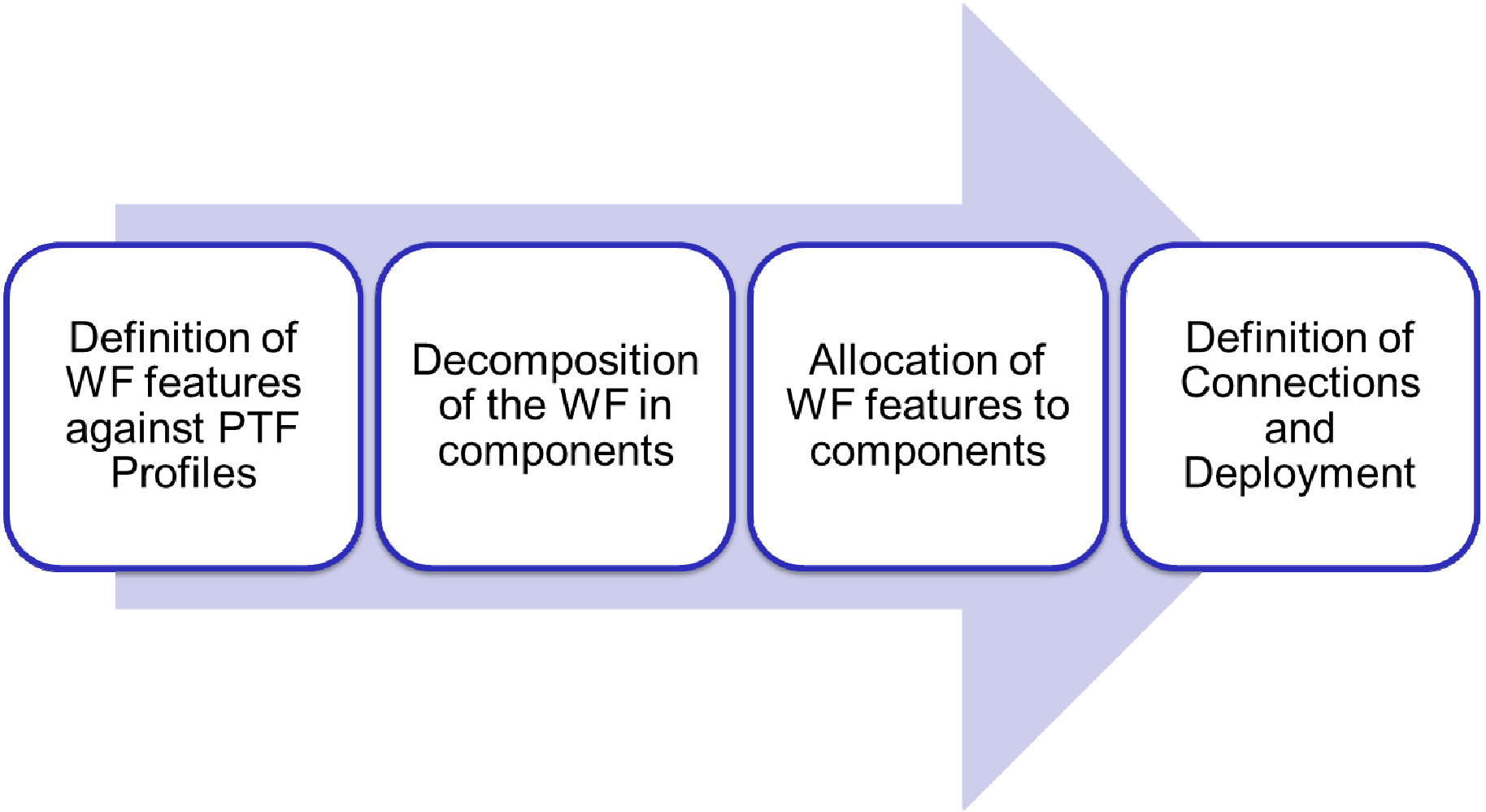
- Reduced modularity may be used in order to improve communication performances

Cons

- Reduced modulary denied the reusage of the supporting modules



The WF Design: putting all together 1/2



The WF Design: putting all together 2/2

SELEX ELSAG
A Finmeccanica Company

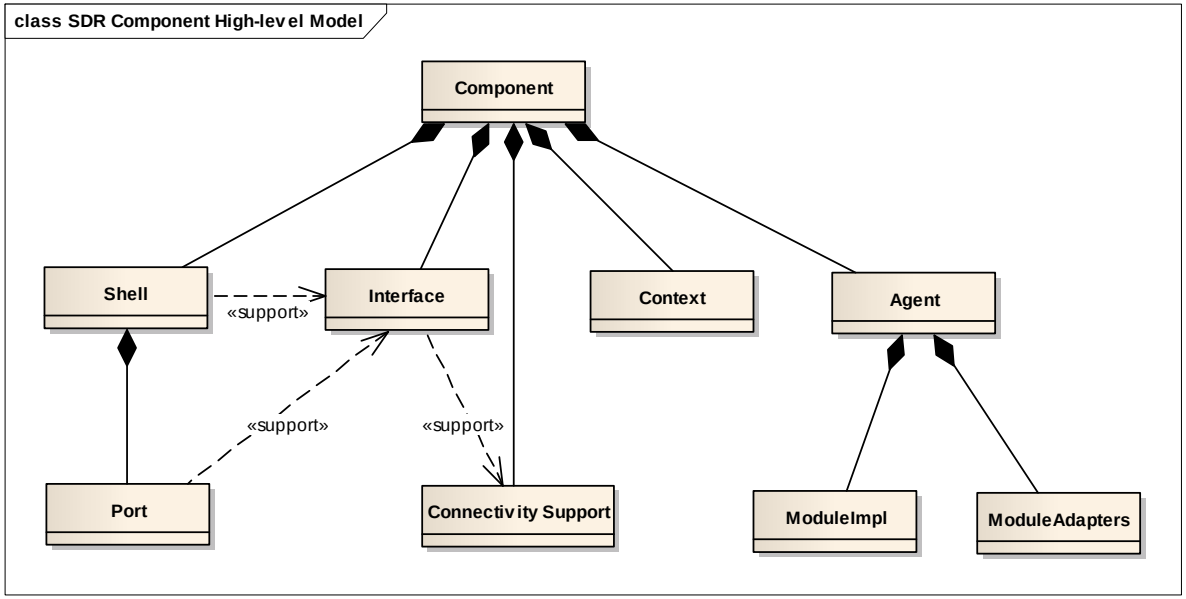
- **Definition of WF Features against PTF Profiles**
  The operative requirements of a Radio WF are allocated to specific WF Features and PTF Capabilities identified by the profile of the Radio equipment

- **Decomposition of the WF Application in components**
  The WF Application is decomposed in several software components (for example, following a application design pattern)

- **Allocation of WF Features to WF components**
  The identified WF Features are partitioned in WF software components. Each component shall implements part of the allocated functionality (usually with the support of PTF capabilities)

- **Definition of components connections and WF deployment**
  The design of the WF Application is finalized by the definition of the WF component deployment and connections (with other WF component or with PTF capabilities)

Highlights: WF Application Components



The **WF Application** may be considered as a standard SW application with a component-related design.

The same considerations of the distributed component design are to be applied.

As general approach, the component model is composed of several modules. For example:

- The **Shell Modules** provide the needed connectivity by the implementation of ports and supported interfaces (connectivity)
- The **Agent Modules** provide the control logic and use:
    - **Impl Modules** in order to provide features composition (modularity and extensibility)
    - **Adapter Modules** in order to provide the abstraction from a specific feature implementation (scalability)

Conclusions

SELEX ELSAG
A Finmeccanica Company

- The **WF Application Portability** is a Design achievement

- The main focus is related to the support provide by the different **Radio Platforms**:
  - **Application Framework**
  - **Supporting Capabilities**
  - **API Definitions**

- **WF Design** challenges
  - **WF Architecture** in terms of application layout of the internal components and with respect of the Radio Platform capabilities
  - **WF Components interactions** within internal modules or with external modules
  - **WF Features Adaptation** needed in order to achieve the porting of a Radio Waveform to a specific Radio Platform

References

SELEX ELSAG
A Finmeccanica Company

- **People**

  Daniele Olmisani, SELEX Elsag Spa
  daniele.olmisani@selexelsag.com

- **Keywords**

  Waveform, SDR, Software Defined Radio, JTRS, SCA, Software Communications Architecture, Design Patterns

- **Web & Documents**
  - JTRS SCA Specification
    http://www.public.navy.mil/jpeojtrs/sca/Pages/default.aspx
  - Design of Multi-Platform & SCA-compliant Software Components
    http://groups.winnforum.org/p/cm/ld/fid=162
  - Quick reference on Design Patterns
    http://en.wikipedia.org/wiki/Design_pattern_(computer_science)
  - **The Anatomy of Software Frameworks**
    http://www.bptrends.com

- **Image credits**

  http://shinmera.deviantart.com/art/Sine-wave-bg-173308508