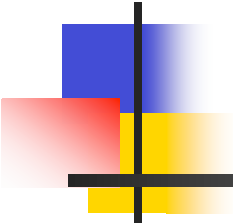


# Standard Software for Enterprise Resource Planning



---

Lecturer: Prof. Dr. Ralf Möller

Lab classes: Rainer Marrone, Michael Wessel

Lecture: Thursdays (90 minutes)

Lab classes: Fridays (60 minutes)

Prerequisite:

Lecture on ECommerce



# Enterprise Services Architecture and Composite Application Framework

---

**This lecture is based on:**

## **Master Thesis**

Mariusz Chechelski

Master degree course IMT

Supervised

Prof. Dr. J.W. Schmidt

Prof. Dr. V. Turau

Supervised

Klaus-Georg Lemke

Rudi Grom



i n v e n t



# Agenda

---

- Objectives
- Enterprise Services Architecture (ESA)
- SAP NetWeaver Technologies
  - Composite Application Framework (CAF)
- Custom Composite Application (HP)
- Conclusions



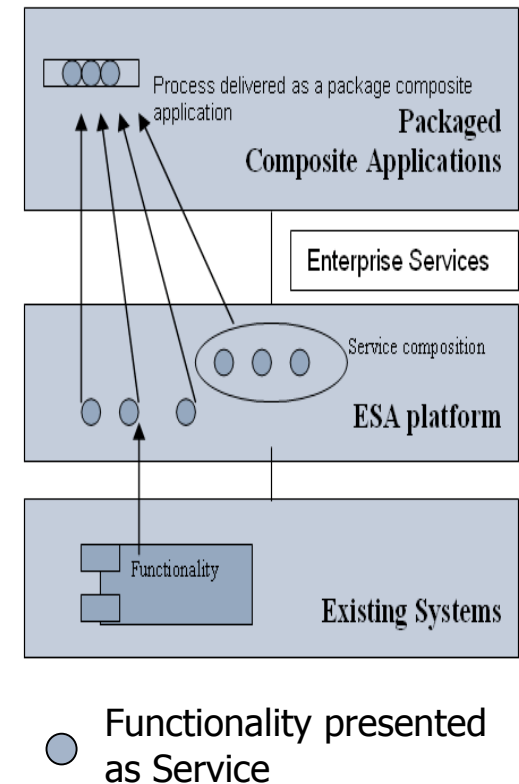
# Objectives

---

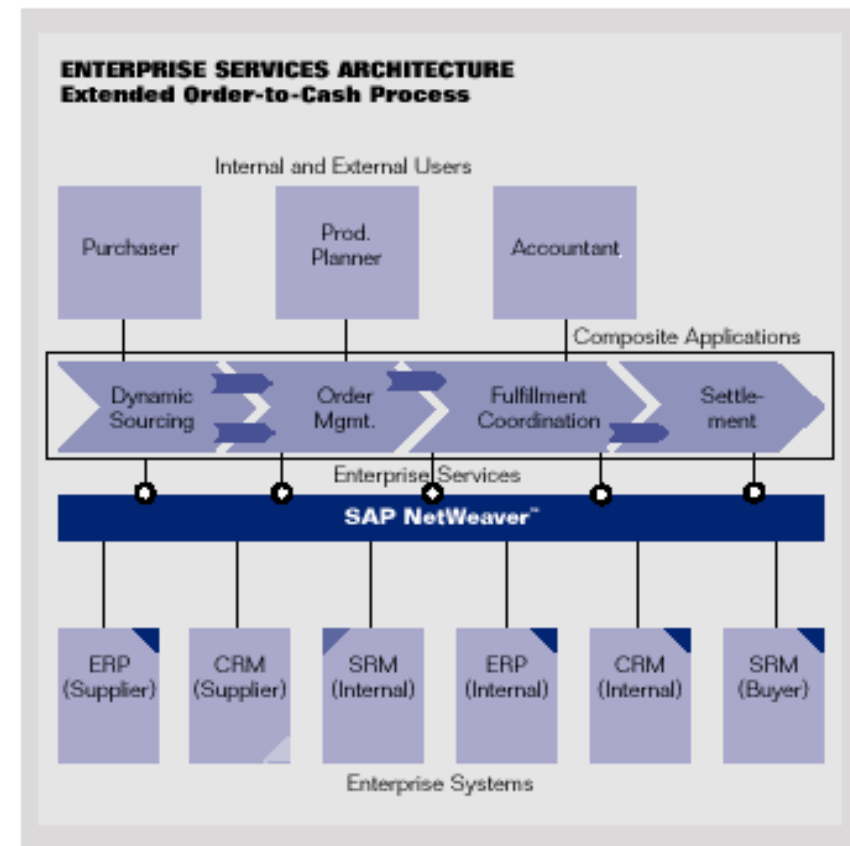
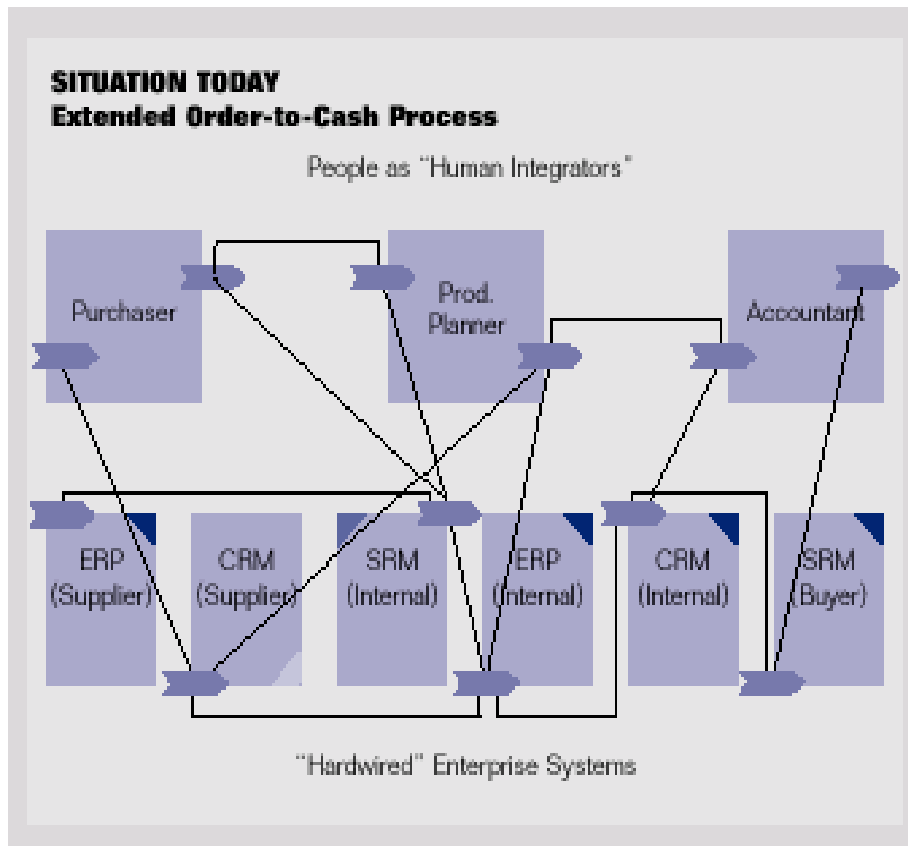
- Development and Evaluation of tools for building composite applications
  - To discuss ESA concepts with regard to the Service-Oriented Architecture (SOA) approach
  - To identify areas giving quick business benefits when using the new approach of interfacing with composite applications.
  - To identify major changes concerning the organizational and implementation methodologies when using the ESA approach

# Enterprise Services Architecture

- Enterprise Services Architecture is a methodology for building reliable services-oriented applications
- Architecture Goals
  - **Enable services centric software design** (Componentization, Services as wrappers for existing functionality)
  - **Establish ONE service infrastructure with uniform**
    - Service Definition
    - Service Implementation
    - Service Usage for all types of scenarios (UI,A2A,B2B, ...)
  - **Enable model-driven development**



# Example for moving to ESA



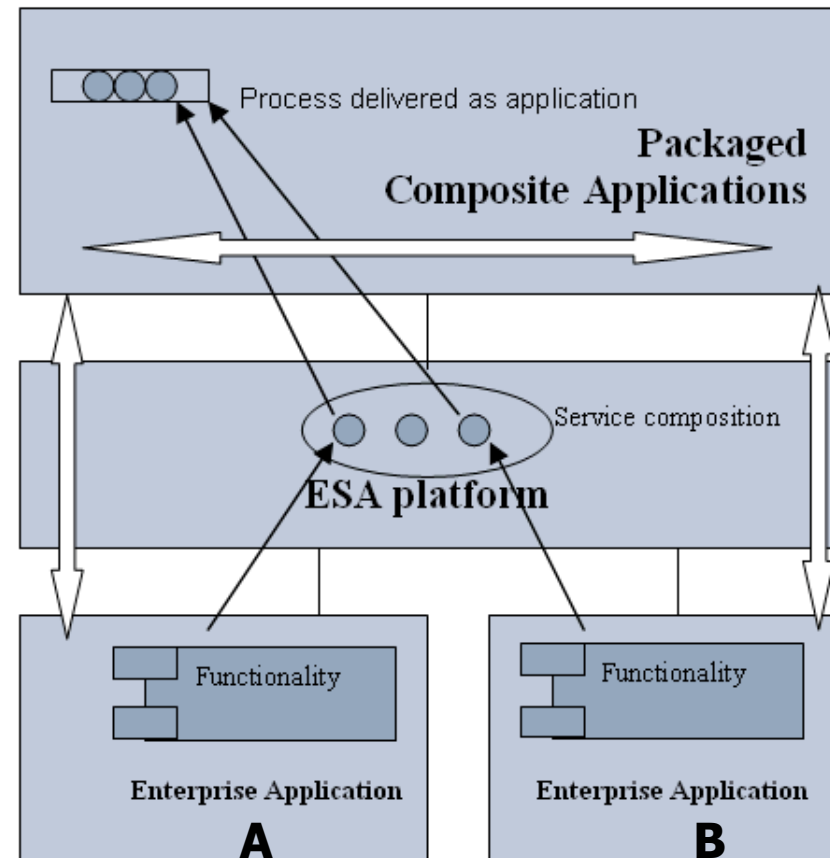
SAP Sources

# ESA – Composite Applications (1)

All applications that consume services and data, and are orchestrated to reflect new processes with the help of service-based architecture are considered **composite applications**.

Characteristics of composite applications:

- composite
- collaborative
- content-driven





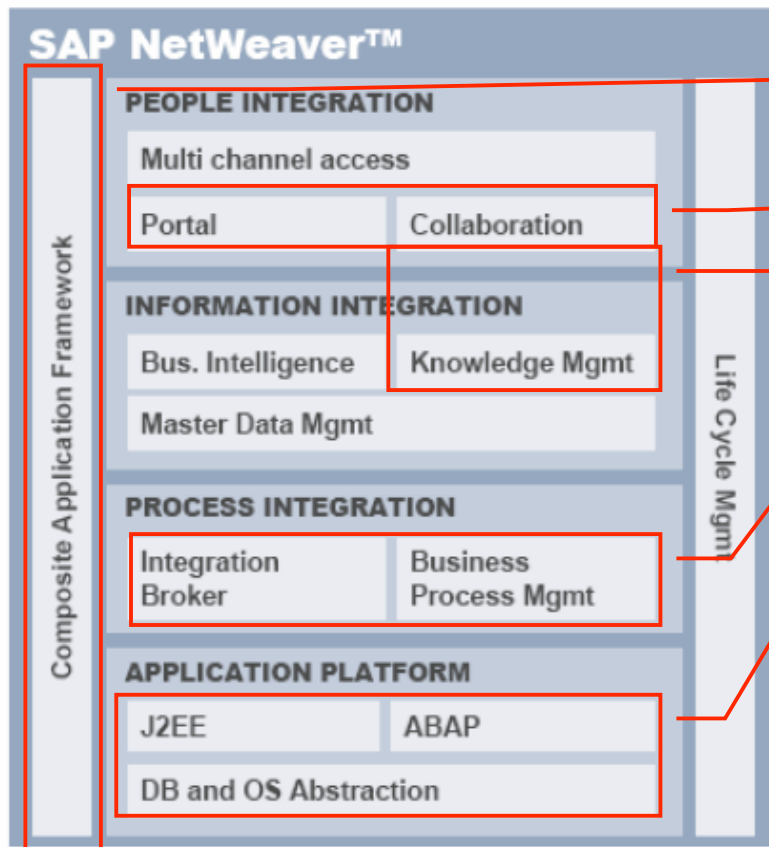
# ESA – Composite Applications (2)

---

Challenges while building composite applications

- **Integrate heterogeneous system landscape**
  - Connect to existing repositories
  - Separate Business Objects from persistency
  - Remote data access to backend systems
- **Leverage existing Services**
  - Support semi-structured, ad-hoc business processes
  - Provide collaboration services for Business Objects
  - Workflow tools and workflow patterns
- **Create a homogeneous user experience**
  - Encapsulate user interface from application data
  - Implement consistent pattern-based user interface
- ...

# SAP NetWeaver Platform



Composite Application Framework

SAP Enterprise Portal (EP)

SAP Exchange Infrastructure (XI)

SAP Web Application Server (WAS)

**SAP Sources**



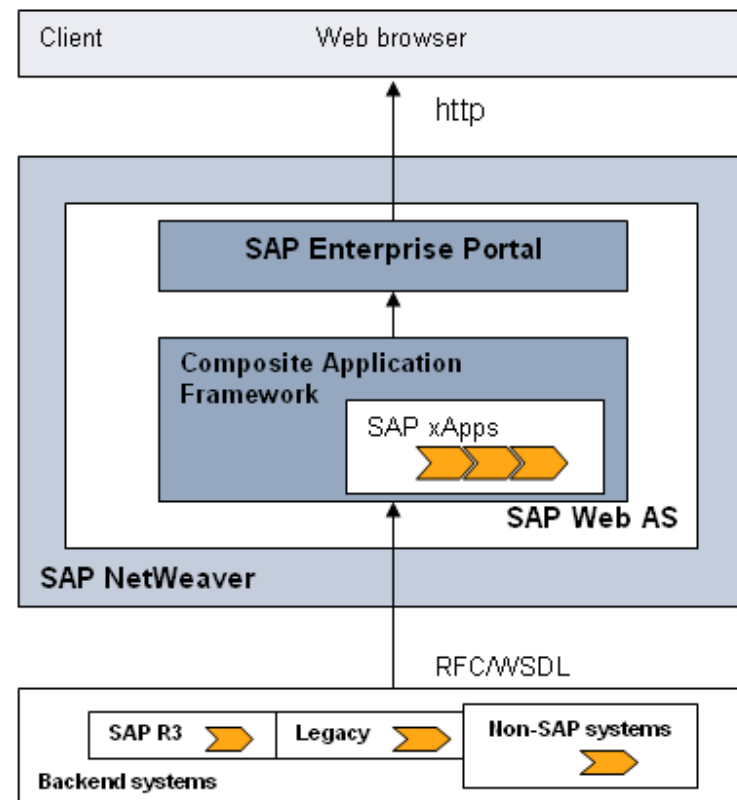
# SAP's Composite Applications

SAP's composite applications portfolio:

- **SAP xApps**
  - SAP xCQM – Cost&Quotation Management
  - SAP xRPM – Resource&Portfolio Management
  - SAP xPD – Product Definition
  - ...
- **Custom Composite Applications**

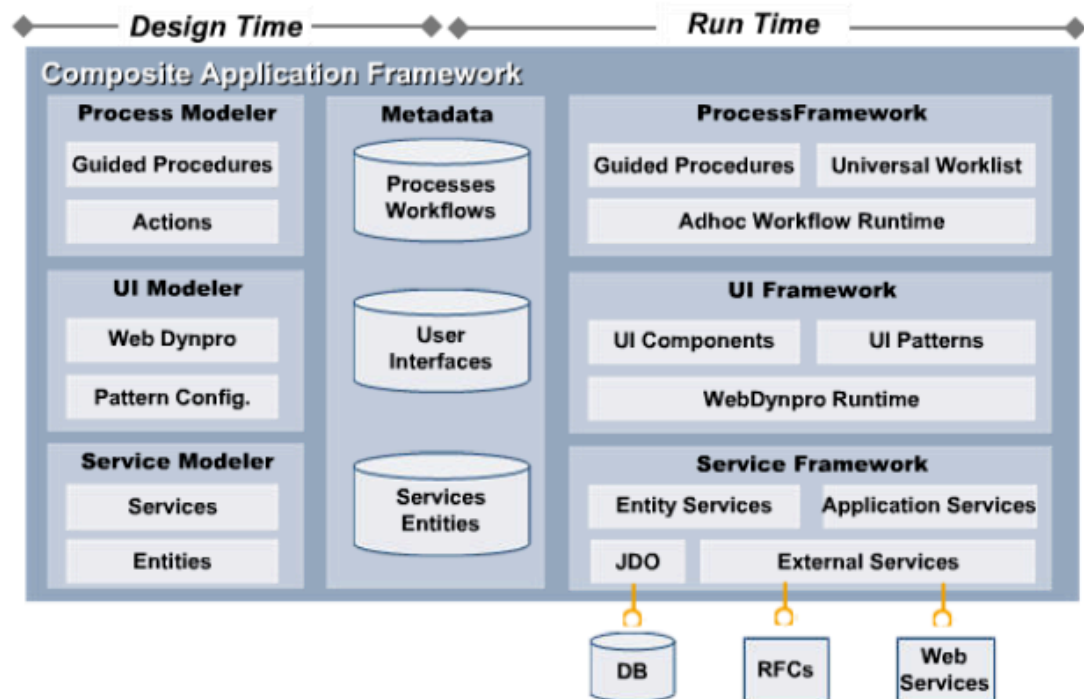


## Architecture

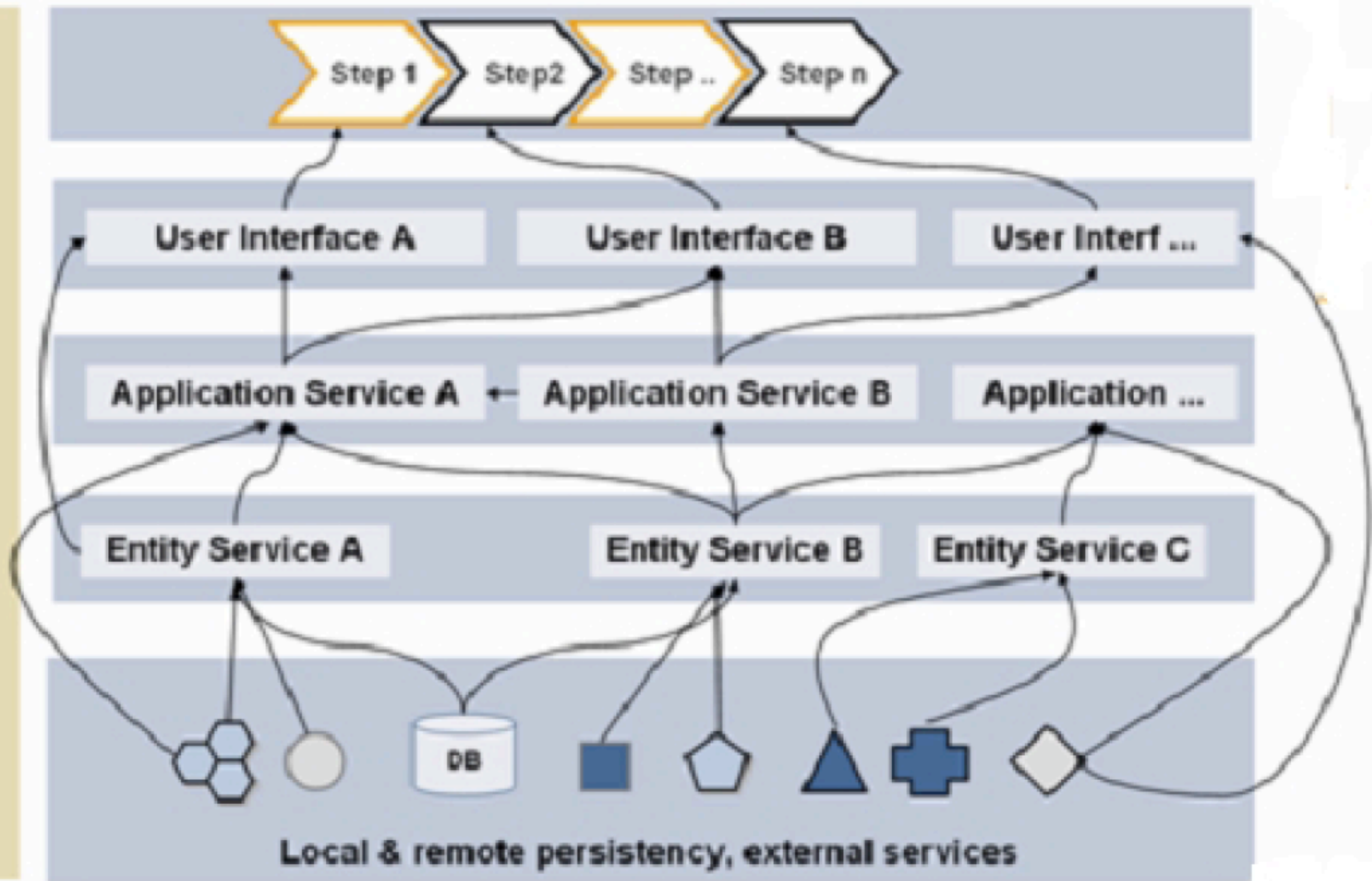


# Composite Application Framework

- A methodology for building composite applications
- A model- and metadata-driven programming model (focus on business logic)
- Modeling and generation tools
- UI & workflow patterns
- Repository
- Runtime framework

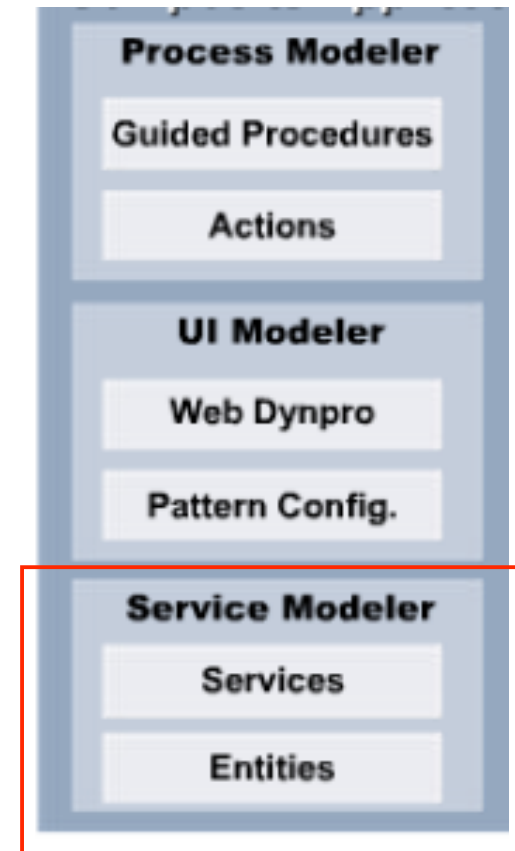


**COMPOSITE APPLICATION FRAMEWORK**



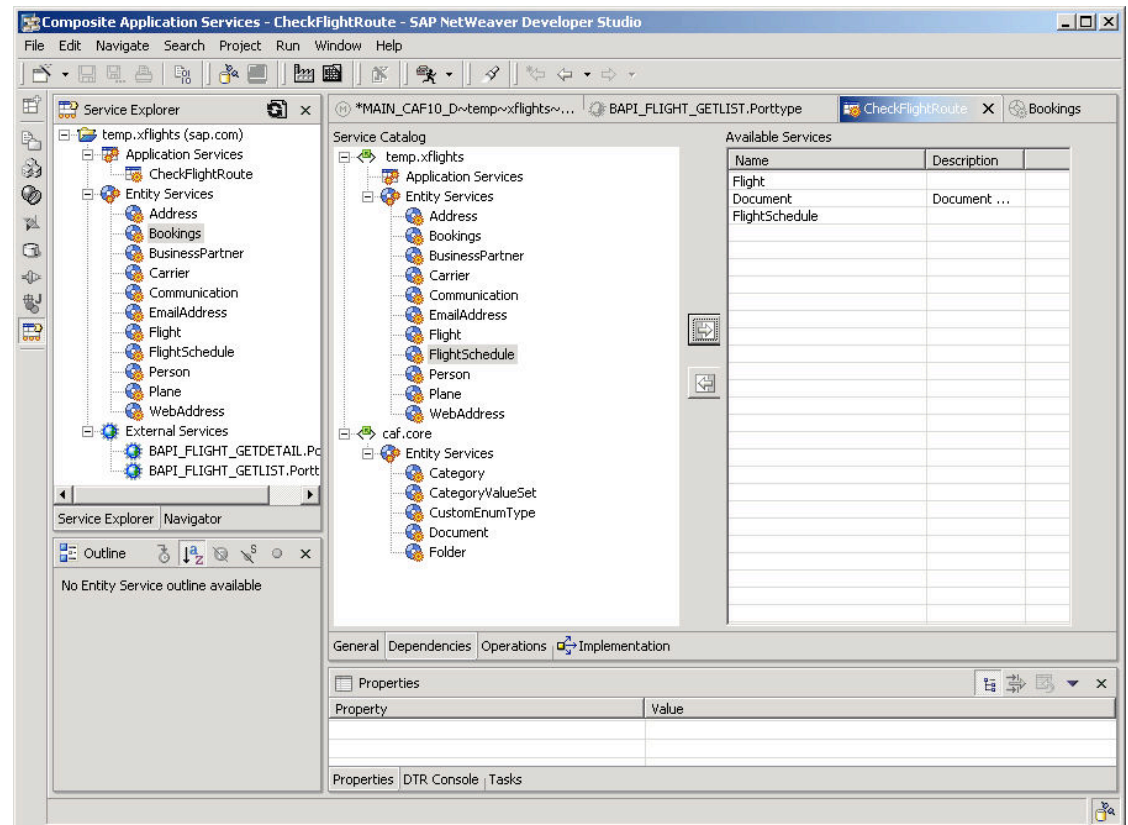
# Services Layer

- External Services
  - Import of existing services (web services, RFC)
- Entity Services
  - business objects (Attributes, Operations, Permissions,...)
  - Local or remote persistency (External Services)
  - Interfaces for Application Services
- Application Services
  - business logic (custom operations, custom code)
  - Composite services (external and entity services)
  - Exposed as Web Services, published in UDDI registry
- Services are implemented as EJB session beans
  - Transaction, authorization, logging & tracing, and collaboration infrastructure provided



# Services Modeler

- SAP NetWeaver Developer Studio
  - Based on Eclipse IDE
  - A model- and metadata-driven programming model

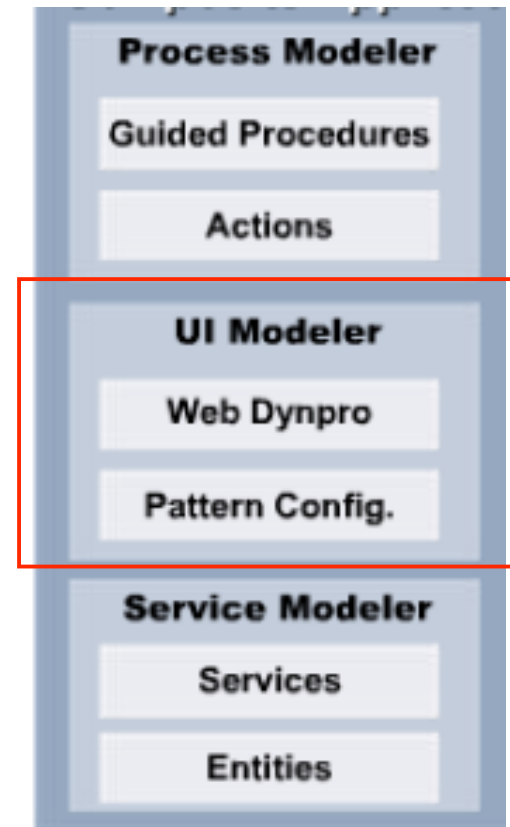




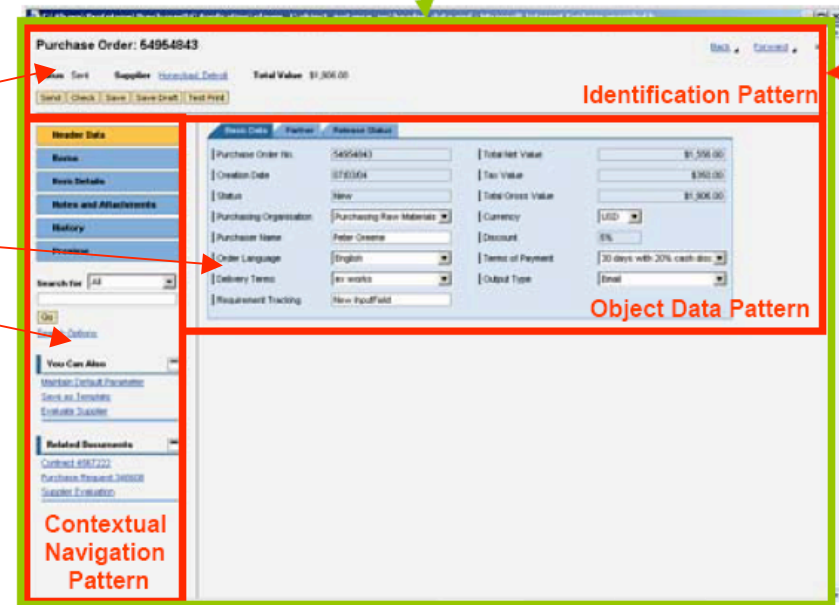
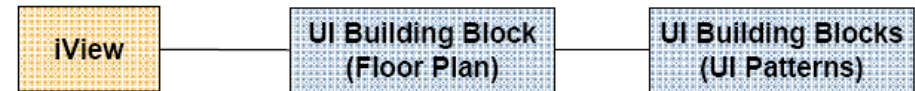
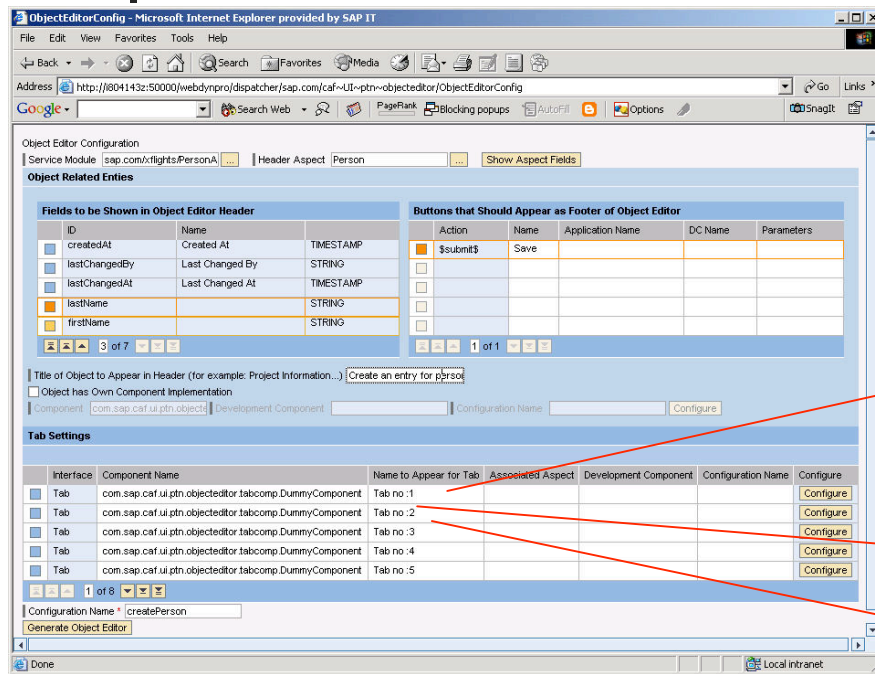
# UI Layer

---

- Web Dynpro Technology
  - Visualization and application logic is separated (MVC pattern)
  - Templates for recurring application patterns
- UI Patterns
  - Page Patterns and Component Patterns
  - Reuse of composite services (application services).
  - Pre-configured data binding
  - UI patterns configuration
- Web Dynpro Foundation Applications can be integrated as well



# UI Building Blocks



# Process Layer – (not available)

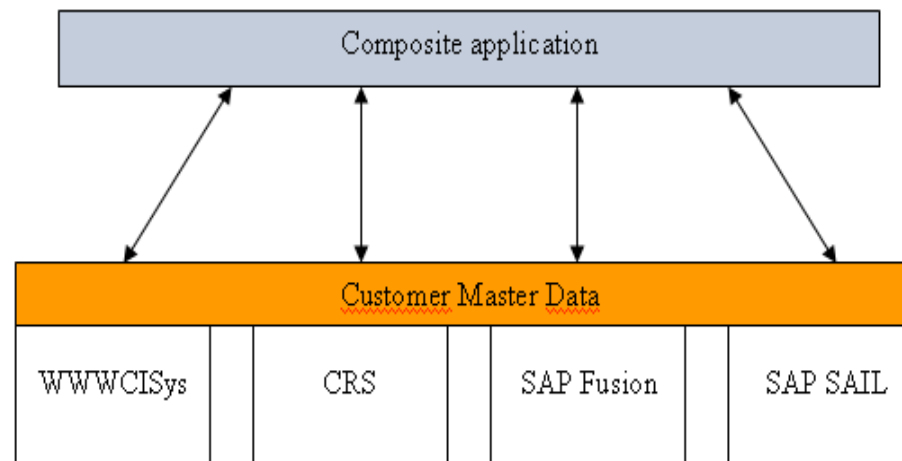
- Guided Procedures is a workflow environment to set up and execute collaborative business processes. It is a Web application that presents business process to users
  - Process workflow (phases and steps)
  - Collaborative workflow patterns (e.g. delegation, approval)
  - Service integration (actions)
- Actions seamlessly integrate in the Guided Procedures user interface. They enable the framework to display the required applications and manage the business process context at runtime. Actions can launch
  - iView, Interactive form, Web service





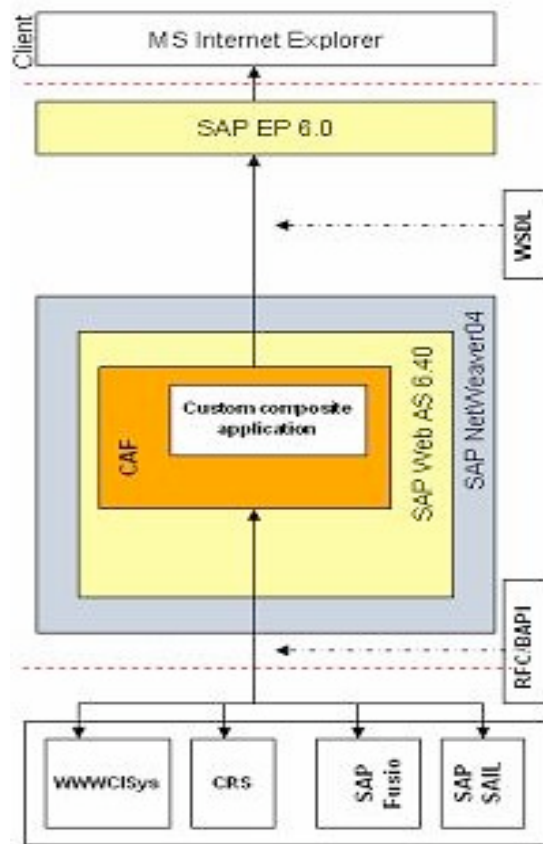
# Example App based on CAF

- Objectives: Implement a UI application with ESA principles and SAP NetWeaver Technologies.
  - Seamless and comprehensive view on customer data shared among different backend systems
  - Simple functionality: create/maintain and search of customer master data

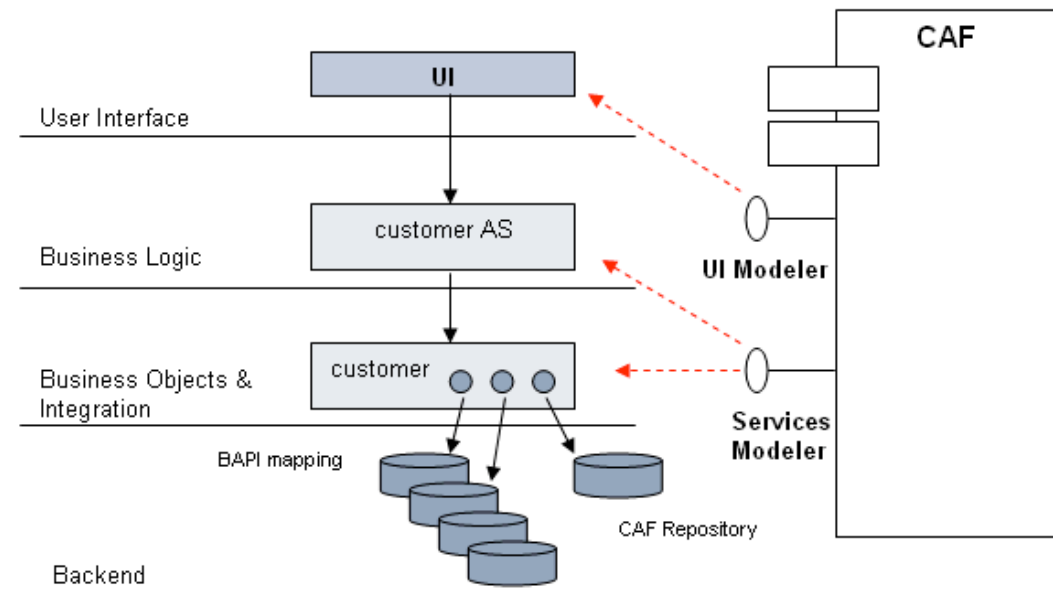


# Custom Composite Application

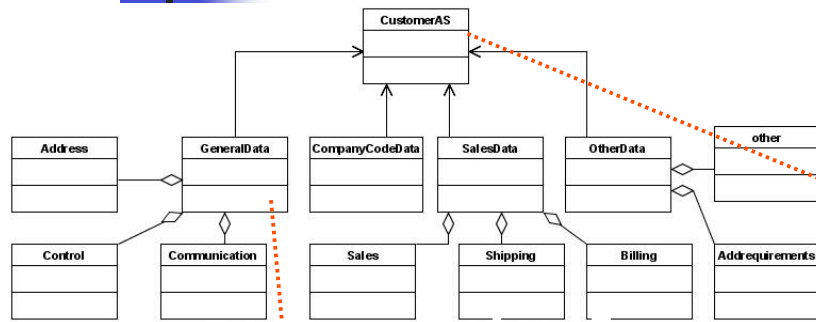
## Target Architecture



## Abstraction Model of Implementation



# Model - Example

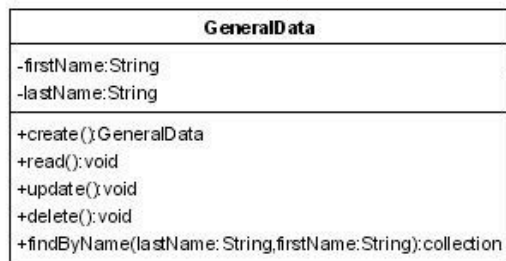


Business Objects Model

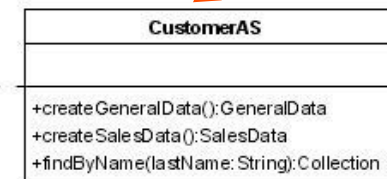
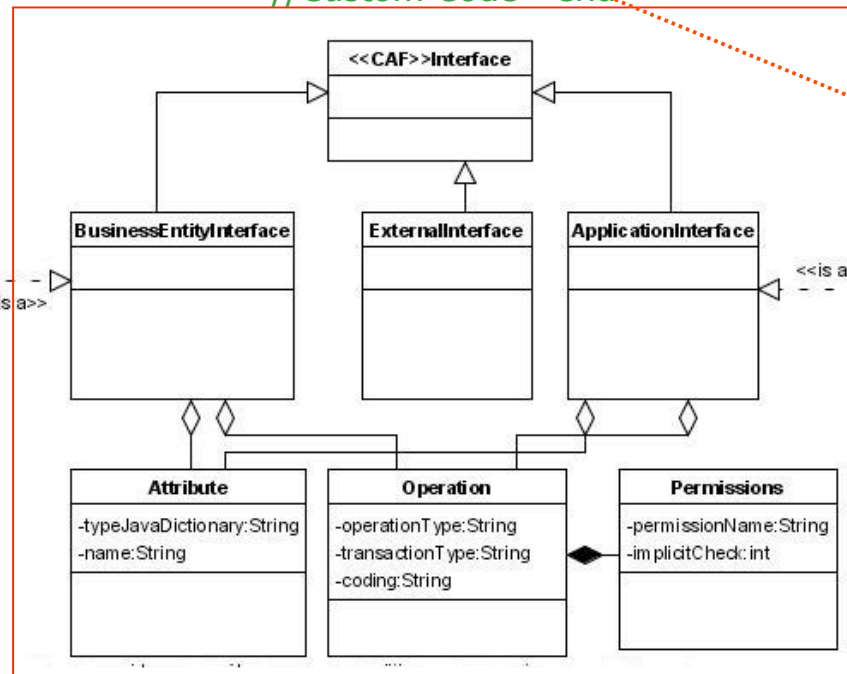
Coding examples for the CustomerAS service

```

//Custom Code start – findByName() operation
GeneralDataServiceLocal local = this.getGeneralDataService();
retValue =
(java.util.List)local.findByName(new QueryFilter(lastName));
//Custom Code - end
  
```



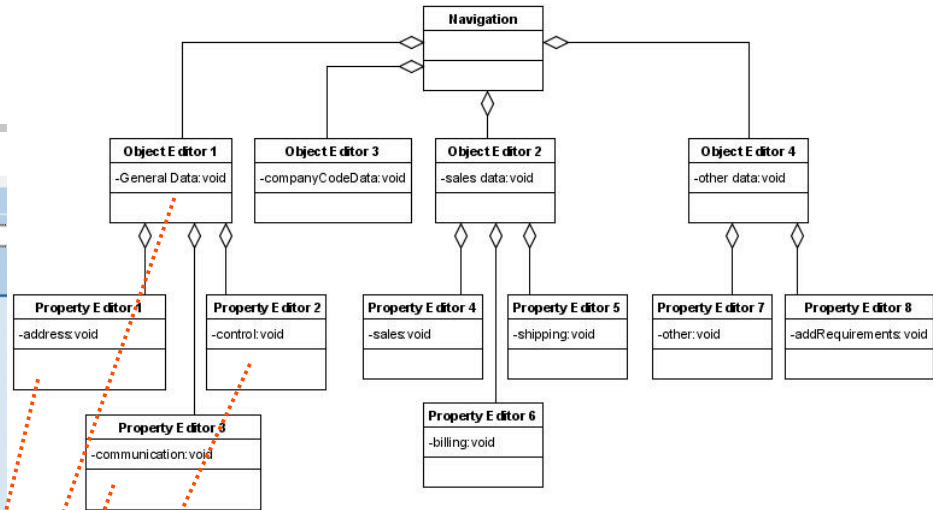
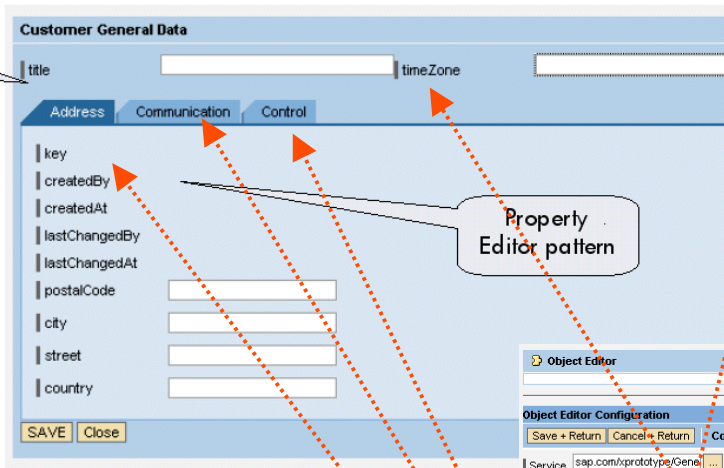
Simplified CAF Metamodel



<<is a>>

# UI modeling

Object Editor pattern



UI pattern - Runtime

UI Pattern configuration

Object Editor Configuration

Save + Return Cancel + Return Configuration name: xGeneralData Reload Preview

Service: sap.com/xprototype/GeneralData Object: GeneralData Retrieve metadata

Object related entities

| Attributes to be shown in Object Editor Header |                 |            |            | Buttons that should appear as footer of Object Editor |            |                  |         |            |
|--|-----------------|------------|------------|---|------------|------------------|---------|------------|
| ID   | Name            | Field Type | Field Type | Operation   | Name       | Application Name | DC name | Parameters |
| createdAt                                      | Created At      | TIMESTAMP  |            | <input type="checkbox"/>                              | \$submit\$ | SAVE             |         |            |
| lastChangedBy                                  | Last Changed By | STRING     |            | <input type="checkbox"/>                              |            |                  |         |            |
| lastChangedAt                                  | Last Changed At | TIMESTAMP  |            | <input type="checkbox"/>                              |            |                  |         |            |
| title  | title           | STRING     |            | <input checked="" type="checkbox"/>                   |            |                  |         |            |
| timeZone                                       | timeZone        | STRING     |            | <input type="checkbox"/>                              |            |                  |         |            |

Title of Object to appear in header (e.g Project Information...): Customer General Data

Object has own component implementation:

Tab Settings

| Interfaces                          | Name to appear for tab | Reference Name   | Component Name  | Development Component       | Configuration name        | Configure |
|-------------------------------------|------------------------|------------------|---|-----------------------------|---------------------------|-----------|
| <input checked="" type="checkbox"/> | Address                | addressRef       | com.sap.caf.ui.ptn.propedit.components.PropertyEditorComp | sap.com/caf-ui-ptn-propedit | xGeneralDataAddress       | Configure |
| <input checked="" type="checkbox"/> | Communication          | communicationRef | com.sap.caf.ui.ptn.propedit.components.PropertyEditorComp | sap.com/caf-ui-ptn-propedit | xGeneralDataCommunication | Configure |
| <input checked="" type="checkbox"/> | Control                | controlRef       | com.sap.caf.ui.ptn.propedit.components.PropertyEditorComp | sap.com/caf-ui-ptn-propedit | xGeneralDataControl       | Configure |
| <input type="checkbox"/>            | Tab no :4              |                  | com.sap.caf.ui.ptn.objecteditor.tabcomp.DummyComponent    |                             |                           | Configure |
| <input type="checkbox"/>            | Tab no :5              |                  | com.sap.caf.ui.ptn.objecteditor.tabcomp.DummyComponent    |                             |                           | Configure |



# Conclusions

---

## **Enterprise Services Architecture**

- Methodology to define core processes within a company; delivered as composite applications
- Based on SOA (componentization, services)
- Enterprise Services for business functionalities (based on Web Services)

## **Composite Application Framework**

- Applications integration through services (external and entity services). Services implemented as EJB session beans
- Services modeling depends on UI pattern knowledge
- UI implemented through patterns
- Guided Procedures not released (May 2005) therefore not used for workflow definition
- No standards for Enterprise Services