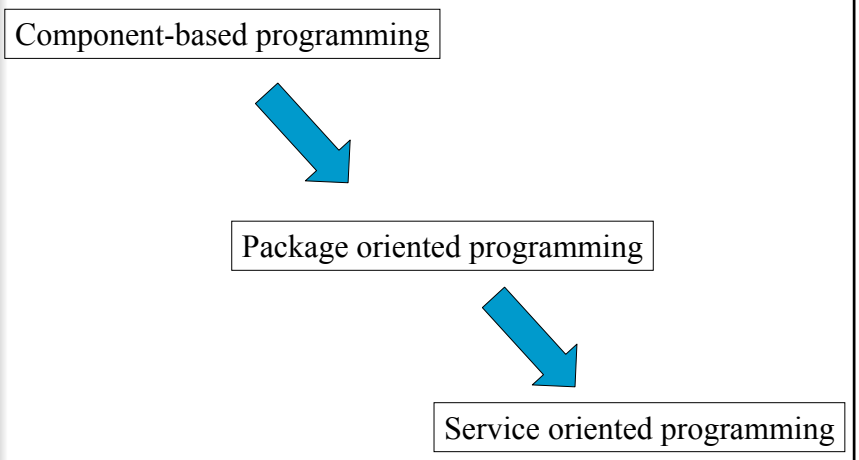
Service Based Product Lines

Alberto Sillitti (alberto@dist.unige.it)
Tullio Vernazza (tullio@dist.unige.it)
Giancarlo Succi (Giancarlo.Succi@unibz.it)

Table of Contents

Integration Oriented Programming



Component-based programming

- Components are usually simple and encapsulate very specific features
- There are many incompatible architectures (COM, CORBA, EJB)
- The chosen architecture define a subset of components that the developer can use (there are adapters that can be used to integrate different technologies but they don't provide specific architecture's features)

Package oriented programming

- It exploits mass-market applications as large components (like MS Office programs)
- It provides a user familiar new product with low costs
- It is platform dependant: it is not possible to integrate components running on different platforms

Service Oriented Programming (1)

- Web services are often characterized by:
  - Independent development and deployment
  - Functionality encapsulation and implementation details hiding
  - Interfaces
- These features make a service like a component
- Traditional product lines are developed integrating components, service based product lines are developed integrating web services

Service Oriented Programming (2)

- WIDL (Web Interface Definition Language) and WSDL (Web Service Description Language) provide a standard way to describe a web service helping the integrator to use it
- It solves problems of component-based programming and package oriented programming:
  - It is possible and easy to integrate components based on different technologies
  - The best platform for every single application can be chosen

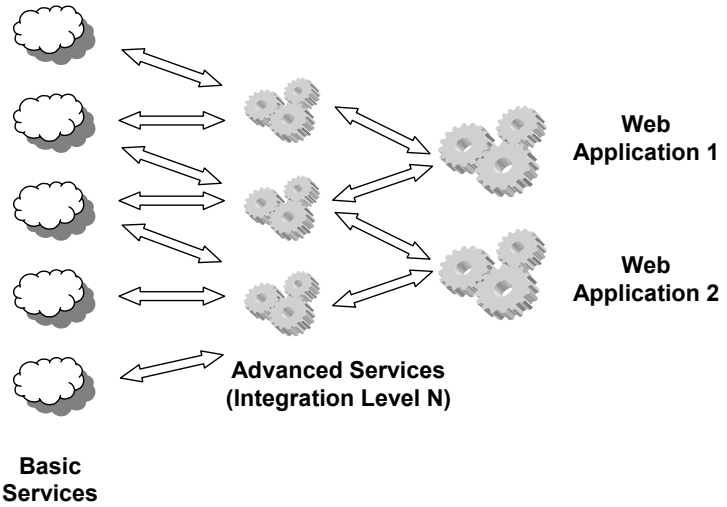
Service Oriented Programming (3)

- It provides further benefits:
  - It is possible to exploit components that run inside incompatible environments (operating systems, software libraries, etc.)
  - No time and effort is required to setup the working environment: it is already configured and working on a remote machine waiting for requests

Service Based Product Lines

- Service could be considered as components, in this way we can talk about product lines based on web services.
- The role of the application developer changes from integrating components running on a single machine to integrating services provided by web distributed computers.
- Users will be able to access application wherever they are and using different kind of clients (PCs, PDAs, Cell Phones, etc.)

Layered Services Integration

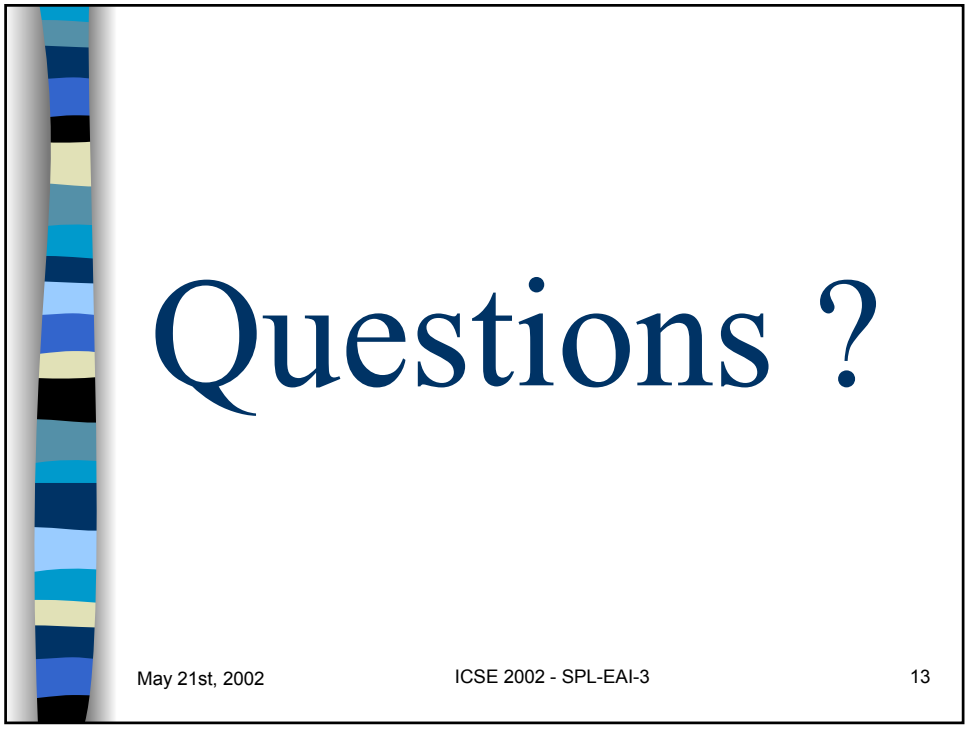


Economic Benefits

- Focus on company core business, not waist time in IT issues.
- No time and effort spent to application installation, configuration, and updating (ASPs provide system maintenance and assure a certain quality of service based on fees)
- Bugs and security issues are solved by ASPs
- Pay-per-use
- Applications are accessible through different clients and wherever users are using a standard Internet connection

To Do

- Automated integration reconfiguration
- Automated identification of set of services providing the same functional abilities
- Evaluation of non functional properties of services
- Support of asynchronous services


Questions ?