Introduction

CMPE 273 Enterprise Distributed Systems

Prerequisites

- CmpE 272
  - Special cases may waive 272 - if you have professional software engineering experience (not internships) – must have instructor consent
- Proficient in one or more OO languages
  - Java preferred
- Strong organizational and team skills

Projects

Projects are an opportunity to further discussions and exploration through practical experience.

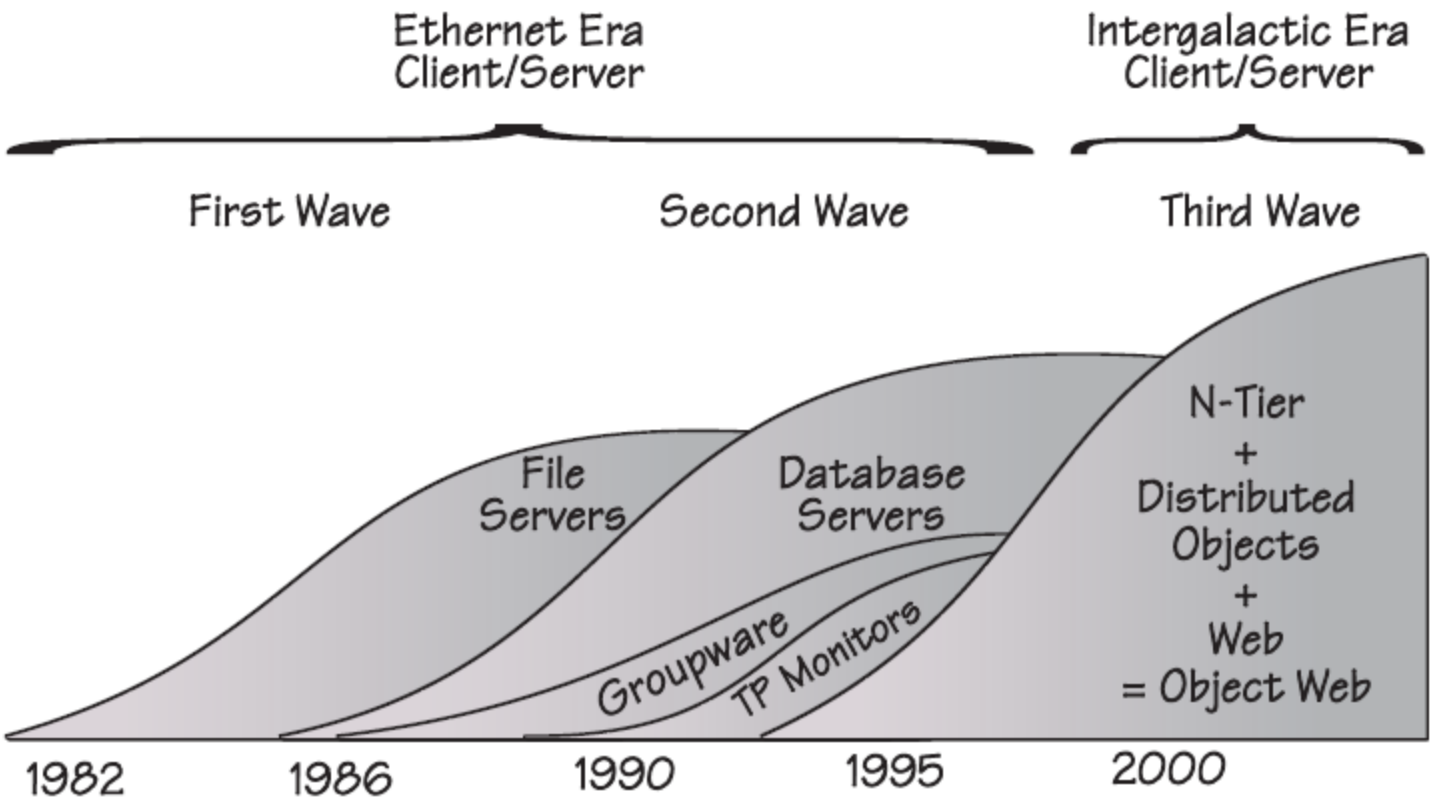
3 Programming projects and 1 class project

1. Programming projects – individual
2. Class project: Team project design and implementation

Manage your time carefully

- Projects take 30-70 hours to complete
  - Do not wait until the last week
- What you have to plan for
  - Project investigation
  - Understanding concepts and technologies
  - Implementation
  - Testing
  - Documentation

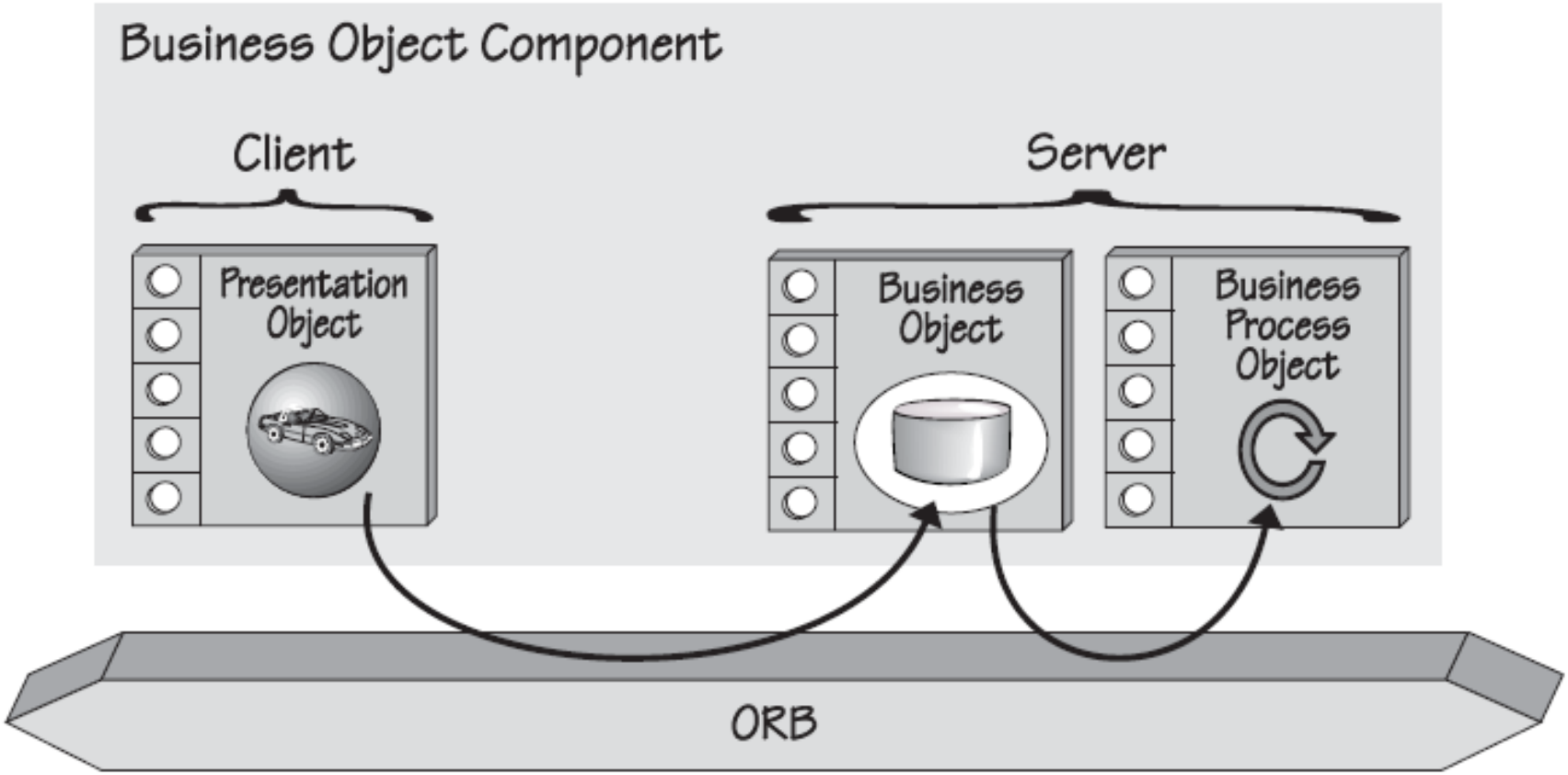
Client/Server



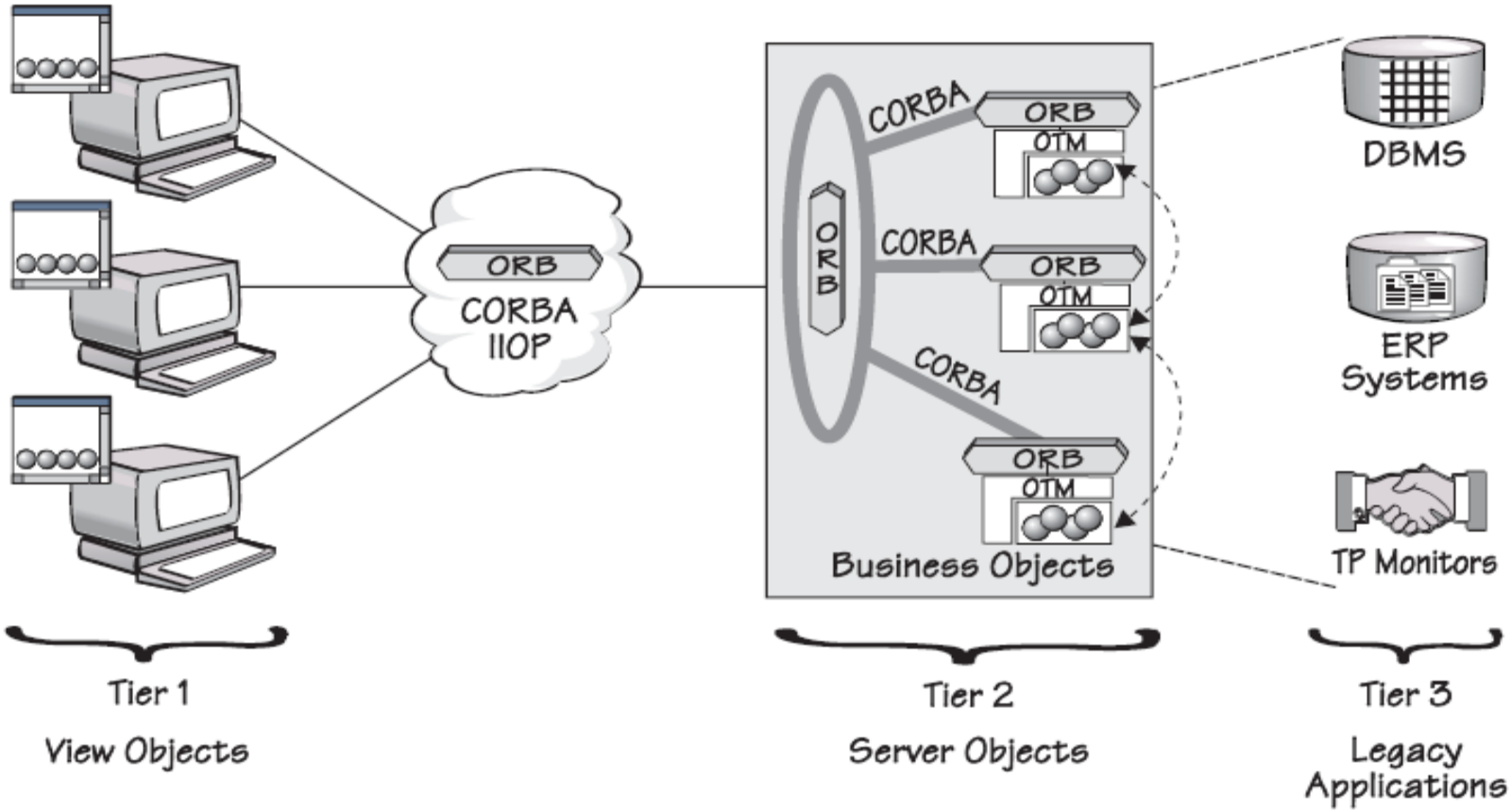
Object Orientation

- Class encapsulation
- Polymorphism
- Inheritance

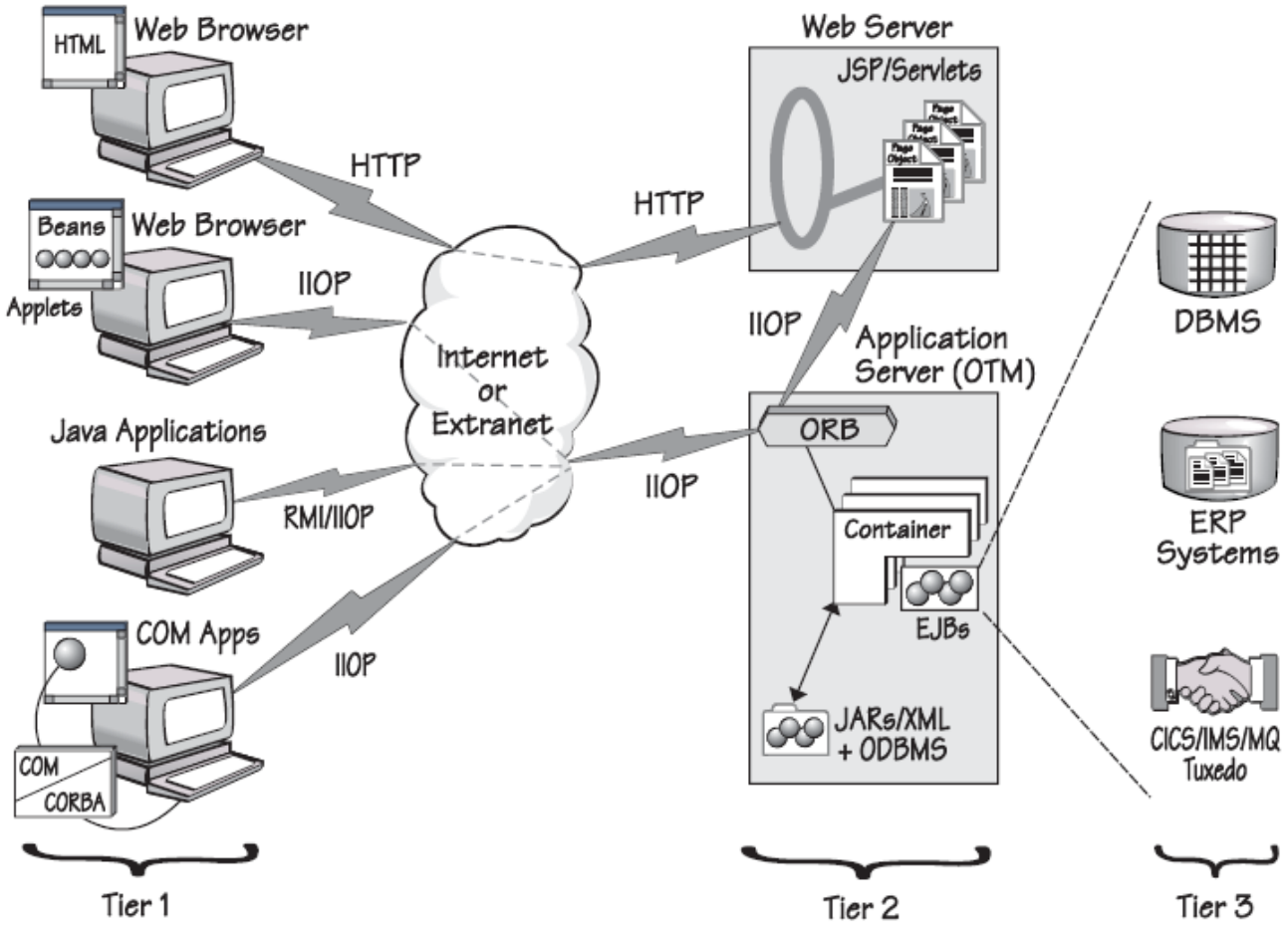
Distributed Business Object Component



3-Tier Client/Server Object Style

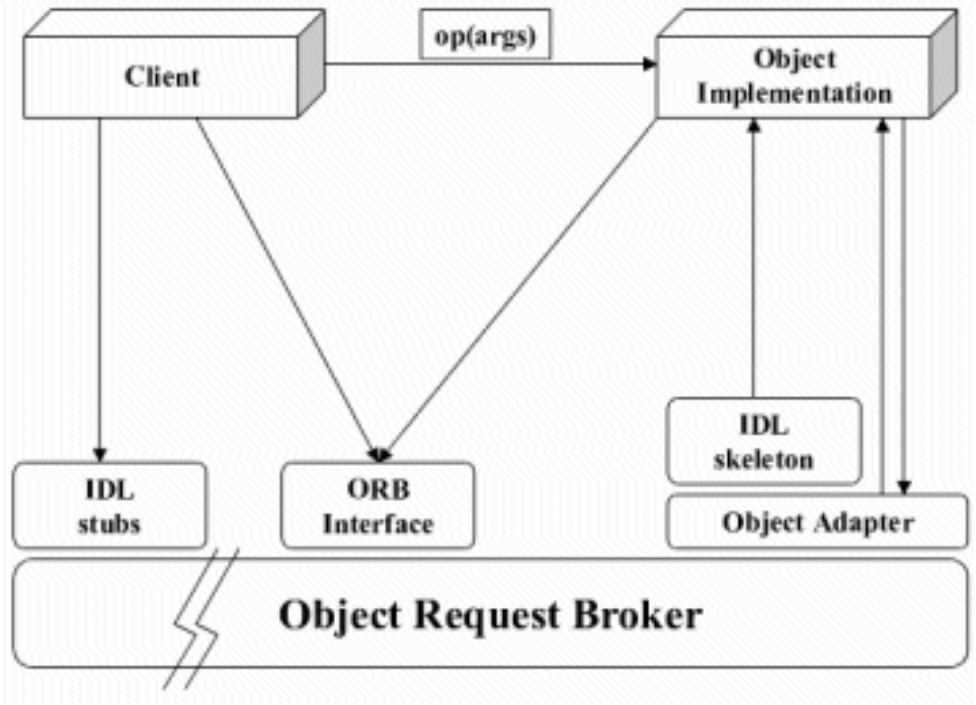


3-Tier Client/Server with CORBA/Java



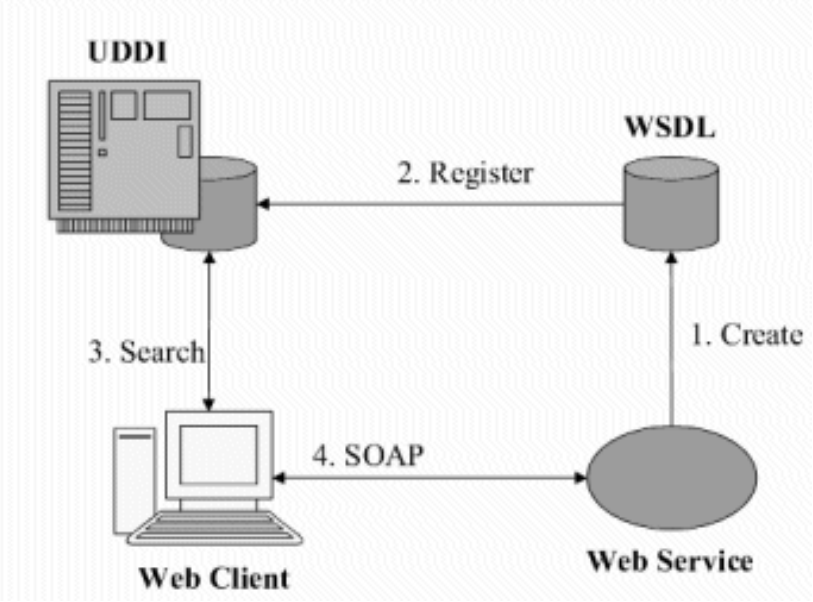
CORBA

- CORBA objects can be either collocated with the client or distributed on a remote server, without affecting their implementation or use.

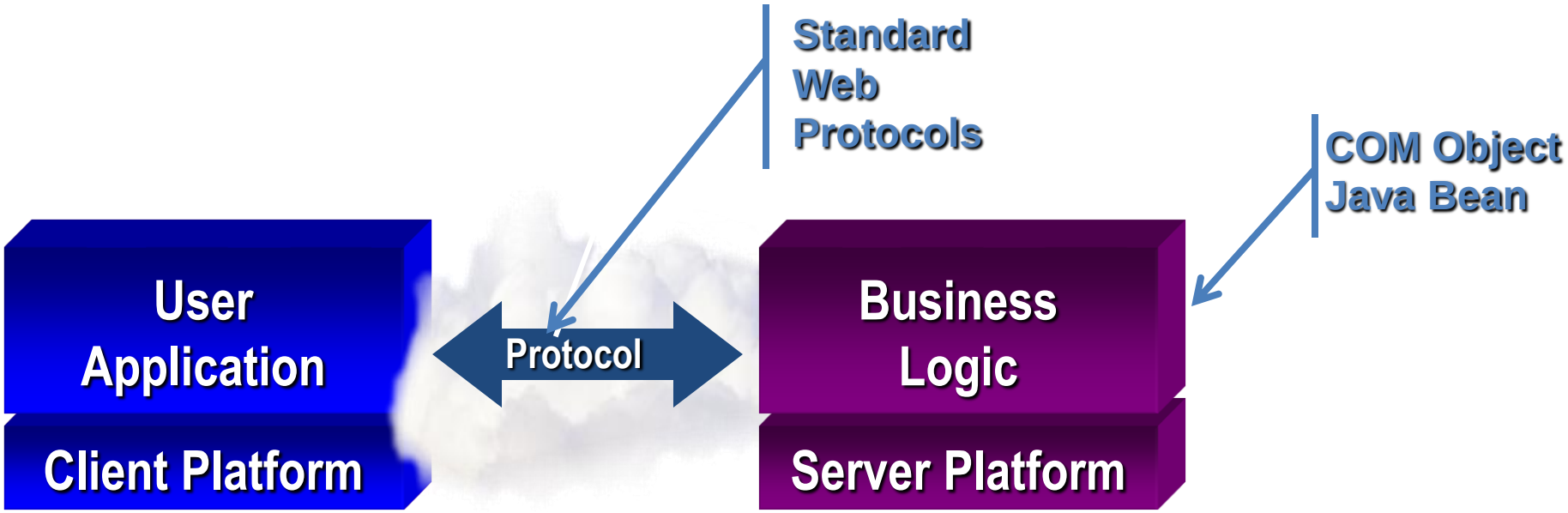


Web services

- HTTP based
- SOAP/UDDI/WSDL is a general framework (based on XML) for describing network services



Real Life Scenario



Standard web protocols makes the platform specific business logic

RESTful Services

- Representational State Transfer (REST)
  - Roy Fielding, 2000 (doctoral dissertation)
    - Examination of the Internet as a stateless service of near-limitless expansion model with a simple but effective information delivery system

- Key concepts
- Resources - source of information
- Consistent access to all resources
  - As in interface and communication – Not content or function
- Stateless protocol
- Hypermedia – links in the information to other data (connectedness)

What is the REST style?

- REST is often described as an architecture style
  - Set of formal and informal guides to creating architectures – “constraints”
    - Client-server
    - Stateless
    - Cacheable
    - Layered system
    - Uniform interface
    - Code on demand (optional)

Cloud Computing

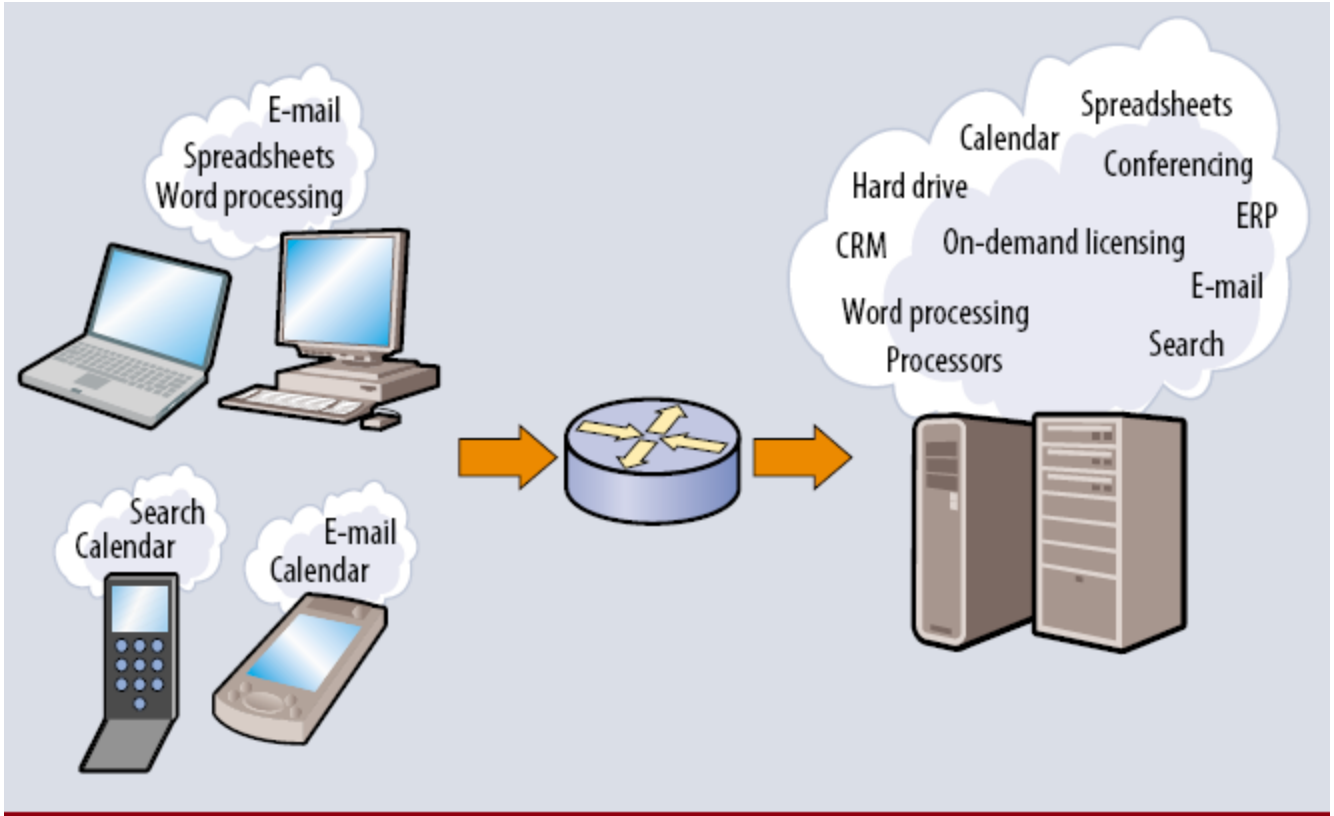
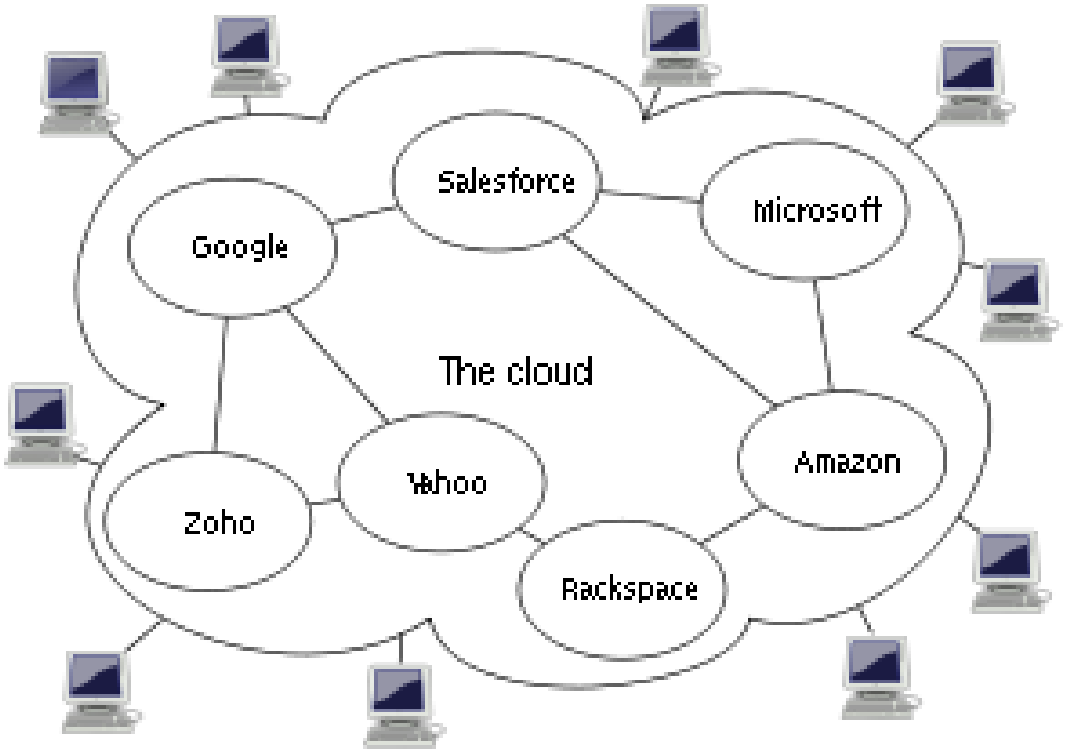


Figure 1. Applications are moving from the local machine to the Ubiquitous Computer.

The Web as the Ubiquitous Computer, IEEE Computer 2009, V Pendyala, S. Shim

Cloud


Salesforce
Microsoft
Google
The cloud
Amazon
Yahoo
Zoho
Rackspace