Contents of the Course

- Tiscover
- Internet & World Wide Web
- Introduction to WE & Characteristics of Web Applications
- WE Tasks and Phases
- Web Modelling
- Web Usability
- Web Performance/Caching
- Personalization/Adaptation
- Web 2.0

Web Engineering

Why?

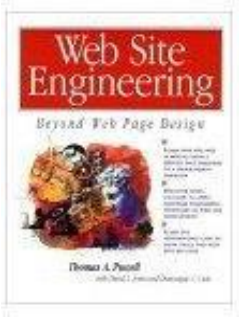
Why Web Engineering ?

Problems in developing Web applications !

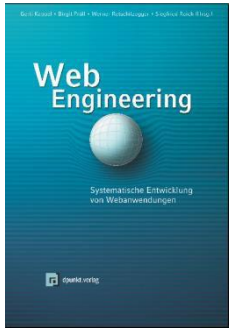
Characteristics of Web applications !

Web Engineering

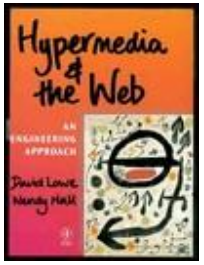
Books on Web Engineering (1/2)



Web Site Engineering
Th. A. Powell
Prentice Hall 1998



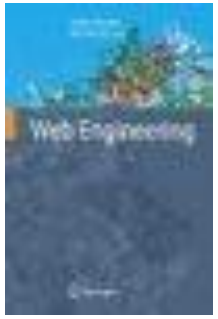
Web Engineering
Systematische Entwicklung von Webanwendungen
G. Kappel, B. Pröll, S. Reich, W. Retschitzegger (Hrsg.), dpunkt.verlag 2004



Hypermedia and the Web: An Engineering Approach
D. Lowe, W. Hall
Wiley 1999

Web Engineering
The Discipline of Systematic Development of Web Applications
G. Kappel, B. Pröll, S. Reich, W. Retschitzegger (Eds.), Wiley 2006
http://www.web-engineering.at

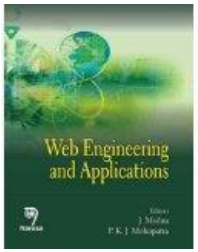
Web Engineering
R. Dumke, M. Lother, C. Wille, F. Zbrog
Pearson 2003



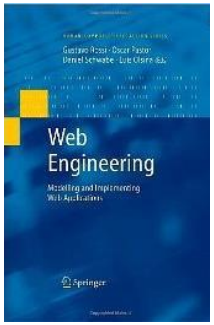
Web Engineering
E. Mendes, N. Mosley (Eds.)
Springer 2006

Web Engineering
Books on Web Engineering (2/2)

Engineering Web Applications
S. Casteleyn, F. Daniel, P Dolog, M. Matera
Springer 2009

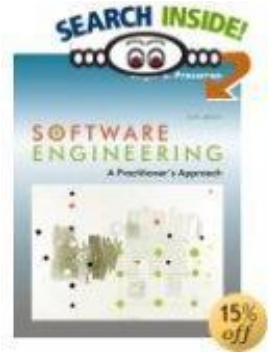


Web Engineering and Its Applications
J. Mishra
Narosa Pub House, 2008

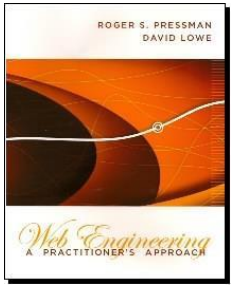


Web Engineering: Modelling and Implementing Web Applications
G. Rossi, O. Pastor, D. Schwabe, L. Olsina
Springer 2007

Web Engineering: Principles and Techniques
W. Suh
IGI Global 2005



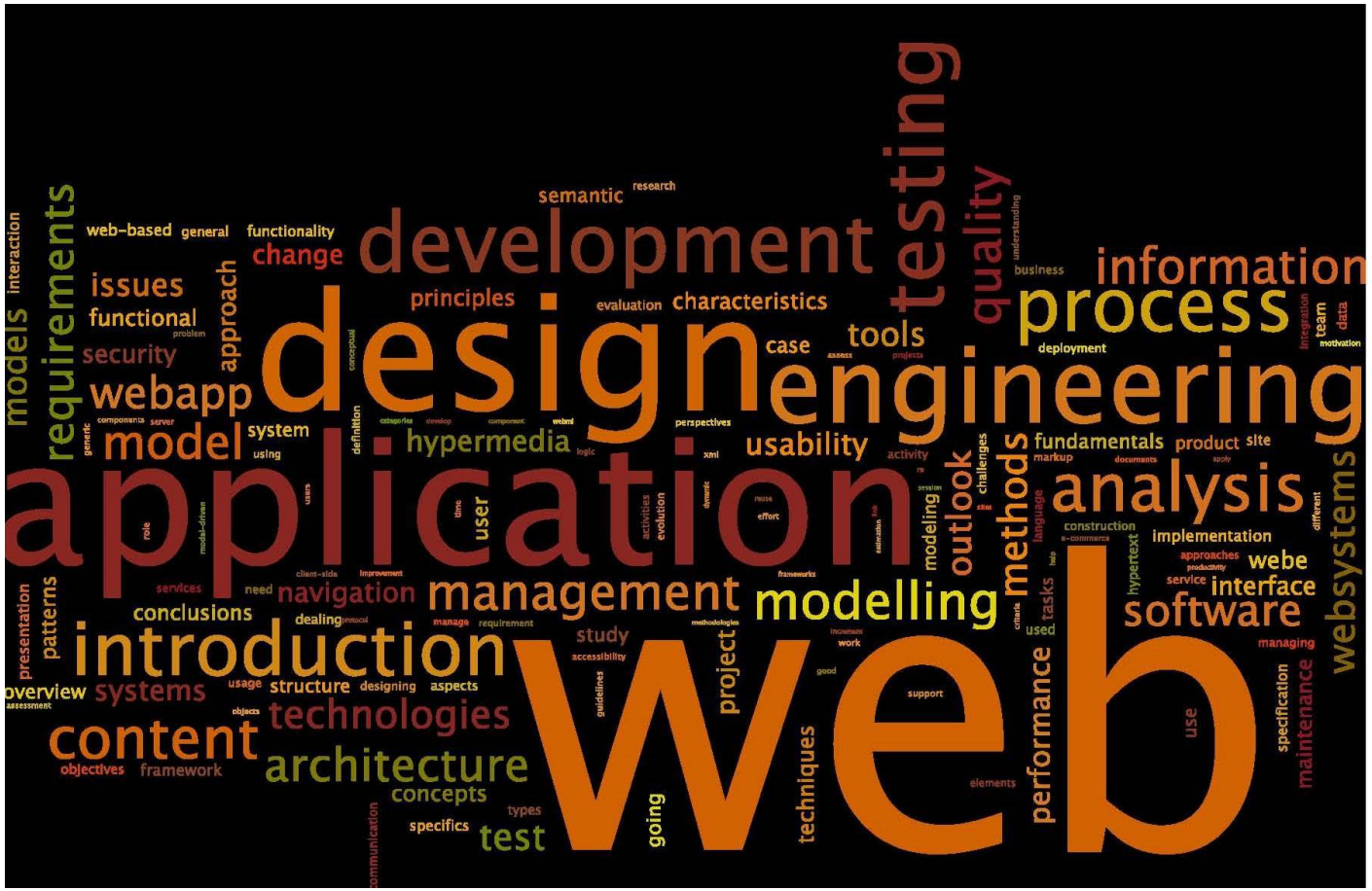
Software Engineering
A Practitioner´s Approach
R. S. Pressman,
McGraw-Hill 2005, 6th edition



Web Engineering: A Practitioner's Approach
R. Pressman, D. Lowe
McGraw Hill, 2008

Web Engineering

Tag Cloud of WE Related Terms



[B. Pröll, S. Reich:
An Analysis of Textbooks for Web Engineering, WECU 2010]

Web Engineering

Defining Web Applications

*A Web application is a **software system** based on **technologies** and **standards** of the World Wide Web Consortium (**W3C**) that provides **Web specific resources** such as **content** and **services** through a **user interface**, the Web browser.*

Point of discussion:

- *Software aspect* – i.e., static HTML pages are not Web applications ?
- *Interface aspect* – i.e., Web services are not Web applications ?

Web Engineering
Development of Web Applications (1/3)

- Situation of Web application development reminds us of the software development practices of the 1960s:
  - Often seen as a **one-time event**
  - Often **spontaneous**
  - Usually based on the knowledge, experiences and development practices of **individual developers**
  - Limited to recycling in the sense of the **"Copy&Paste paradigm"**
  - Characterized by **inadequate documentation** of design decisions
  - Often **no project management**

> „The world has glimpsed a subset of hypermedia functionality and its potential for structuring and accessing information through the recent surge in World Wide Web (WWW) activity. Yet, **we lack guidelines and tools** to design and develop hypermedia applications"
> (Bieber and Isakowitz, 1995)

Web Engineering
Development of Web Applications (2/3)

- Encountered problems of large scale Web projects
  - Failure to meet business needs
  - Project schedule delays
  - Budget overrun
  - Lack of functionality
  - Poor quality of deliverables
  - Lack of maintainability

***Source**:* Cutter Consortium, *Poor Project Management Number-one Problem of Outsourced E-projects,* Cutter Research Briefs, November, 2000, http://www.cutter.com/research/2000/crb001107.html

Web Engineering
Development of Web Applications (3/3)

- Reasons for encountered problems
  - **Assumed simplicity of development**
    - Document centric approach: development is seen as an authoring activity creation and linking of web pages and inclusion of graphics; but: requirements changes from simple hypertext to complex interactive processes
    - Due the availability of different tools, such as HTML editors or form generators
  - **Know-how from related disciplines is not used**
    - Common misconception that the development of Web applications is analogous to the development of traditional applications
    - Know-how from disciplines which could be used is not applied (e.g., Hypermedia and Human Computer Interaction)
  - **Technology evolution**

    „You must always be prepared to be surprised ...
    by completely new types of applications“ [Deshpande]
  - **Short history of Web Application Development**

Web Engineering

Defining Web Engineering

*(1) Web engineering is the application of **systematic** and **quantifiable approaches** (concepts, methods, techniques, tools) to **cost-effective requirements analysis**, **design**, **implementation**, **testing**, **operation**, and **maintenance** of high-quality Web applications.*

*(2) Web engineering is also the **scientific discipline** concerned with the study of these approaches.*

***Sources**:*

- Bourque, P., Dupuis, R., Abran, A., Moore, J. W., Tripp, L. L. *Guide to the Software Engineering Body of Knowledge*, IEEE Computer Society, 2004
- Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M., White, B., *Web Engineering*, Journal of Web Engineering, 1 (1), 2002, pp. 3-17.

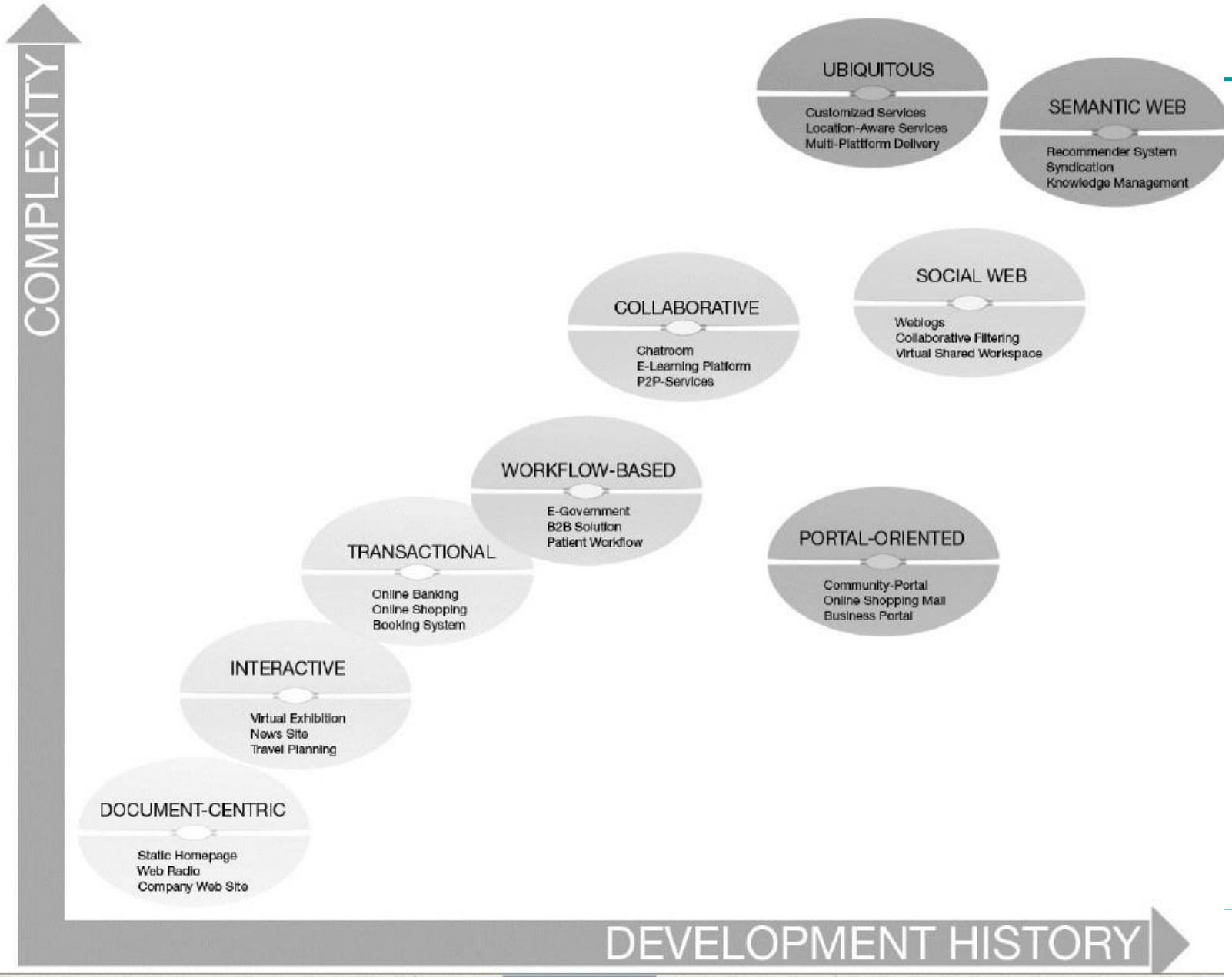
Web Engineering
Basic Principles

- Clearly defined **goals** and **requirements**
- Systematic development of a Web application in **phases**
- **Careful planning** of these phases
- Continuous **audit** of the entire development process

***Source**:* Lowe, D., *Engineering the Web - Web Engineering or **Web Gardening**?* WebNet Journal, 1 (1), January-March, 1999.

Web Engineering

Categories of Web Applications (1/2)

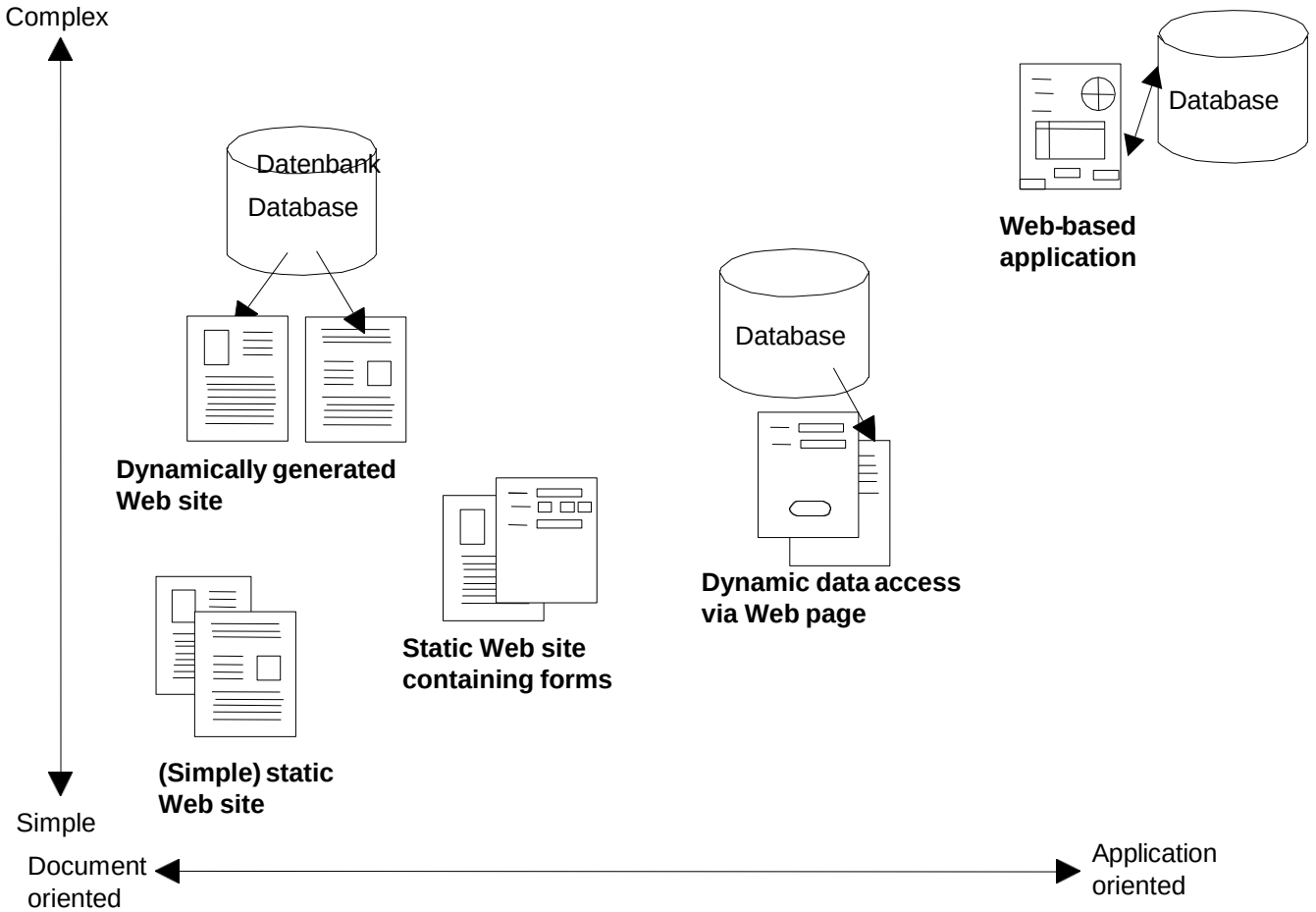


Web Engineering

Categories of Web Applications (2/2)

- **Any of these categories** is still valid today
  - each has its own specific fields of application
- **Newer categories** are generally **more complex**
- Complex Web applications can be assigned to **several categories at once**, e.g. Online Shopping Malls
  - offer different search options (interactive)
  - allow to buy products (transactional)
  - offer order status monitoring (workflow-based)
  - offer online auctions (collaborative)
- Web applications may cover many **traditional fields of application**, e.g., online banking
- **New fields of applications** are created, e.g., location-dependent services

Web Engineering
Web Application Architectures: [Powell98] Classification



Web Engineering
Conclusions

- Web applications **differ** from traditional, not Web-based applications, **in a variety of features => characteristics**
  - that traditional applications **lack completely** (e.g. non-linear navigation)
  - that are **particularly pronounced** in Web applications (e.g. frequency of updates)
- **Presence and strength** of a certain characteristic **depend partly on the type of Web application**, e.g., e-commerce systems vs. digital libraries

=> **proven methods** from **related disciplines** (e.g., software engineering, HCI or Hypermedia)

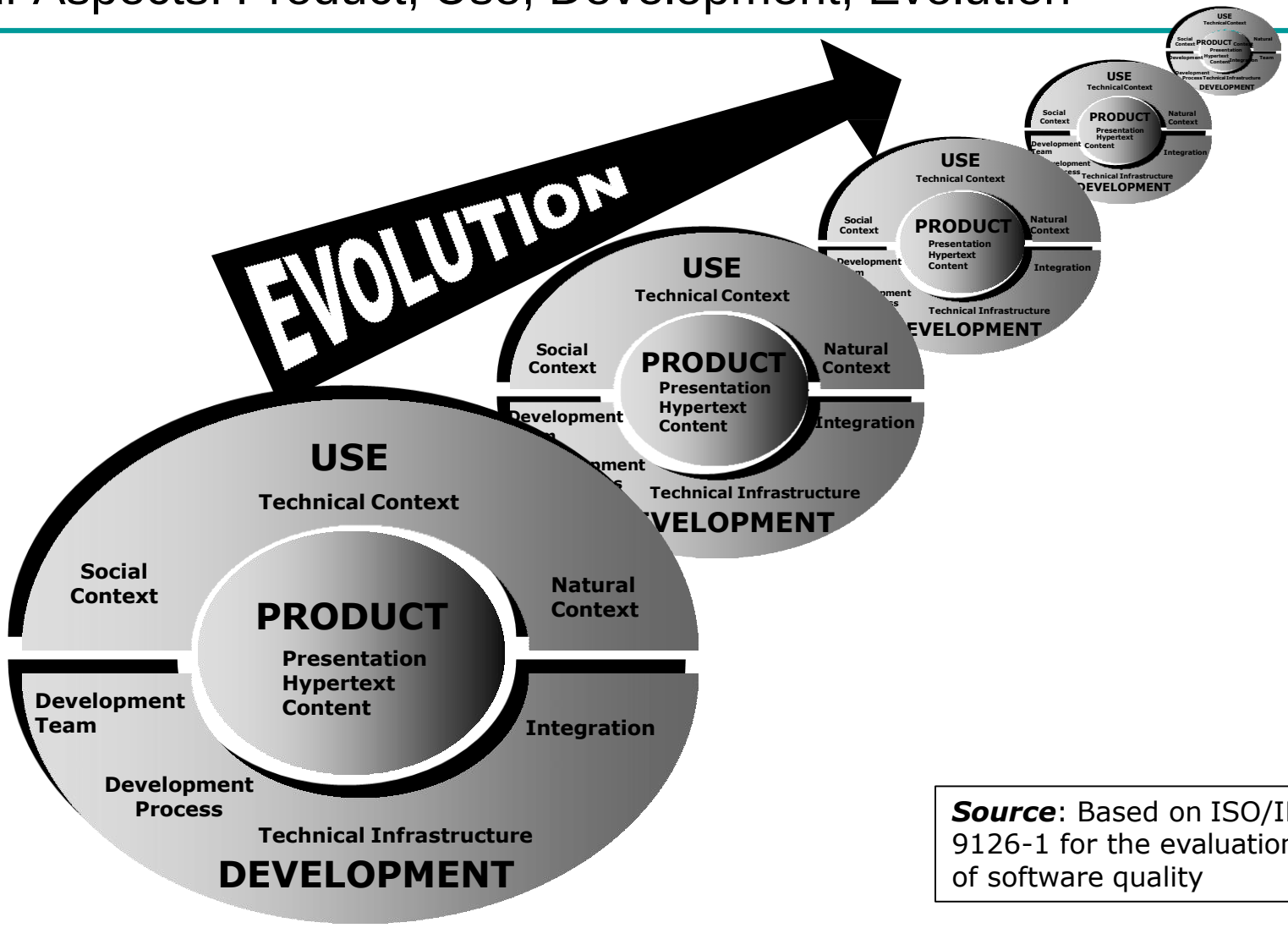
- should be **taken as they are**
- **have to be adapted** to the needs of Web Engineering
- **new solutions** have to be developed

Contents of the Course

- Tiscover
- Internet & World Wide Web
- Introduction to WE & Characteristics of Web Applications
- WE Tasks and Phases
- Web Modelling
- Web Usability
- Web Performance/Caching
- Personalization/Adaptation
- Web 2.0

Characteristics of Web Applications

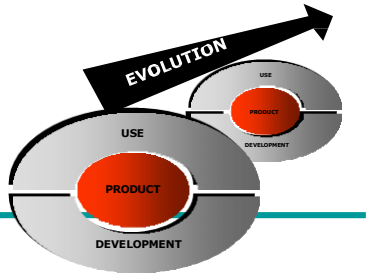
Four Aspects: Product, Use, Development, Evolution



Source: Based on ISO/IEC 9126-1 for the evaluation of software quality

Characteristics of Web Applications

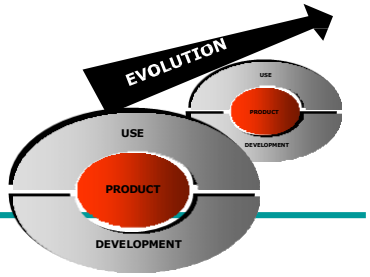
Product-related Characteristics (1/3)



- **Content**
  - **Document centric character and multimediality**
    - content is provided as tables, text, graphics, animations, audio, or video
    - documents are generated in an appropriate way
    - special requirements on usability
  - **Quality Demands**
    - being up to date, exact, consistent, reliable, …
    - high quality is required for price and availability information in online-shopping systems
    - critical factor for the acceptance of a Web application

Characteristics of Web Applications

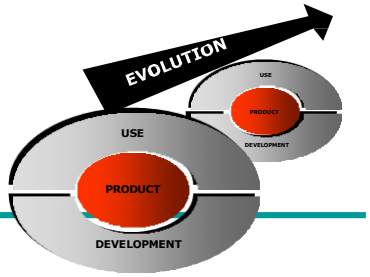
Product-related Characteristics (2/3)



- **Hypertext**
  - **Non-linearity**
    - differ from traditional software applications by the possibility of systematic reading (”browsing”, ”query”, ”guided tour”)
    - move freely through the information space, depending on interests and previous knowledge
    - a challenge for the authors
  - **Disorientation** and **Cognitive Overload**
    - **Disorientation**: the tendency to lose one´s bearings in a non-linear document
    - **Cognitive overload**: concentration required to keep in mind several paths or tasks simultaneously

Characteristics of Web Applications

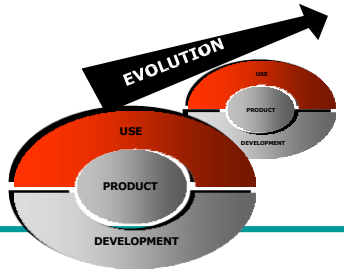
Product-related Characteristics (3/3)



- **Presentation**
  - **Esthetics**
    - “Look and Feel” of the user interface
    - fashion trends
  - **Self-explication**
    - usage without documentation
    - navigation and interaction behaviour must be consistent within the whole application

Characteristics of Web Applications

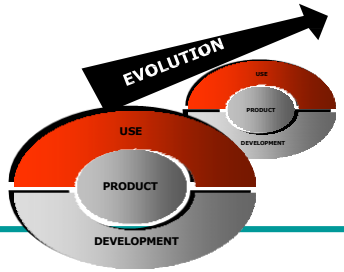
User-related Characteristics (1/2)



- **Social Context: Users**
  - **Spontaneity**
    - users can visit a Web application whenever they want and leave it again
    - the number of users cannot be reliably predicted
    - scalability is extremely important
  - **Multiculturalism**
    - development for different user groups (known vs. anonym)
    - hard to define a representative sample for requirements analysis

Characteristics of Web Applications

User-related Characteristics (2/2)

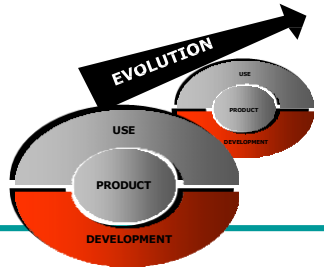


- **Technical Context: Network and Devices**
  - **Quality of Service**
    - unknown network properties
  - **Device-Independent Delivery**
    - devices with very different specifications

- **Natural Context: Place and Time**
  - **Globality**
    - internationalisation regarding regional, cultural and linguistic differences
    - location dependent services
    - increased demands on security
  - **Availability**
    - immediate and permanent availability (24x7)
    - time-dependent services

Characteristics of Web Applications

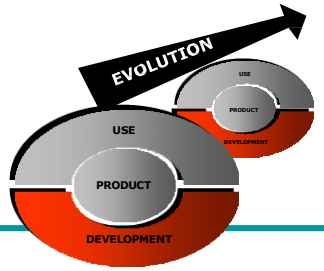
Development-related Characteristics (1/3)



- **Development Team**
  - **Multidisciplinarity**
    - mixture between print publishing and software development, between marketing and computing, between art and technology
    - IT experts, hypertext experts, designers, and application experts are responsible
  - **Young average age**
    - Web application developers are on average significantly younger and less experienced, care not too much about old conventions
  - **Community Development**
    - open source software freely available on the Web

Characteristics of Web Applications

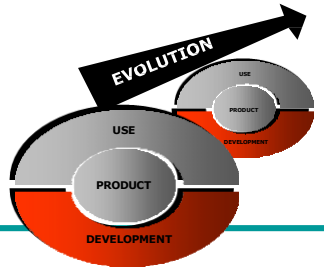
Development-related Characteristics (2/3)



- **Technical Infrastructure**
  - **Inhomogeneity**
    - two external components
      - server (usually configured and operated as desired)
      - browser (no influence on preferences)
  - **Immaturity**
    - increasing time-to-market pressure
    - Bugs
- **Development Process**
  - **Flexibility**
    - no rigid, predefined project plan
  - **Parallelism**
    - parallel development of application parts
    - parallel running of phases

Characteristics of Web Applications

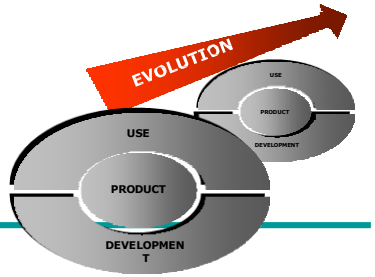
Development-related Characteristics (3/3)



- **Integration**
  - **Internal Integration** with existing legacy systems
  - **External Integration** of content and services of external Web applications
    - large number of frequently changing sources
    - high degree of autonomy concerning availability and schema changes
    - few details about the properties of these sources
    - heterogeneity at various levels (data level, schema level, data model level)

Characteristics of Web Applications

Evolution-related Characteristics



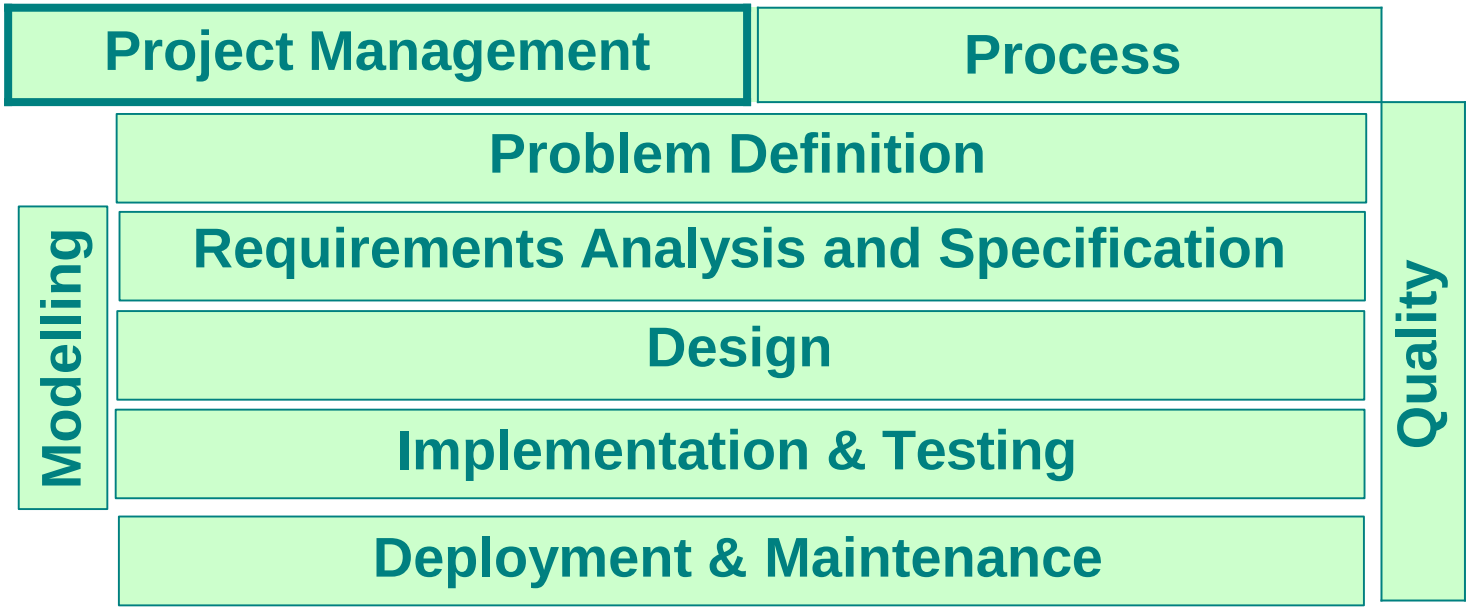
- **Continuous Change**
  - permanent evolution due to constantly changing requirements or conditions
  - changes may concern all three dimensions of a Web application – product, use, and development
- **Competitive Pressure**
  - shorter product lifecycles and extremely short development cycles
  - no room for a systematic development process
- **Fast pace**
  - extreme time pressure due to the rapid change on the Web
  - ”lean” versions of traditional processes with special emphasis on requirements analysis/specification and operation/maintenance

Contents of the Course

- Tiscover
- Internet & World Wide Web
- Introduction to WE & Characteristics of Web Applications
- WE Tasks and Phases
- Web Modelling
- Web Usability
- Web Performance/Caching
- Personalization/Adaptation
- Web 2.0

WE Process and Tasks

Overview



WE Process and Tasks

Project Engineering: Project Team

- **Project team** is predominantly small and consists of young people
  (low experienced but highly motivated!)
  - Customers, application specialists
  - Project engineer (project manager)
  - Web visionary
  - Web administrators
  - Web technologist („technology observer“!)
  - Web programmers, code checkers
  - „Design“ / multimedia experts
  - Database experts
  - Security experts
  - Network experts
  - Test team
  - Marketing Team
  - Usability experts
  - Law experts
  - Business Experts
  - Sociologists, Psychologists
- **Project Communication**!
  - Conflict management and use of a common language

WE Process and Tasks

Project Engineering: Tasks and Challenges (1/2)

- **Short budgets**, **lack of time** („time to market“, „time to schedule“)
- **Interface to customer**: Need of explanation for customers (static vs. dynamic vs. application-oriented Web application)
- **Heterogeneous project team**
- Several parts of project developed by „universal“ teams
- => Assuring the **consistency of the whole project**!
- **Merge of development**, **deployment (operation)** and **maintenance phase**; operation/maintenance simultaneously with 24x7 availability; small, short, consecutive developments
- **Version and release management** (change requests)

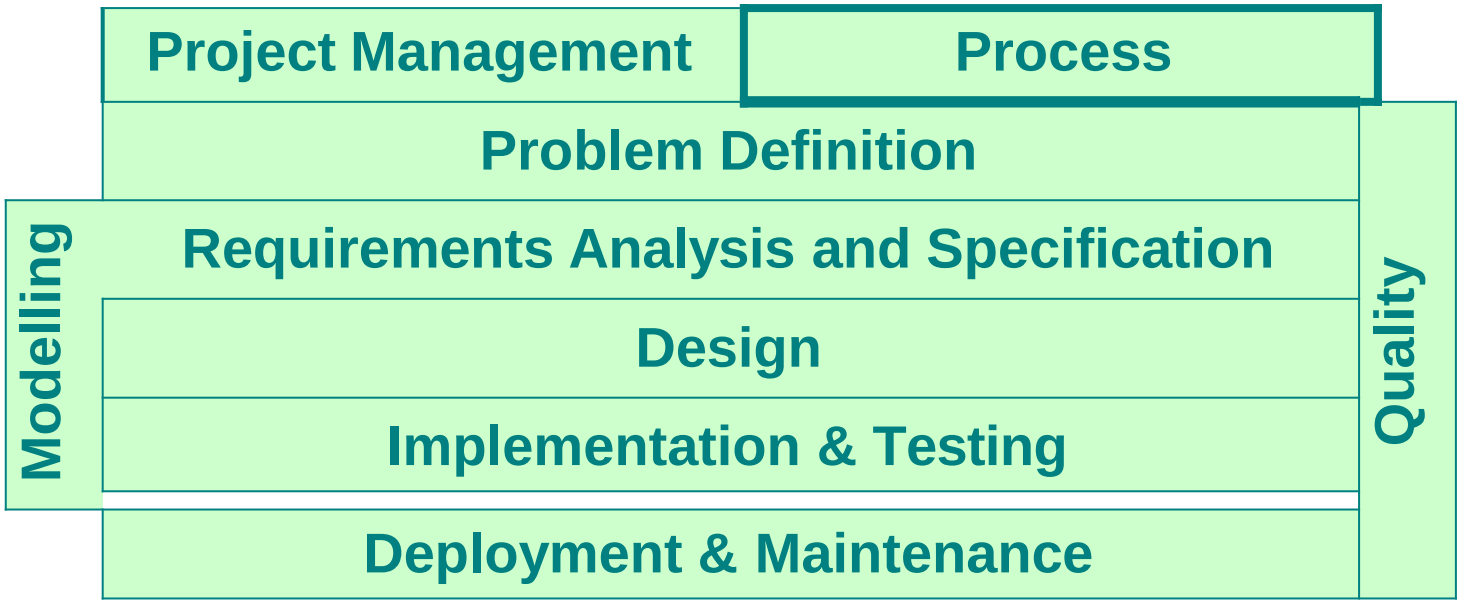
WE Process and Tasks

Project Engineering: Tasks and Challenges (2/2)

- Decision for use of **new technologies and tools**
- **Risc management** (risc identification – risc analysis etc.): use and update of basic components / outsourced components, quality of content, launch & re-launch, performance and reliability, scalability
- **Tool support**: Project management tools, requirement /change request management tools, configuration management tools

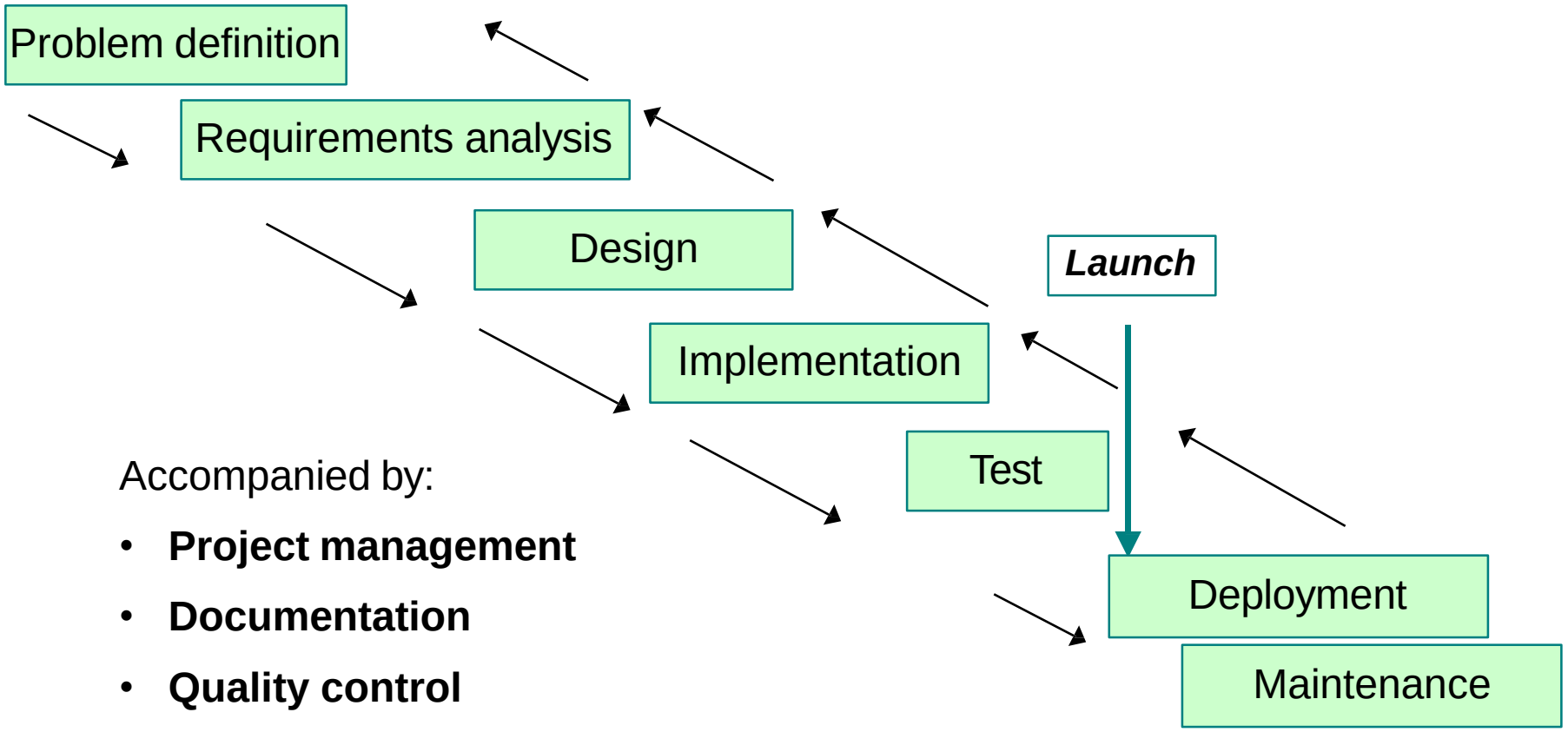
WE Process and Tasks

Process



WE Process and Tasks

Process: **Waterfall Model** (traditional)



Accompanied by:

- **Project management**
- **Documentation**
- **Quality control**

WE Process and Tasks
Process Requirements

WE Process has to cope with:

- Short development cycles
- **Changing requirements**
- Releases with fixed deadlines but variable extent (functions etc.)
- Parallel developments, reuse and integration
- Adaptability to complexity of Web application
- **Customer interaction**

➔ **Agile Process Models**

- Rational Unified Process (RUP)
- Extreme Programming (XP) [Beck99]
- Adaptive Software Development (ASD) [Highsmith00]
- Scrum [Sutherland90]

WE Process and Tasks

Process: **Agile Development** (1/2)

Idea

- **Software developers** and **stakeholders** (managers, users, end-users) co-operate in an agile self-organising team
- Focus is on **collaboration** and **communication**

Manifesto for Agile Software Development

[Beck et al.“Agile Alliance“, 2001]

- **Individuals and interactions** over process and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

WE Process and Tasks

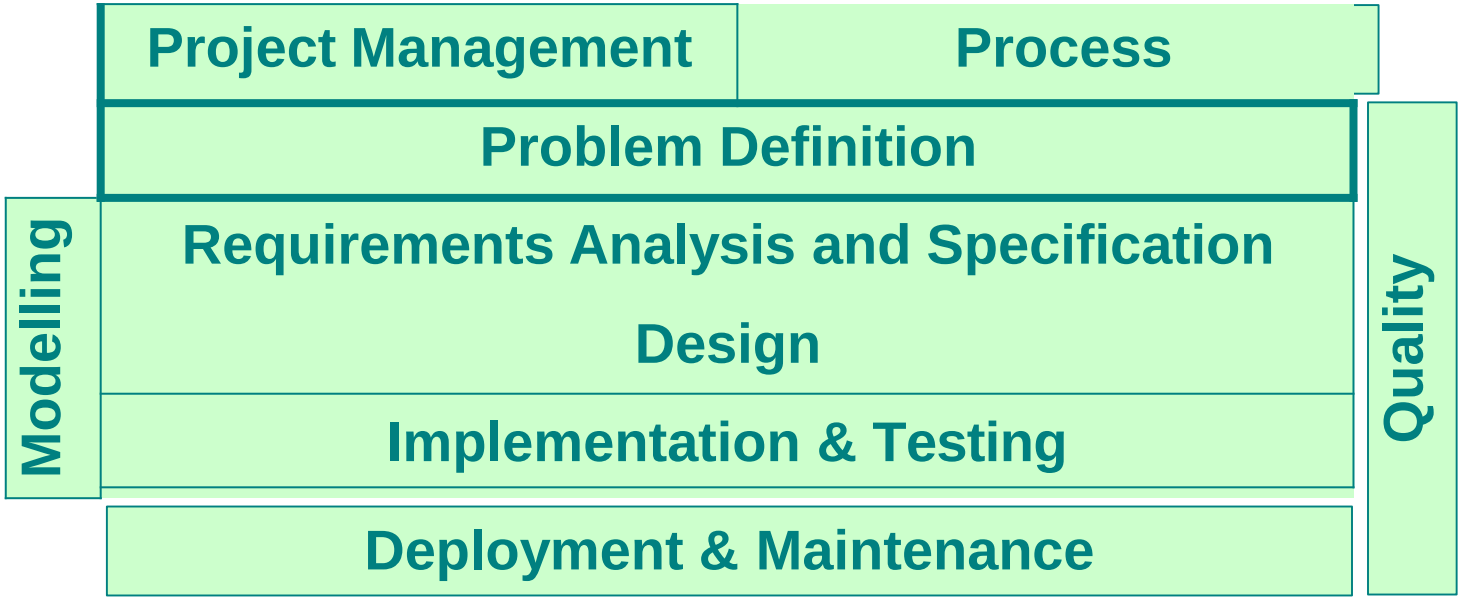
Process: Agile Development (2/2)

- **Satisfy customer** through valuable software
- **Welcome changing requirements**, even late in development;
- care for customer´s competitive advantage
- Deliver working software frequently
- Business people and developer must **work together** daily
- Build projects around **motivated people**
- Support face-to-face **communication**
- Working software is the primary measure of progress
- Reflect at regular intervals how to become more effective
- etc.

[Pressman, page 73ff.]

WE Process and Tasks

Problem Definition



WE Process and Tasks

Problem Definition (1/2)

- **Project Intention**
  What is the intended solution of the system to be developed? What is the current state of the system (if there is one)?
  Example: simple Web presentation of a company, Web access of legacy system, e-commerce site
  Why shall we develop a Web site? (competing WebSites?) What are measures of success?
  Who are the users?
- **Market analysis and target audience analysis**
  Who are the users of the Web site? What is the equipment of the users?
- **CONTENT**
  Where do we get the content from (content sources)? What are the legal issues (copyright etc.)? Which media, languages do we use?
- **Legal Issues**

WE Process and Tasks

Problem Definition (2/2)

- **Infrastructure**
- Provider, Web Hosting, Web Housing,...
- **Business Model**
- Who pays? (marketing, customers, etc.)
- **Inhouse**-development or **outsourcing**
- **Project team**
- Availability of programmers, designers, marketing experts, network experts, etc.
- **Launch**
- Date and content; beta version?
- **Expected change requests** after launch

WE Process and Tasks

Excursus: **eLaw**

- There are a lot of legal things every Web application developer should be aware of…
- European Community: **E-Commerce directive**
  - http://ec.europa.eu/internal_market/e-commerce/index_en.htm
  - (…only one of a number of laws, which are important for t… domain)
- **Nice-to-knows**
  - Illegal (at least according to Austrian law) (by 2…
    - Photos on Web sites of less than 7 per… without
    - Logos of companies
    - Not implemented!: Deep links (… ily all… pages)
      Exception: search engines (… ogle
    - Copy & paste of URL li…
  - E-Commerce: imprints… ing compan… e, address etc. must be accessible
  - Web designers can be sued

- https://www.spiegel.de/reise/aktuell/0,1518,562217,00.html
- http://rorschachstagebuch.wordpress.com/2008/07/11/lg-hamburg-entsc

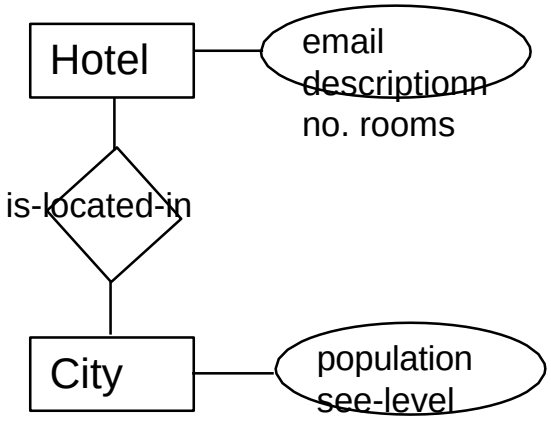
WE Process and Tasks

Modelling

WE Process and Tasks

Modelling

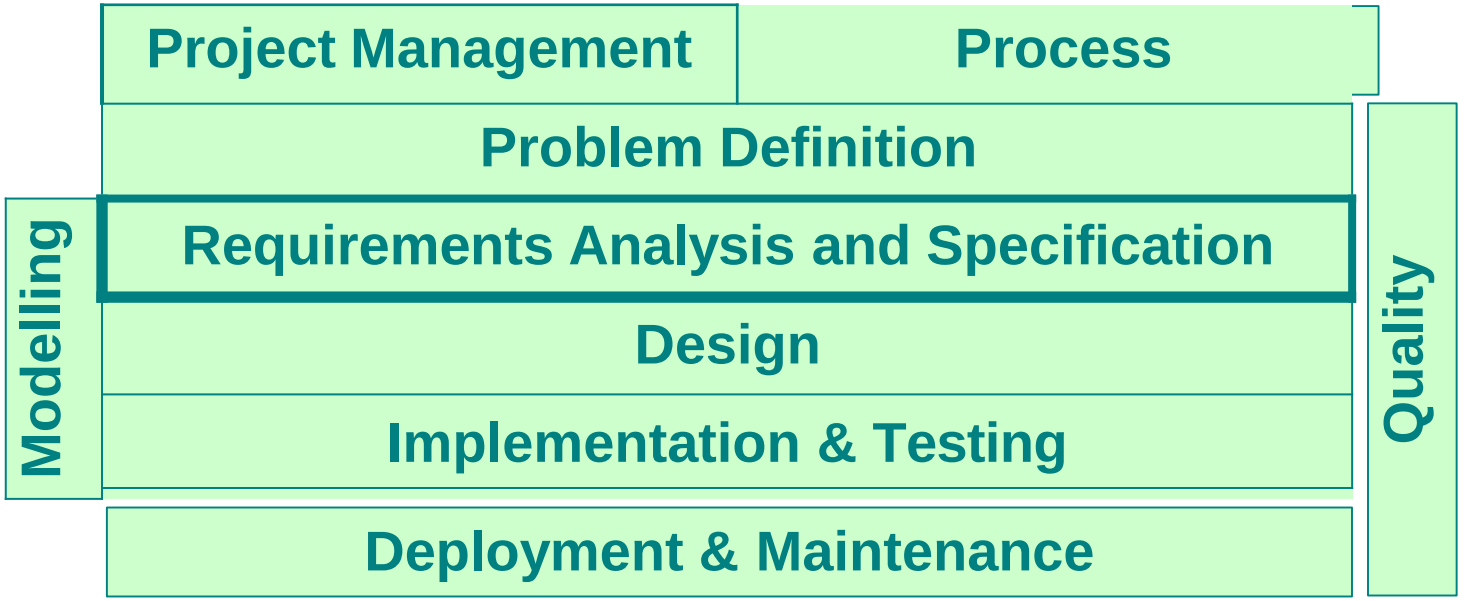
- **Modelling**
  - Abstraction of the (essential part of the) real world
  - Often graphical representation
  - Focus on requirement analysis and technical design; but universally applicable
  - Examples: data model, navigation model, flowchart, etc.



ER-Diagram

WE Process and Tasks

Requirements Analysis and Specification



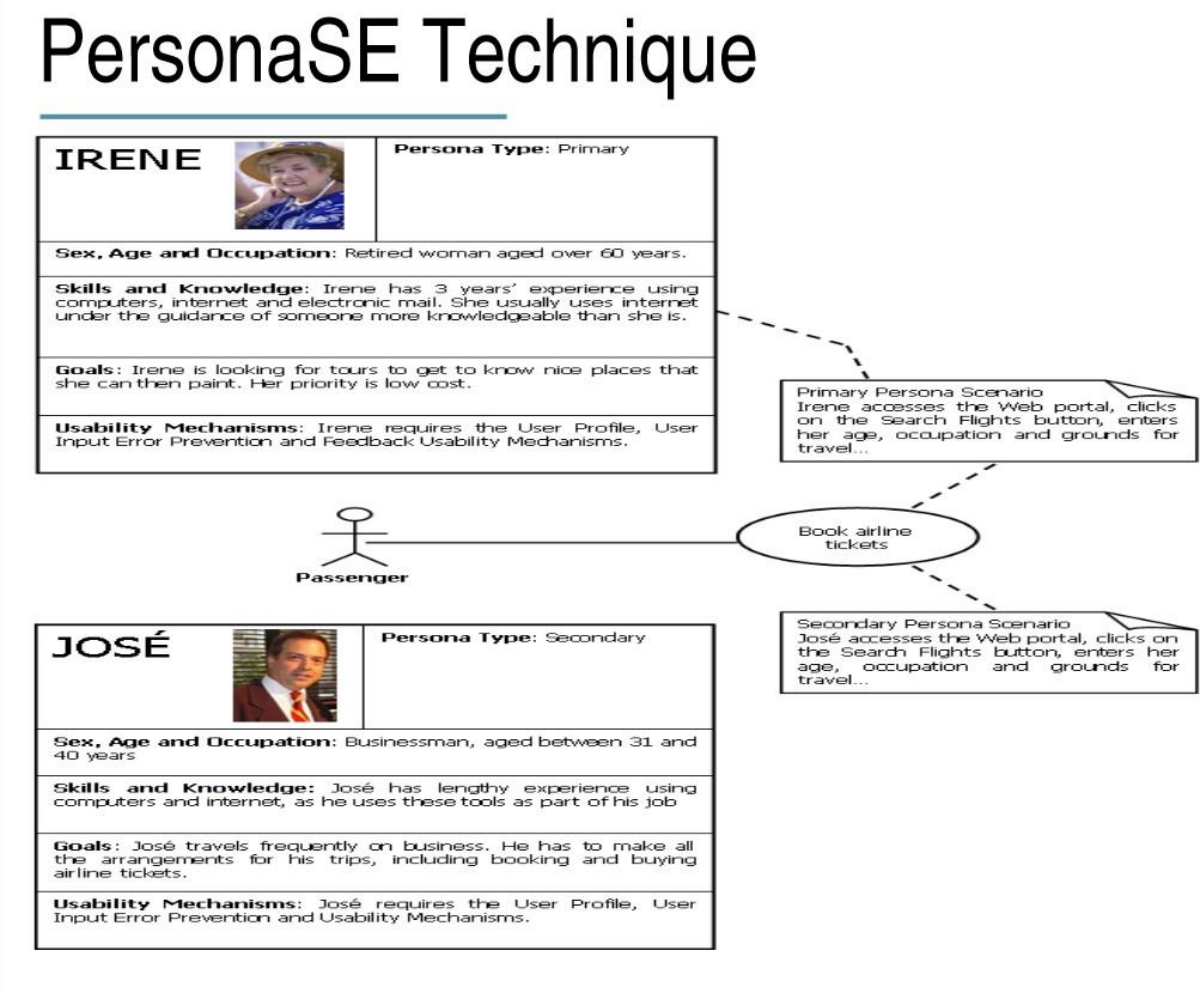
WE Process and Tasks
Requirements Analysis and Specification

- **Outcome of requirement analysis/specification** (requirement engineering)
  - Collection and negotiation of requirements
  - Documentation of requirements
- Estimation and resource planing
- Often provides basis for legal contract

- **Requirement types**
  - Functional requirements: presentation of information, search, business processes
  - Content – hypertext - presentation (layout) requirements
  - Quality requirements
  - Requirements with respect to the system environment
  - HCI (human computer interface requirements) => usability
  - Requirements with respect to evolutionary aspects
  - Project constraints

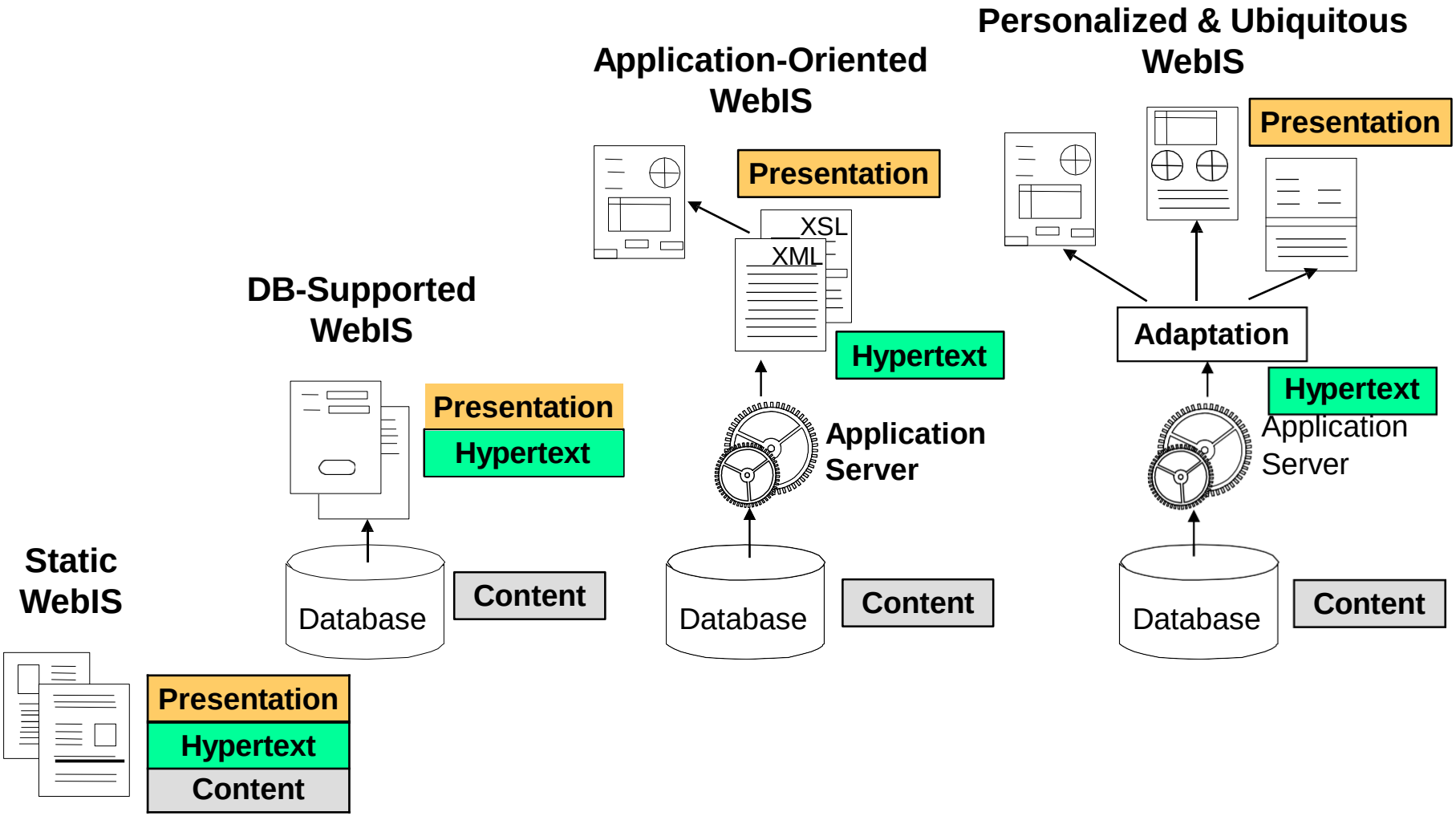
WE Process and Tasks

Who are the Users? => Personas



WE Process and Tasks

Excursus: Content, Hypertext, Presentation



WE Process and Tasks

Requirements: Content, Hypertext, Presentation

- **Content requirements**
  - **Object- (document) analysis**
    - Identification of object types (document types: articles, books, etc.)
    - Identification of object attributes
  - **Content quality**
    - Actuality, precission, completeness, trustworthiness, consistency, etc.
- **Hypertext requirements**
  - Navigation requirements
    - Metapha (map, book, etc.), hierarchy, process, browsing and searching, linking, access, etc.
- **Presentation requirements**
    - Icons, symbols, graphics, colours, new media (video, audio, animation), embedded programms, adaptable presentation (e.g., people with special needs)
    - Marketing and corporate identity

WE Process and Tasks

Requirements: System Environment

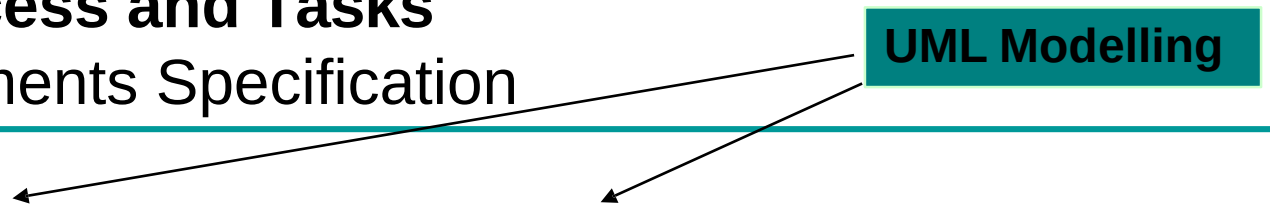
- **Client-side requirements**
  - Browser, operating system, grafic card, plug-ins, etc.
- **Server-side requirements**
  - Hardware: Web server, database server, proxy server, etc.
  - Computing capacity, memory, etc.
  - Performance, number of (simultaneously accessing!) users
- **Network infrastructure**
  - Network capacity and reliability
- **Security aspects**
  - Resticted parts of Websites? => authorization
  - Sensible data, privacy aspects? => encryption
- **Environment:** hardware for operation, development, test, user training
- **Legacy systems,** open souce components

WE Process and Tasks

Requirements: Project Constraints

- Budget and time constraints
- Technical constraints
- Standards
- Predetermined technologies
- Legal issues
- etc.

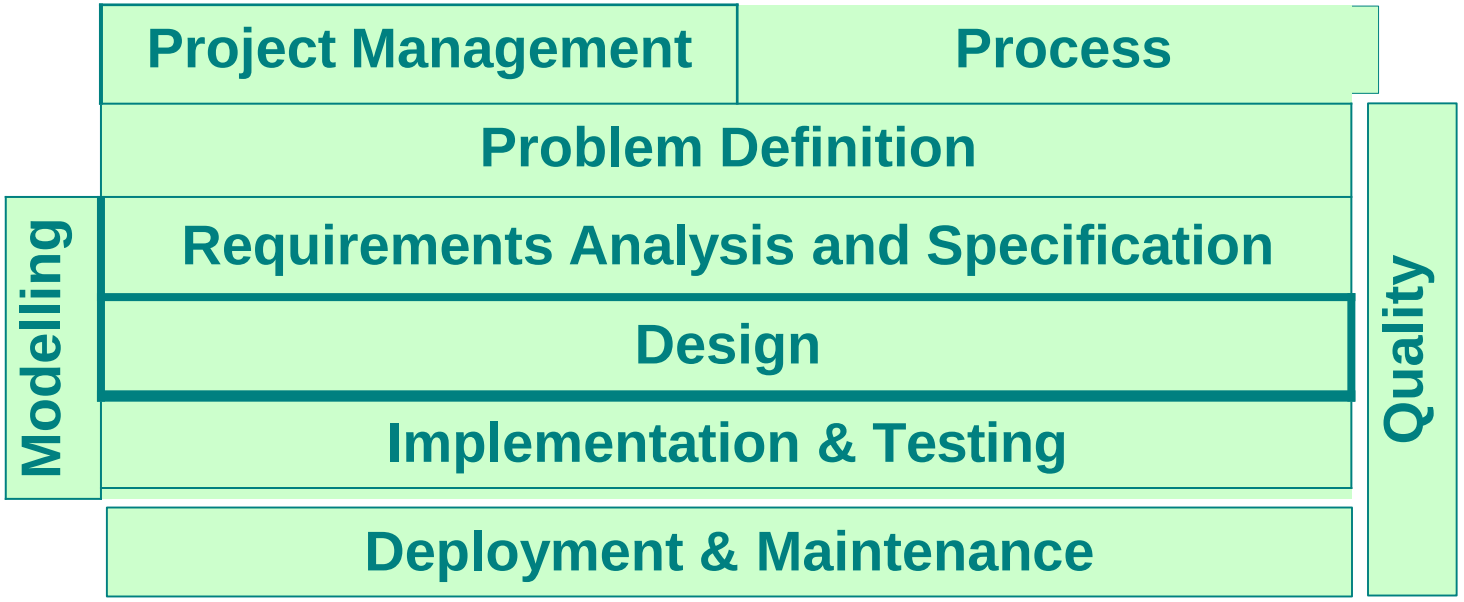
WE Process and Tasks
Requirements Specification



- **UML use cases** and **activity diagrams**
- **Stories** (e.g., user stories produced in XP)
  - Example: A user checks the products she put in the shopping chart. The input is validated as soon as the user clicks <Continue>. If no error is found the order will be accepted and a confirmation email will be sent to the user.
  - Advantage: less effort, useful communication with non-experts
- **HTML prototype**
  - IKIWIS (I know it when I see it)
  - No functionality, no database access
  - **Prototype review** and maybe re-prototyping
- **Specification document** (containing Screenshots)

=> Different methods might be applicable for different parts of the Web application

WE Process and Tasks
Design



WE Process and Tasks

Technology Decision

- **Domain specific product**
  - E.g. ec-shop solution
  - => Adaptability / individuality might be restricted
- **CMSTool**
  - => Which one?
- **Framework / Application Server**
  - z.B. Cocoon, WebSphere, Struts, ~~San Francisco (IBM)~~, RubyOnRails
  - => Product dependency
- **Open source**
- **Inhouse development**
  - CGI, JSP, PHP,…

➔ Costs & product dependency versus development from scratch & individuality

➔ Might often be a combination of outsourced components and self-development

WE Process and Tasks

Content – Hypertext (Structure/Navigation) – Presentation Design

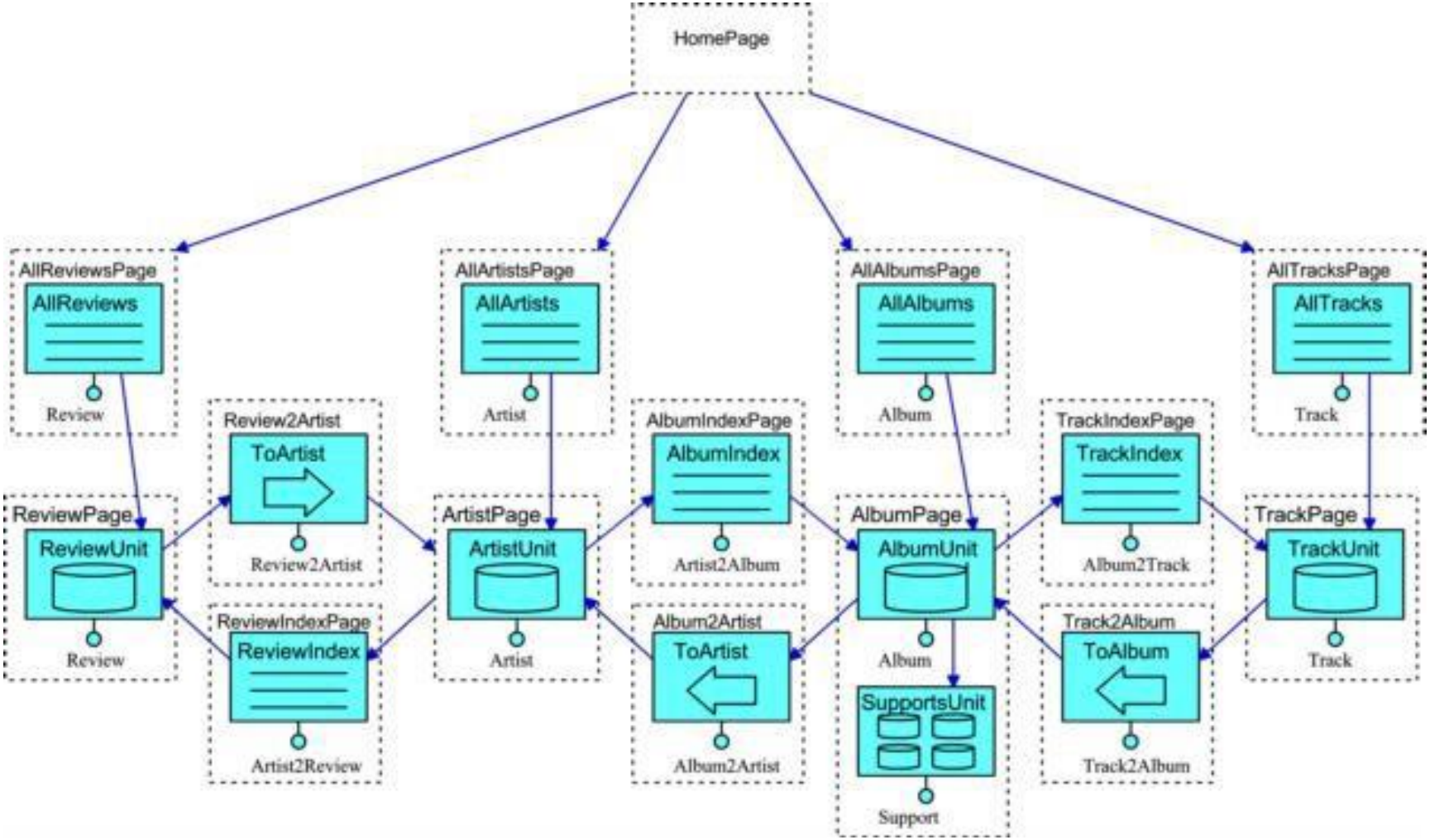
- **Content design** (data model, directory)
  - Attributes and relations (same as in conventional database related systems)
- **Structural design** (Web page design)
  - Web page design: Which Web pages do we need?
  - Node design: Which attributes shall be allocated to a certain Web page?
- **Navigation design**
  - How shall Web pages (resp. information) be „linked“?
    - Simple links
    - Index links (lists, menues)
  - Offset links (within a Web page)
  - Frames
- **Presentation design**

Hyptertext Design

=> Web Modelling

WE Process and Tasks

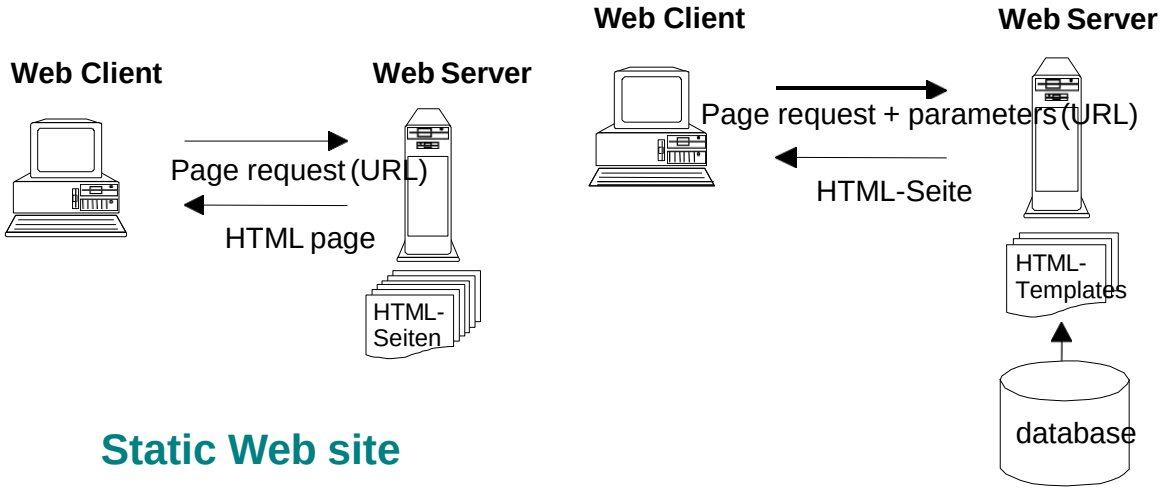
Excursus Web Modelling – Example WebML



WE Process and Tasks

Architectural & Technology Design

- **Architectural & technology design**
  - Programming language, class library etc.
  - Server-side or client-side application
  - Static or dynamic application

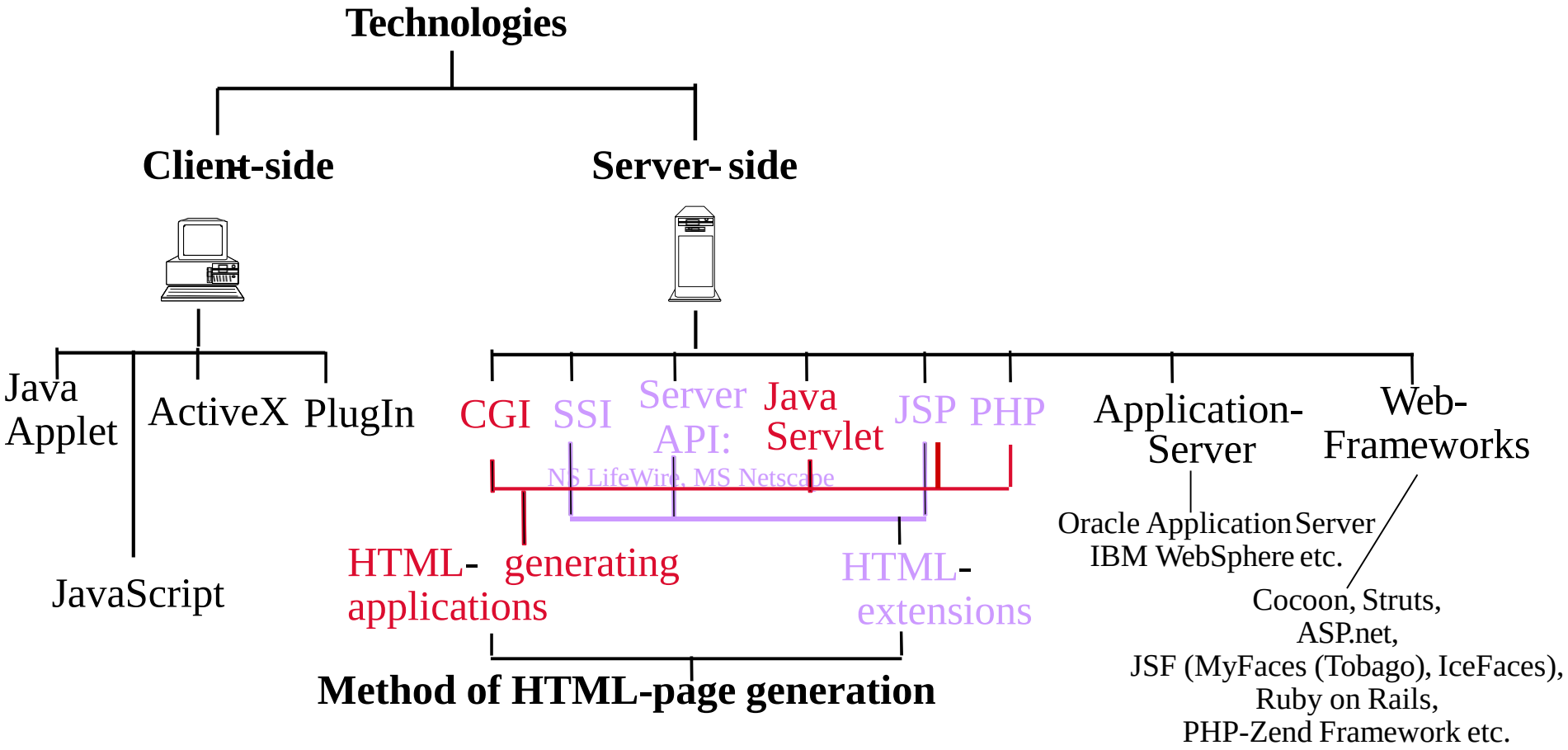


Static Web site

Dynamic Web application

WE Process and Tasks

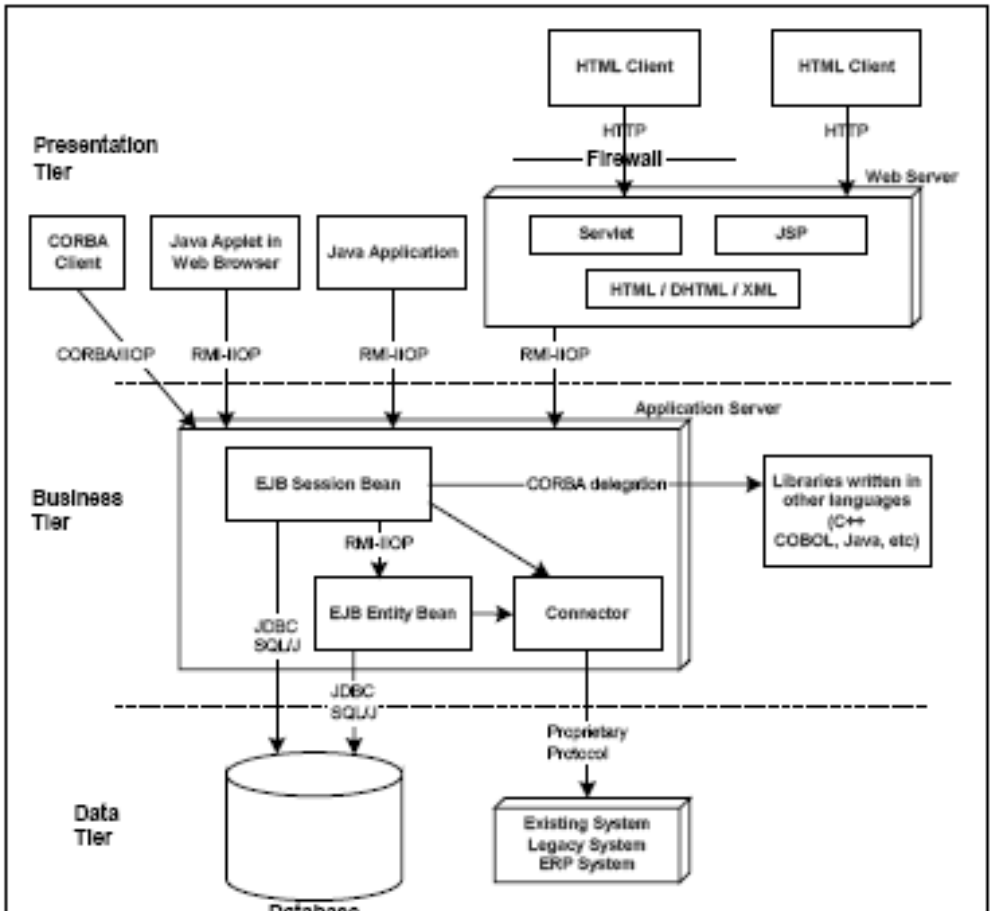
Architectural & Technology Design



J2EE Architecture "Model View Controller" (MVC-) Pattern

SUN: White Papers on Java EE (Java Platform Enterprise Edition):

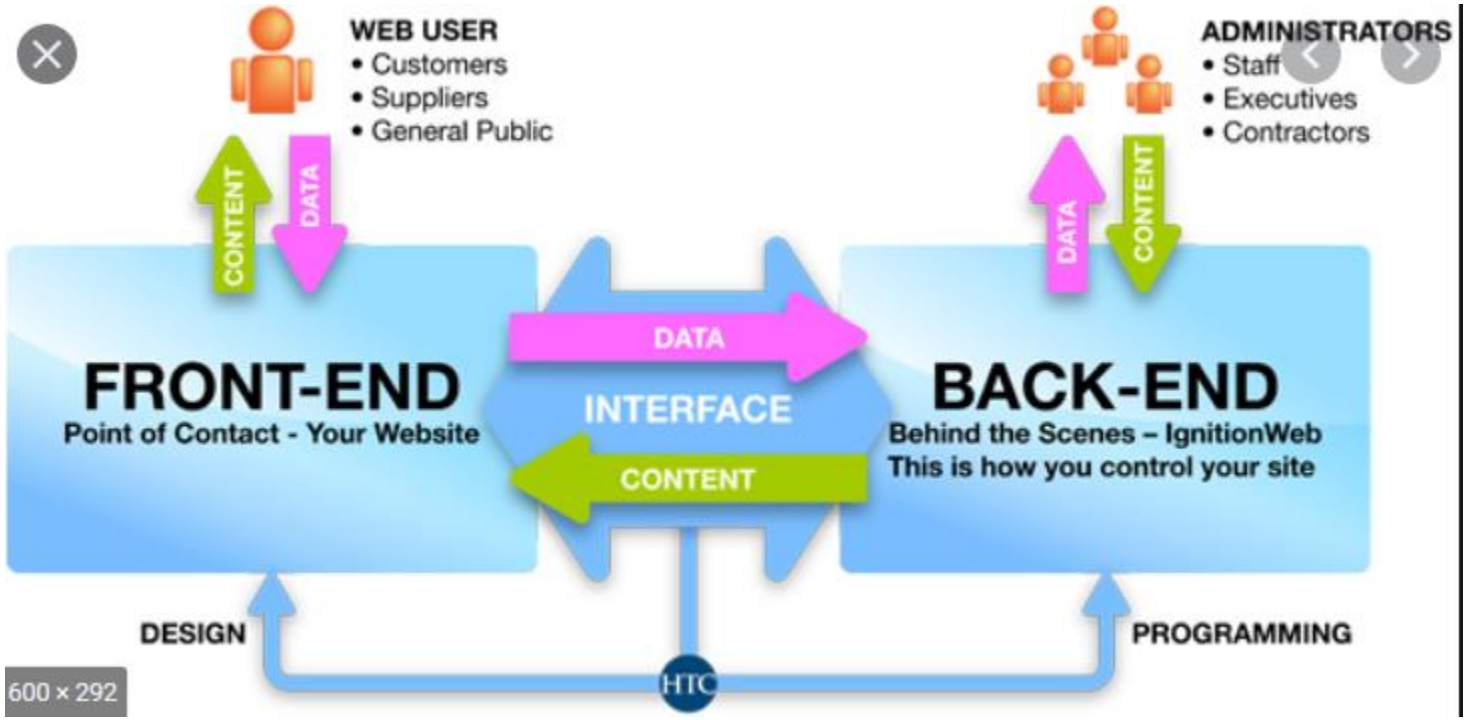
Java Servlets, JSP, J2EE, etc.: http://java.sun.com/javaee/overview/whitepapers/index.jsp



WE Process and Tasks

Architectural & Technology Design

Frontend – Backend Architecture



https://www.3nytechnology.com/website-frontend-and-backend/

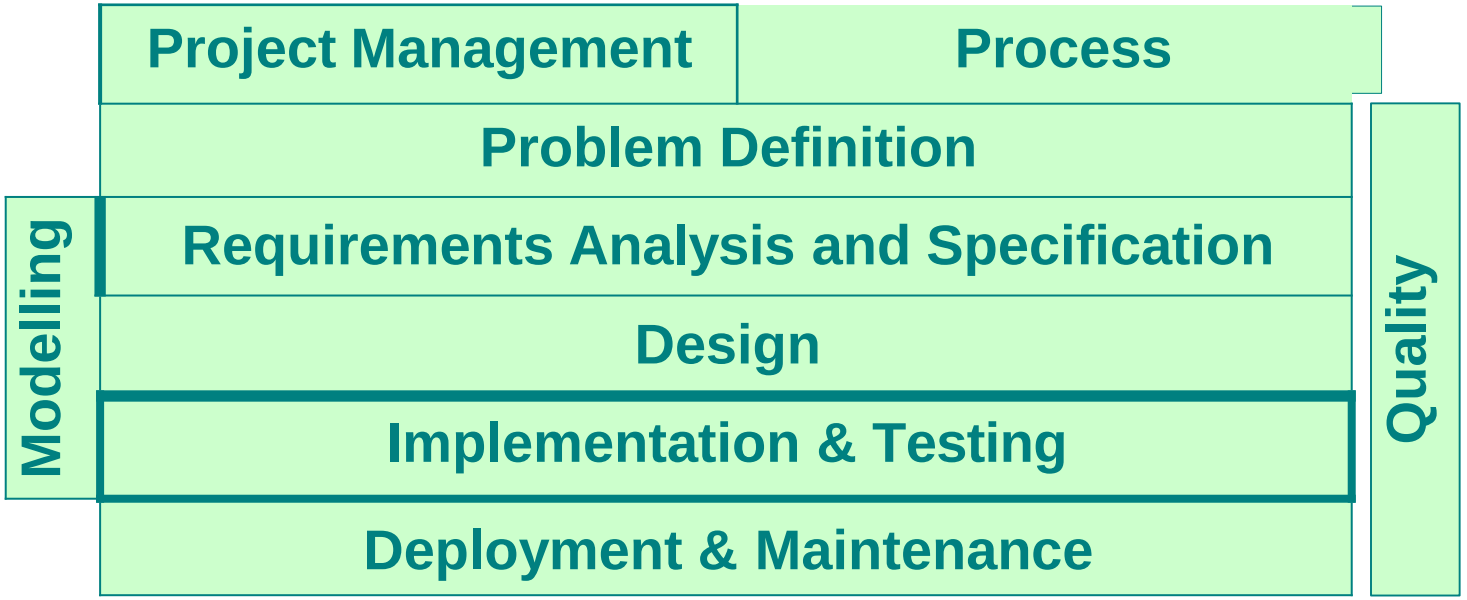
WE Process and Tasks

More Design Tasks…

- **Database design**
  - Data types, constraints
  - Data replication, triggers
  - (e.g., to meet performance requirements)
  - => No-SQL database
- **Network/server design**
- **Graphical design / layout (presentation)**
  - Fonts, style guides, etc.
  - Consistency!
- **Security Design**
  - Authorisation: Access models (e.g. RBAC – role based access model)
  - Authentification: User/PW?
  - Which firewall, encryption algorithm etc.

WE Process and Tasks

Implementation & Testing



WE Process and Tasks

Implementation

- **Preparation**
  - Evaluation and decision for development tools (e.g., Eclipse)
  - Decision for a versioning tool (e.g., Github, Subversion)
  - Quality guidelines: documentation, programming conventions, naming conventions, URL conventions, directory, etc.
- **Ongoing**
  - Class libraries
  - Technology “observer“ (blogs, news groups)
  - (Maybe) revisions of technology decision
  - Project management, documentation, quality management

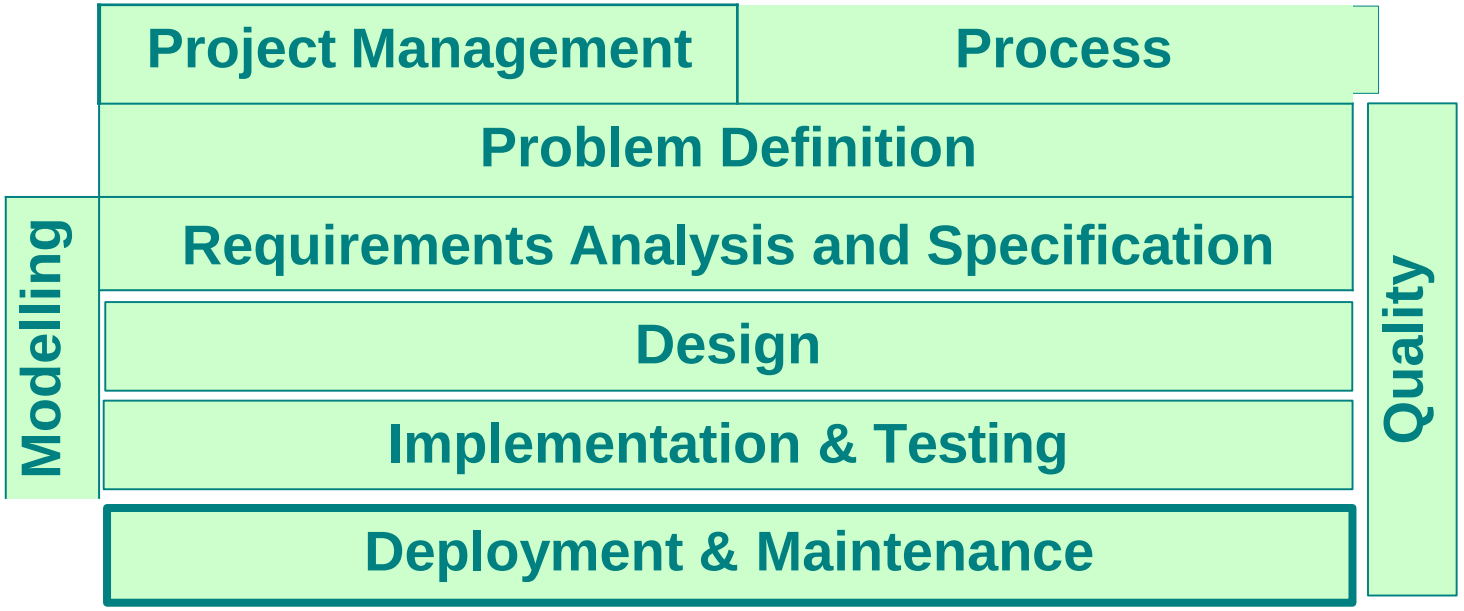
WE Process and Tasks
Testing

- **Testing**
  - Functionality tests
  - Launching of beta-version? (users become testers)
  - Browser compatibility (browser versions / operating systems)
  - Usability tests
  - Accessibility tests
  - Test environment!
  - Performance tests!: Simulation of user access: StressTools

- **Evaluation (before launch)**
  - User surveys (GUI etc.)

WE Process and Tasks

Deployment & Maintenance



WE Process and Tasks

Deployment: System Integration

- **Integration into organization**
  - Often the integration of a Web application (e.g., e-commerce system) turns out in a **re-organisation of processes** within the company
- **Additional staff**
  - System administration
  - Helpdesk
  - User training (if necessary)
- **Content**
  - Launch with Content!, Content syndication
  - Content maintenance, Content quality assurance
- **Translation effort** in case of multilingual Web application

WE Process and Tasks

Deployment: Marketing

- **Conventional marketing**
  - Marketing brochures have to be ready when system is launched
- **Webvertising**
  - Banner, link exchange etc.
- **Search engines**
  - Search engine optimization (SEO)!
  - Paid inclusion

WE Process and Tasks

Deployment: Web site operation

- **Usage Analysis**: Log-File Analysis etc. => Google Analytics
- **Access statistics** (IVW, ÖWA,...) => Placing advertisements
- Backups!

#	IP Address	Userid	Time	Method/ URL/ Protocol	Status	Size	Referrer	Agent
1	123.456.78.9	-	[25/Apr/1998:03:04:41 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.04 (Win95, I)
2	123.456.78.9	-	[25/Apr/1998:03:05:34 -0500]	"GET B.html HTTP/1.0"	200	2050	A.html	Mozilla/3.04 (Win95, I)
3	123.456.78.9	-	[25/Apr/1998:03:05:39 -0500]	"GET L.html HTTP/1.0"	200	4130	-	Mozilla/3.04 (Win95, I)
4	123.456.78.9	-	[25/Apr/1998:03:06:02 -0500]	"GET F.html HTTP/1.0"	200	5096	B.html	Mozilla/3.04 (Win95, I)
5	123.456.78.9	-	[25/Apr/1998:03:06:58 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
6	123.456.78.9	-	[25/Apr/1998:03:07:42 -0500]	"GET B.html HTTP/1.0"	200	2050	A.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
7	123.456.78.9	-	[25/Apr/1998:03:07:55 -0500]	"GET R.html HTTP/1.0"	200	8140	L.html	Mozilla/3.04 (Win95, I)
8	123.456.78.9	-	[25/Apr/1998:03:09:50 -0500]	"GET C.html HTTP/1.0"	200	1820	A.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
9	123.456.78.9	-	[25/Apr/1998:03:10:02 -0500]	"GET O.html HTTP/1.0"	200	2270	F.html	Mozilla/3.04 (Win95, I)
10	123.456.78.9	-	[25/Apr/1998:03:10:45 -0500]	"GET J.html HTTP/1.0"	200	9430	C.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
11	123.456.78.9	-	[25/Apr/1998:03:12:23 -0500]	"GET G.html HTTP/1.0"	200	7220	B.html	Mozilla/3.04 (Win95, I)
12	209.456.78.2	-	[25/Apr/1998:05:05:22 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.04 (Win95, I)
13	209.456.78.3	-	[25/Apr/1998:05:06:03 -0500]	"GET D.html HTTP/1.0"	200	1680	A.html	Mozilla/3.04 (Win95, I)

WE Process and Tasks

Maintenance

- **Content maintenance!**
- **Reliability** and **performance**
  - Network, hardware, database tuning, etc.
- **New requirements**
  - Functional requirements
  - Presentational requirements
- **Technological Evolution**
  - New tools: browsers, operation systems, Web servers, class libraries, etc.
  - New Technologies: Shockwave, Ajax, etc.
- **New releases**
  - Configuration Management!
  - 24x7 Operation!

WE Process and Tasks

Quality