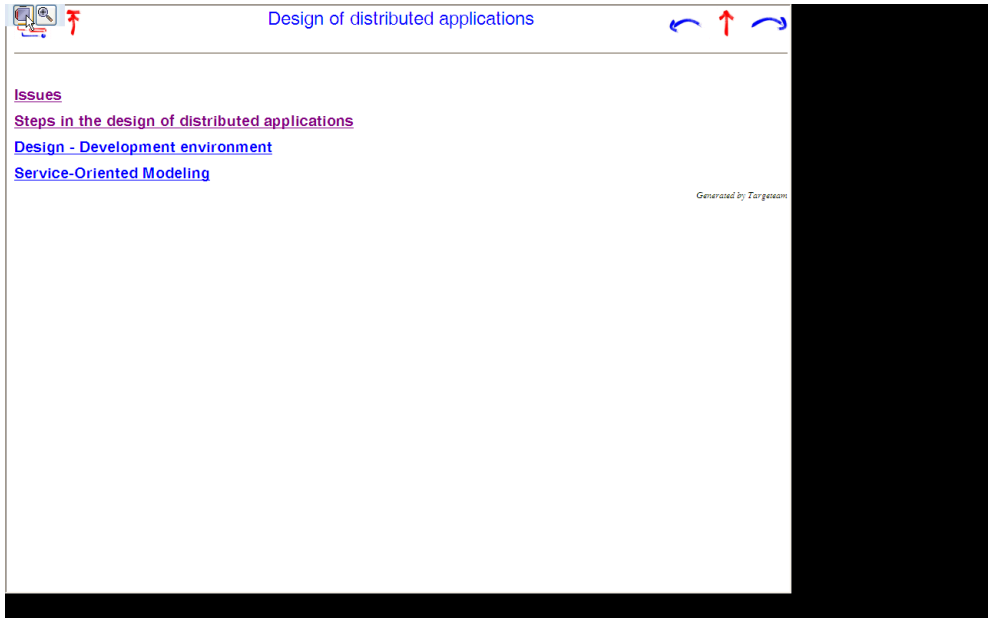
Script **generated by TTT**

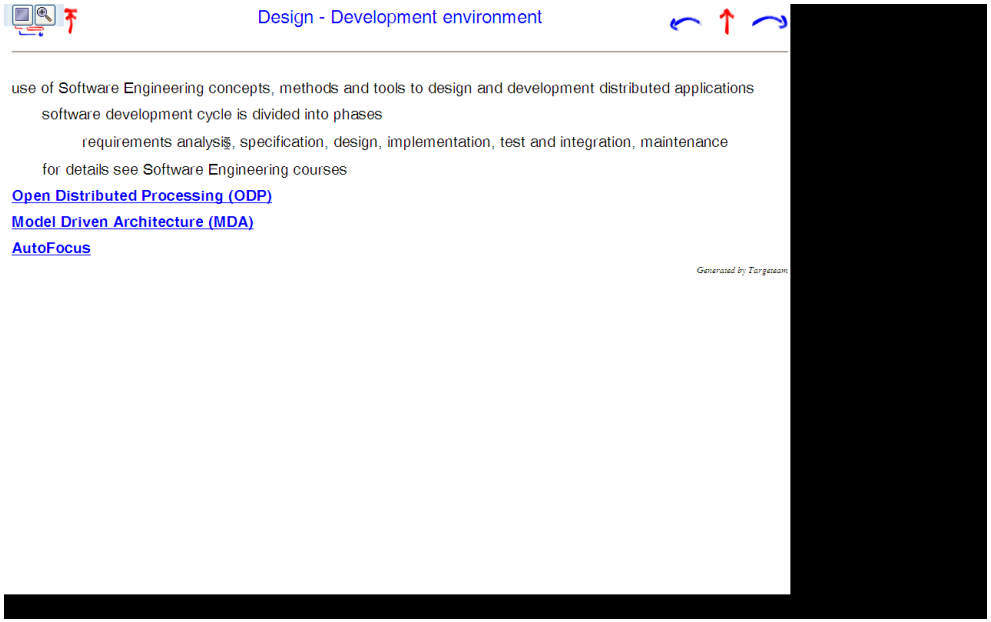
Title: Distributed_Applications (09.07.2013)

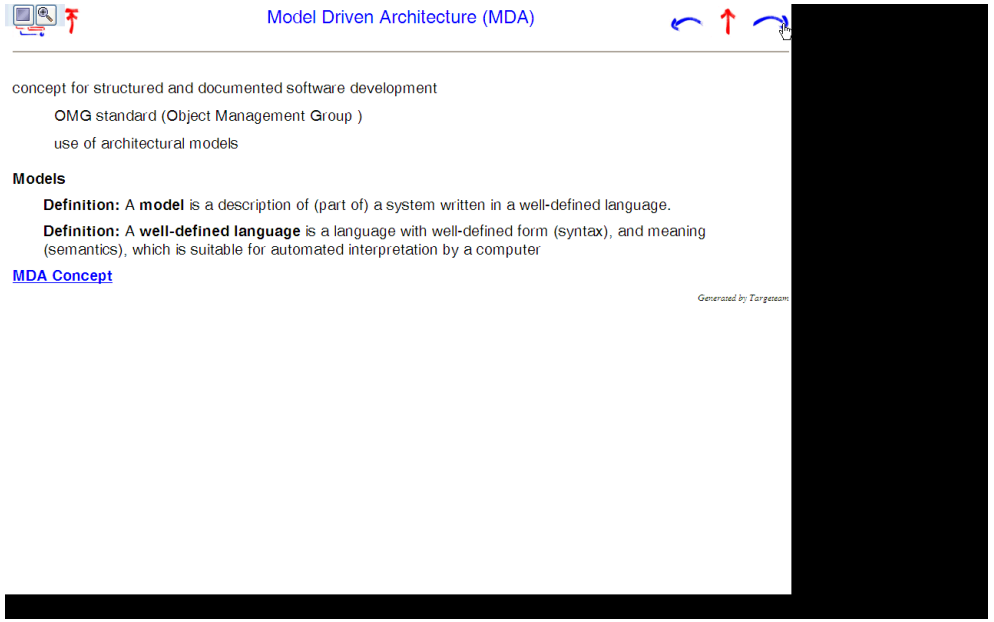
Date: Tue Jul 09 14:31:28 CEST 2013

Duration: 90:53 min

Pages: 19





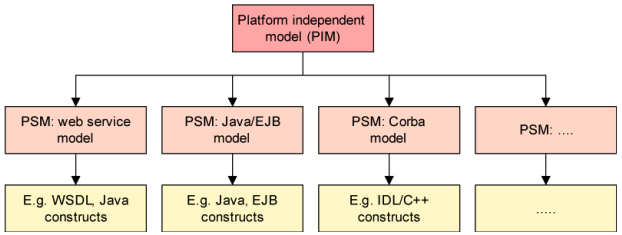


3. Step: code generation

generation of specific technological constructs, e.g. Java packages

implementation of system functionality

use of tools for automatic code generation



MDA Concept

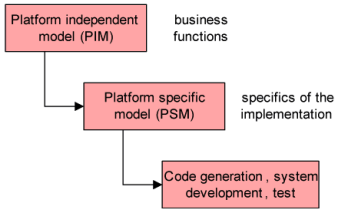
consists of 3 steps

development of platform independent models (PIMs)

mapping to platform dependent models (PSMs)

implementation, integration and test

transformation between models (PIM → PSM, PSM → code)



1. Step: development of PIM

2. Step: mapping to PSM

3. Step: code generation

AutoFocus

AutoFocus is a platform to specify distributed systems

developed by the group of Prof. Broy, TU München

based on formal methods of systems engineering

integrates hierarchical description techniques

allows distributed and platform independent development

project advanced to AutoFocus 2 supporting the following functionality

requirement analysis tool (AutoRAID), such as use-cases and scenarios, business and application requirements

Design modelling views and editors, such as system structure diagram, state transition diagram, message sequence charts

interactive simulation environment, code generation, consistency maintenance support,

Service-Oriented Modeling

Ideas and proposals emerged to transfer the service approach to the design and modeling of software systems.

Definition: Service-oriented modeling (SOM) is the discipline of modeling business and systems, for the purpose of designing and specifying service-oriented business systems within SOA.

create models that provide a comprehensive view for the analysis, design, and architecture of all software components in an organization.

envision the coexistence of services in an interoperable computing environment.

Definition: The **service-oriented modeling framework** (SOMF) is a service-oriented development life cycle methodology that provides practices, disciplines and a universal language to provide tactical and strategic solutions to enterprise problems

Service Evolution

Life Cycle Structure

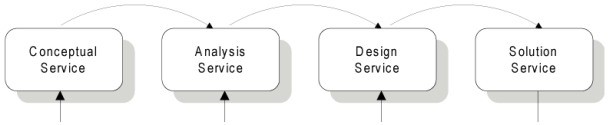
Life Cycle Modeling

SOM Framework

Other SOA Design Methodologies

Service Evolution

SOM advocates the transformation of a service through 4 states.



1. conceptual service: in its inception, a service appears merely as an idea or concept.
2. analysis service: it becomes a unit of analysis.
3. design service: it evolves into a design entity.
4. solution service: it ends in a physical solution that is ready to be deployed in the production environment.

Life Cycle Structure

identifies the elements for service development and operations. It consists of 4 major components.

timeline : defines the life span of a service.

events : 2 types of events during the service life span.

predicted and scheduled events, e.g. milestone, planning stage or deployment stage.

unexpected events, e.g. stock market crash, trading volume exceeds capacity of trading service.

events have beginnings and may last for a while.

seasons : services live through 2 major life cycle seasons.

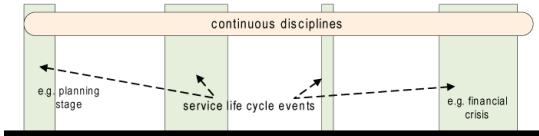
design-time season: services are conceptualized, analyzed, designed, constructed and tested.

run-time season: services are managed, monitored, and controlled to ensure proper performance.

disciplines : identify modeling and nonmodeling best practices and standards to be pursued throughout the service life cycle.

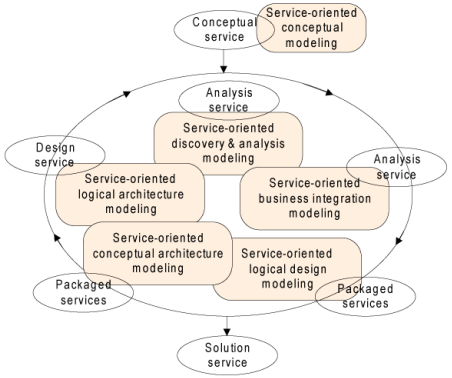
season disciplines: e.g. service-oriented conceptualization, business integration or construction.

continuous disciplines: e.g. service portfolio management, service governance.



Life Cycle Modeling

The following core processes can be identified in which business and IT personnel must be engaged to produce design and solution artifacts.



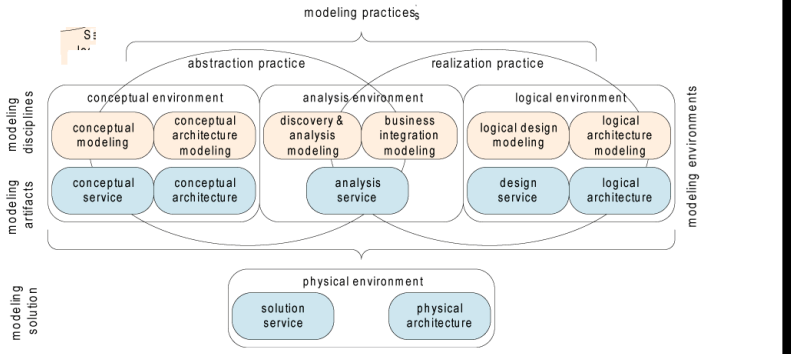
Conceptual modeling : identify driving concepts behind future solution services.

Discovery & analysis modeling : discover and analyze services for granularity, reusability, interoperability, loose-coupling, and identify consolidation opportunities for the existing software assets.

Business integration modeling : identify service integration and alignment opportunities with business

SOM Framework

Modeling components and disciplines are integrated into a SOM framework.



Other SOA Design Methodologies

A brief overview of some other SOA design methodologies

- Creating Service-Oriented Architectures (CSOA) by Barry & Associates
  - focus is on technical aspects
  - consist of the 5 phases
    - experiment with Web Services
    - adapt existing systems to use Web Services
    - remove intersystem dependencies
    - establish internal SOA
    - incorporate external services
- Service-Oriented Transformation of Legacy Systems (SOTLS) by Nadhan
  - targets the stepwise evolution of existing application systems towards service-oriented architectures
  - focus is on technical aspects
- Service-Oriented Design and Development (SOAD) by Papazoglou
  - incorporates the perspectives of the service provider as well as the service consumer
  - consist of the phases
    - planning, analysis, service design, service construction, service test, service deployment/execution and service management/monitoring

Design of distributed applications

Introduction

When a group of programmers has the task to build a distributed application, in addition to distributed code management there is also the need for distributed file services.

Introduction

When a group of programmers has the task to build a distributed application, in addition to distributed code management there is also the need for distributed file services.

Definitions

Motivation for replicated files

Two consistency types

Replica placement

Generated by Targeteam

Introduction

When a group of programmers has the task to build a distributed application, in addition to distributed code management there is also the need for distributed file services.

Definitions

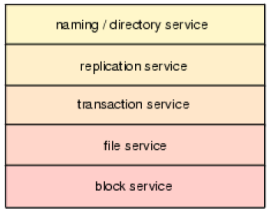
Motivation for replicated files

Two consistency types

Replica placement

Layers of a distributed file service

The functions of a distributed file service are usually arranged in a hierarchical way.



Layer semantics

Update of replicated files

Basically, there are two types of approaches for multiple update control: the optimistic and the pessimistic approach.

Optimistic concurrency control

Pessimistic concurrency control

Voting schemes

Voting schemes

Voting schemes provide pessimistic concurrency control.

Introduction

Voting schemes are algorithms for maintaining mutual consistency of replicates even in situations of computer crashes and network partitionings.

Let us assume, there exist REP replicas of file d.

Let sg(r) be the weight of the vote of computer r; K be the set of all computers considered.

Let the sum of all weights be SUM = $\sum_{r \in K} sg(r)$.

Definitions

Multiple-reader-single-writer strategy

Voting scheme variants