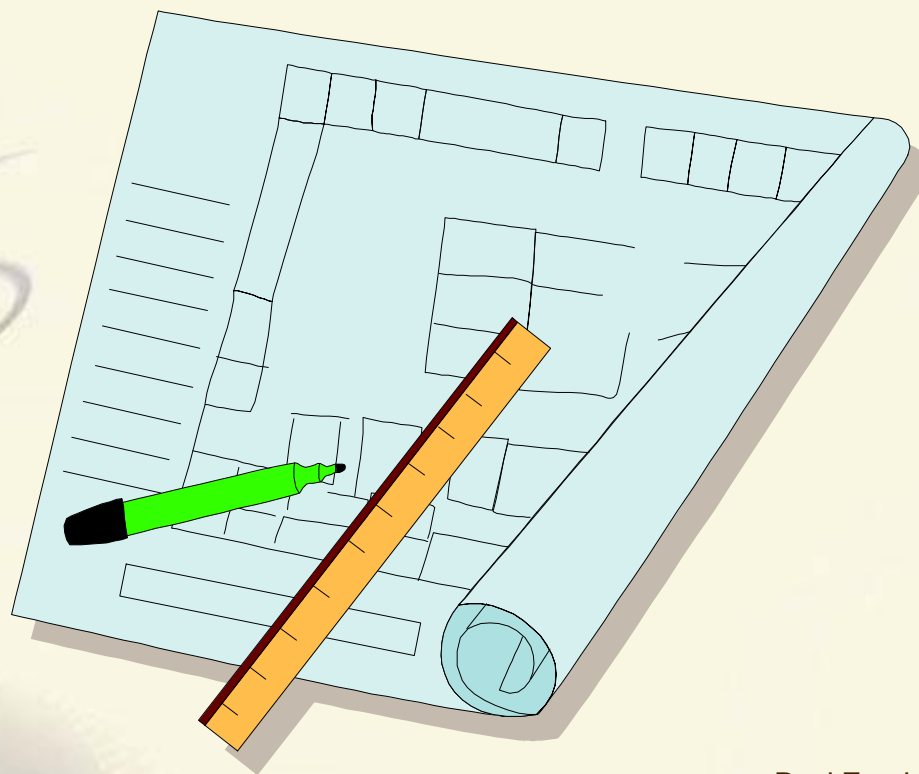


SIoux



Paul Zenden  
Sioux Technische Software Ontwikkeling  
Science Park Eindhoven 5709  
5692 EP Son  
Tel: +31 (0) 40 26 77 100  
E-mail: paul.zenden@sioux.nl





**Experiences with**

***Enterprise Architect***

***UML Modeling & Design tool***

***..and more***

Paul Zenden  
Sioux Technische Software Ontwikkeling  
Science Park Eindhoven 5709  
5692 EP Son  
Tel: +31 (0) 40 26 77 100  
E-mail: paul.zenden@sioux.nl

# Agenda

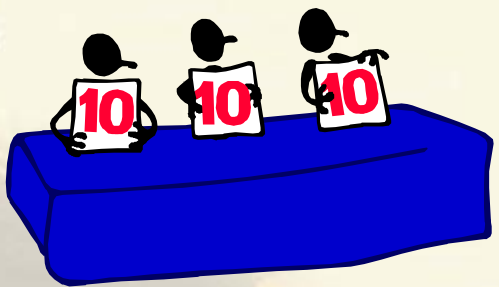
*Overview*



*Final Word(s)*



*Some examples*



*Best bargain?*



*Architect(-ing) tool?*



# Agenda

Overview



*Final Word(s)*



**ENTERPRISE  
ARCHITECT**

*Some examples*

*Best bargain?*

*Architect(-ing) tool?*



The Sioux logo consists of the word "SIOUX" in white, uppercase, sans-serif font, set against a black rectangular background.

# Overview (1)

---



<http://www.sparxsystems.com.au/ea.htm>

## ■ One-liner:

- A comprehensive UML analysis and design tool, covering software development from requirements gathering, through the analysis stages, design models, testing and maintenance, designed to help you build robust and maintainable software. It features flexible and high quality documentation output.

## ■ User base:

- 29,000 licenses world-wide (US, Canada, many Western European countries, and many more)
- Used for the development of various kinds of software systems for a wide range of industries:
  - e.g. banking, web development, engineering, finance, medicine, research, academia, transport, retail, utilities (gas, electricity etc.), electrical engineering
- Used effectively for UML and business architecture training purposes:
  - training companies and universities around the world.

## Overview (2)

---

- **UML Modeling**
  - UML 2.0 diagrams and elements
  - custom, analysis diagram
- **Requirements management:**
  - Business & system requirements
  - Link requirements to realization elements
- **EA Project & Change management:**
  - Can be used within many project life cycle approaches
  - Reusability of project templates
  - Maintain risks, issues, changes & defects
  - Maintain resources, roles & tasks
  - Estimation & Metrics
- **Test management:**
  - Define tests: unit, system, integration, acceptance



# Overview (3)

---



## ■ Profiles & Patterns:

### ■ UML Profiles

- E.g. Eriksson-Penker Business Extensions, Web Modeling, XSD Schema
- Custom defined

### ■ Patterns:

- E.g. GoF
- Custom defined

## ■ Software code engineering:

### ■ Model Driven Generator (MDG) Technologies:

- Bundle specific technology related resources (patterns, profiles, code definitions)
- Ejb, Testing, Visual Studio.Net link (Add-on)

### ■ Generating & reverse engineering of:

- Code: C#, C++, Delphi, Java, VB, VB.Net, PHP
- Database DDL: DB2, MS Access, MS Sql Server, MySQL, Oracle 9i, other ODBC sources

# Overview (4)

---

- **Team development & models sharing**
  - Stand-alone, replicated, central
  - User based security
  - Integration other tools: XMI 1.1
  - Software API
- **Project documentation:**
  - RTF based
  - HTML
  - VB for Applications
  - Filtering what to output
- **Support:**
  - Active users forum
  - Update 2/month: New features, bug fixes
  - 1 year free update
  - Context sensitive extensive help
  - Plug-in: Zicom help system: Visual UML Dictionary







## Enterprise Architect Features

### Comprehensive support for UML 2.0

- [Use Case Model](#)
- [Business Process Model](#)
- [Dynamic Models](#)
- [Logical Model](#)
- [Component and Deployment diagrams](#)
- [Custom Extensions](#)
- Collaborations

### Comprehensive and flexible documentation

- [Output in rich text format](#)
- Flexible output options
- Save report templates for later re-use
- RTF Bookmarks for improved document merging/linking in Word

### Forward and Reverse Code Engineering for

- C++
- Java
- Visual Basic
- Delphi
- C#
- VB.Net
- PHP

### Easy to use, explorer like interface

#### Support for testing

- Unit tests
- Integration tests
- System tests
- Acceptance tests
- Scenarios

#### Support for maintenance

- Change control details
- Maintenance and fault recording

#### Additional support for requirements gathering

#### Project Resource Support

#### Traceability reporting

#### Multi-user capable. Fast!

#### Excellent search facilities

#### Flexible appearance, colour and display options

#### Project "guesstimate" metrics based on hours per adjusted Use Case point

#### Support for Glossary

#### XML import and export

#### Spell Checker

The screenshot shows the Enterprise Architect v4.1 website. The main content area features a large heading: "ENTERPRISE ARCHITECT v4.1 The Feature-Rich UML Modeling Tool for the Entire Development Team". Below this, there is a quote: "Enterprise Architect is a flexible, complete and powerful UML modeling tool for the Windows platform, providing the competitive edge for system development, project management and business analysis; an object oriented CASE tool for the full development life-cycle - at a sensible price."

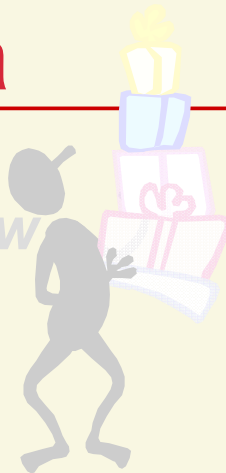
Other visible text on the page includes:
 

- "Space Systems are proud to announce the release of Enterprise Architect 4.1 and 7000 plus the Visual Studio .NET"
- "User Testimonials: 'I had using Rational know the other EA for the first time after using EA - and it was amazingly easy to compare. You've got a very capable product at UML the cost!'"
- "Enterprise Architect v4.1 builds on the exceptional success of previous releases with comprehensive support for the new UML 2.0 standard as defined by the OMG (www.omg.org). With EA 4.1, installed, you have the power and responsiveness of all UML 2.0 diagrams at their fingertips."
- "In addition to standard UML elements, Enterprise Architect supports extensions for user interface design, requirements gathering and process modeling. Provides full traceability from user Requirements to Use Cases, Logical models, User Interfaces, Component models and Deployment models."
- "Support for all UML 2.0 diagrams and associated diagram elements, including: Use Case Diagrams."



# Agenda

Overview

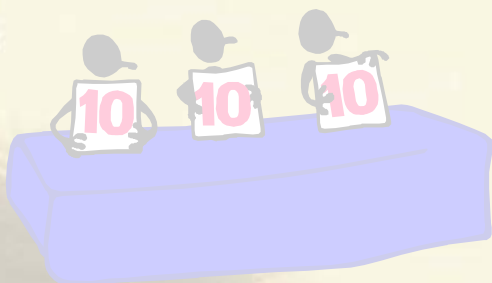


**Some examples**

*Final Word(s)*



**ENTERPRISE  
ARCHITECT**



*Best bargain?*



*Architect(-ing) tool?*



EA Desktop interface showing a Statechart Diagram for "State -- History".

**Statechart Diagram:** "State -- History" created: 9/02/2004 modified: 28/04/2004 73% 799 x 1067

The diagram shows a state machine with the following components and transitions:

- Initial State:** "OS running" (part of "Install Software" state).
- Transitions:**
  - "OS running" to "Install" (labeled "Get(Start)").
  - "Install" to "Disk error" (labeled "Get(Start)").
  - "Install" to "Show log" (labeled "Get(Start)").
  - "Disk error" to "OS running" (labeled "Get(Start)").
  - "Show log" to "OS running" (labeled "Get(Start)").
  - "OS running" to "OS running" (labeled "Get(Start)").
- States:**
  - Install Software:** Contains "OS running" and "Install".
  - Disk error:** Contains "OS Action: Show error handling" and "OS Action: Ask alternative".
  - Show log:** Contains "OS Action: Show log handling" and "OS Action: Ask alternative".

**Project View:** Shows a tree structure with "State -- History" as the root, containing "Activity", "Disk error", "Install Software", "Memory Low", "Region", "State1", and "State2".

**Properties:** Shows "General Settings" with fields for Name, Scope, Type, Stereotype, Akes, Complexity, Version, Phase, Language, and Filename.

**System Table:**

Priority	Task	Type	Status	Owner	Description

Model Tasks: Model Issues: Model GI





State: Memory Low

General Require Constraints Link Scenario Tags Files

Name:

Stewtype:   Abstract

Author:  Status: Proposed

Scope: Public Complexity: Easy

Alias:  Language: C++

Keywords:

Phase: 1.0 Version: 1.0

Note:

Apply OK Cancel Help

**Object Properties**

State: Memory Low

General Require Constraints Link Scenario Tags Files

Constraint:  Type: Post-condition

Status: Approved

Defined Constraints

Constraint	Type	Status
User Registered	Pre-condition	Approved

Apply OK Cancel Help

**Object Constraints**

Use Case: Use Case1

General Require Constraints Link Scenario Tags Files

Requirement:  Type: Functional

Status: Proposed Difficulty: Medium Priority: Medium Last Update: 25/04/2004

**Object Requirements**

Defined

Requirement	Type	External
1. Log in to System	Functional	
1.1. Log in to Web Site	Functional	
2. Go to Registration Screen for new U...	Functional	
3. Encrypt User details over HTTPS	Functional	
4. Log Off System	Functional	

Apply OK Cancel Help





## Rich Text Format Report

Document Item:

Document Type:

Output Filename:

Template Name:  Style:

Heading:

Heading Style:

Initial Heading Level Indent:

### Introduction

The Use Case Model describes the proposed functionality of the new system. A Use Case represents a discrete unit of interaction between a user (human or machine) and the system. A Use Case is a single unit of meaningful work; for example creating a train, modifying a train and creating orders are all Use Cases. Each Use Case has a description which describes the functionality that will be built in the proposed system. A Use Case may 'include' another Use Case's functionality or 'extend' another Use Case with its own behaviour. Use Cases are typically related to 'actors'. An actor is a human or machine entity that interacts with the system to perform meaningful work.

**Filter**

Only include objects:

Where Package Phase:

With element status:

<p><b>Template</b></p> <input type="button" value="New"/> <input type="button" value="Save"/> <input type="button" value="Delete"/> <input type="button" value="Load"/>	<p><b>Language</b></p> <input type="button" value="Adjust"/> <input type="button" value="Page Setup"/>	<p><b>Document</b></p> <input type="button" value="Create"/> <input type="button" value="Close"/> <input type="button" value="View"/> <input type="button" value="Help"/>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### For each Object Include

- Object Details
- Requirements
- Associations
- Scenarios
- Child Diagrams
- Attributes
- Methods
- Constraints
- Resources
- Tagged Values
- Tags
- Aliases
- Element Changes
- Element Defects
- Element Issues
- Element To Do
- External Hyperlinks
- Internal Hyperlinks
- Test Cases
- Assigned Tasks
- Detailed Connectors
- Feature Tags
- Simple Name Only
- Extended Properties

### Diagram format

- GIF
- EMF
- PNG
- Bitmap
- WMF
- JPG

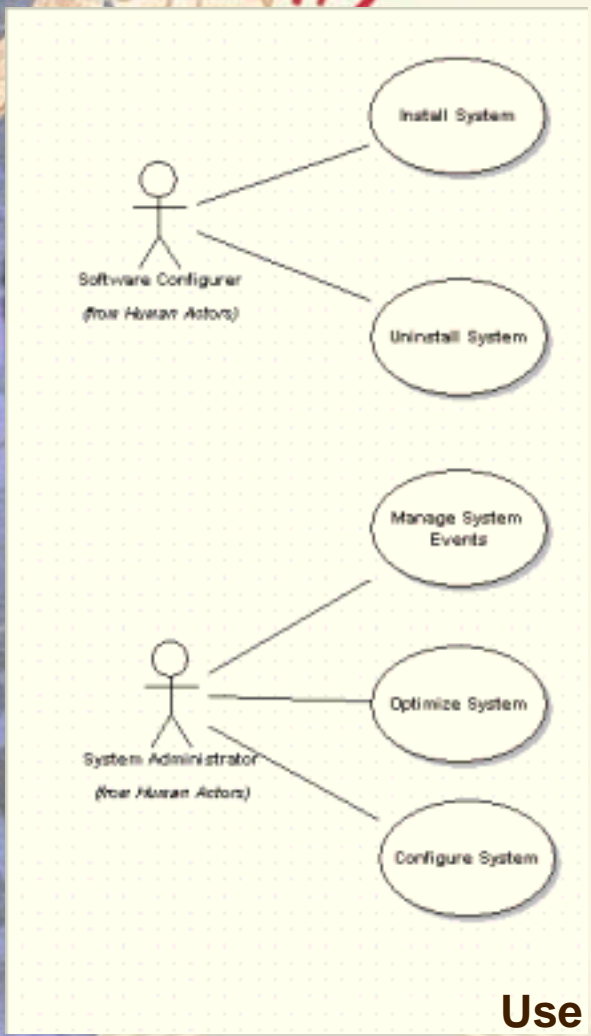
### Exclude details for

- Action
- Activity
- ActivityFinal
- ActivityInitial
- ActivityPartition
- ActivityRegion
- Actor
- Artifact
- Boundary
- Change
- Choice
- Class
- Collaboration
- Collaboration Msgs
- CollaborationOccurrence
- Component
- Constraint
- Decision

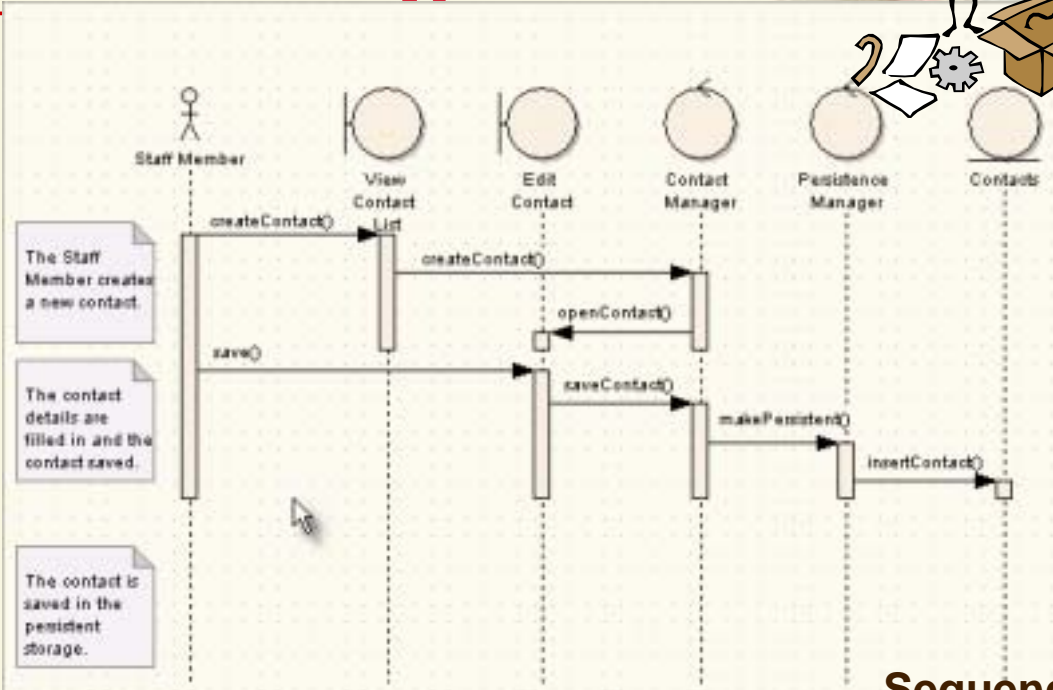
### Options

- Process all children
- Show Diagrams
- New page per package
- Document all elements
- Document Packages
- Hide 'note-less' elements
- Embed Diagrams in Document
- Skip root package
- Document Linked Elements

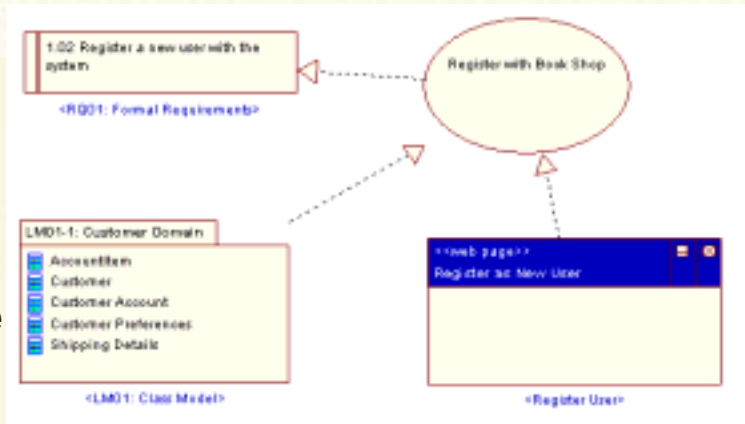
# Screens: Diagrams -1

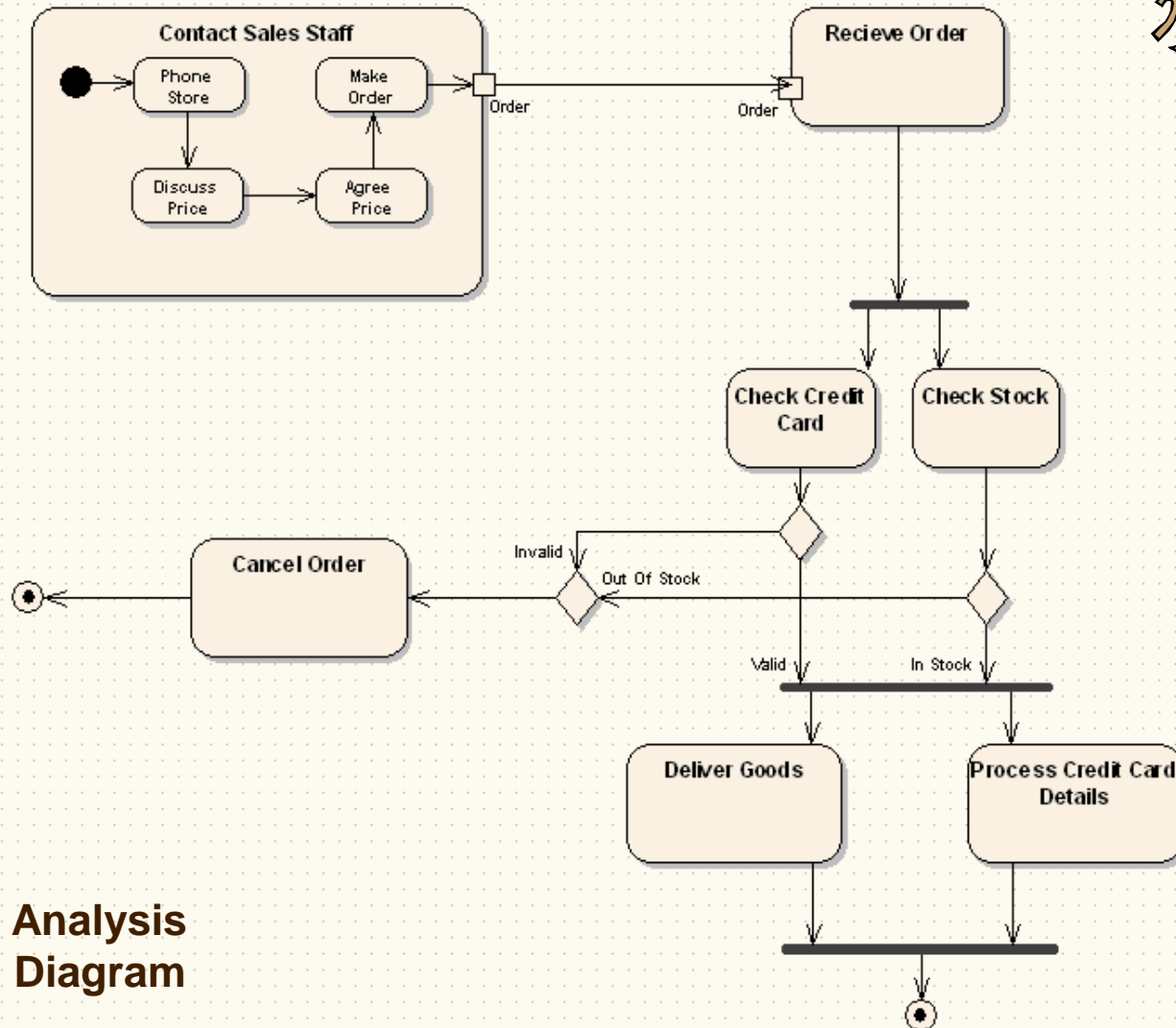


**Use Case Diagram**

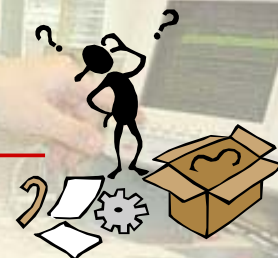


**Sequence Diagram**

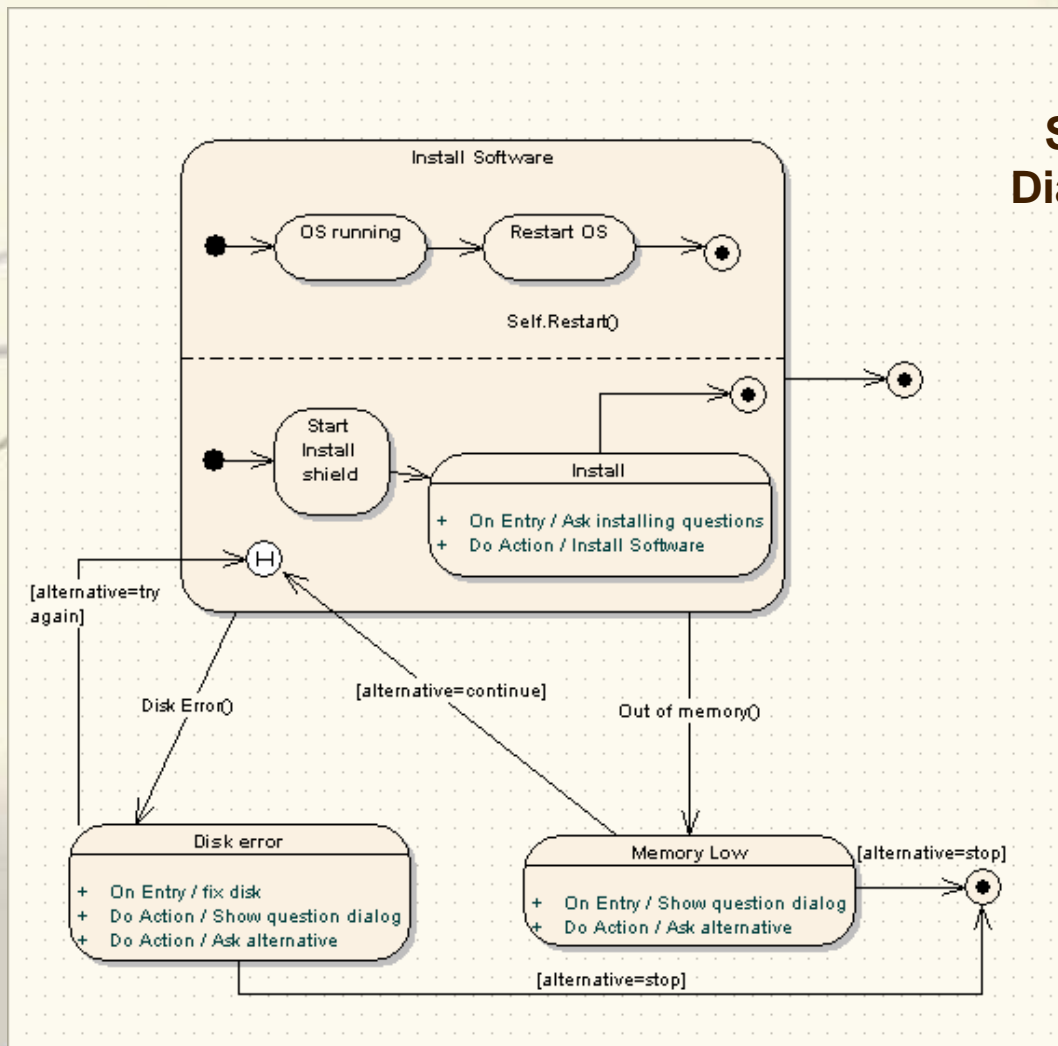




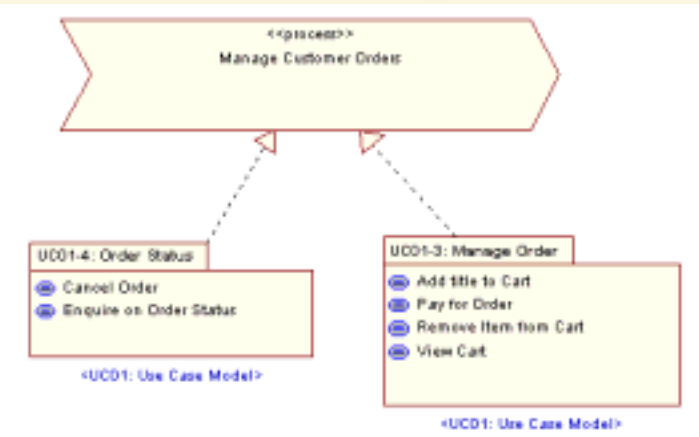
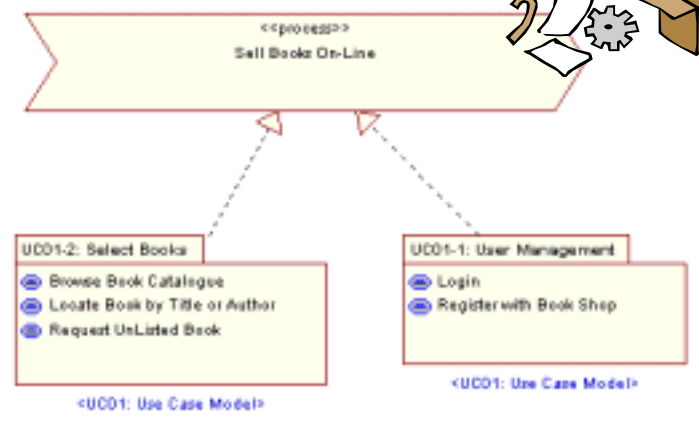
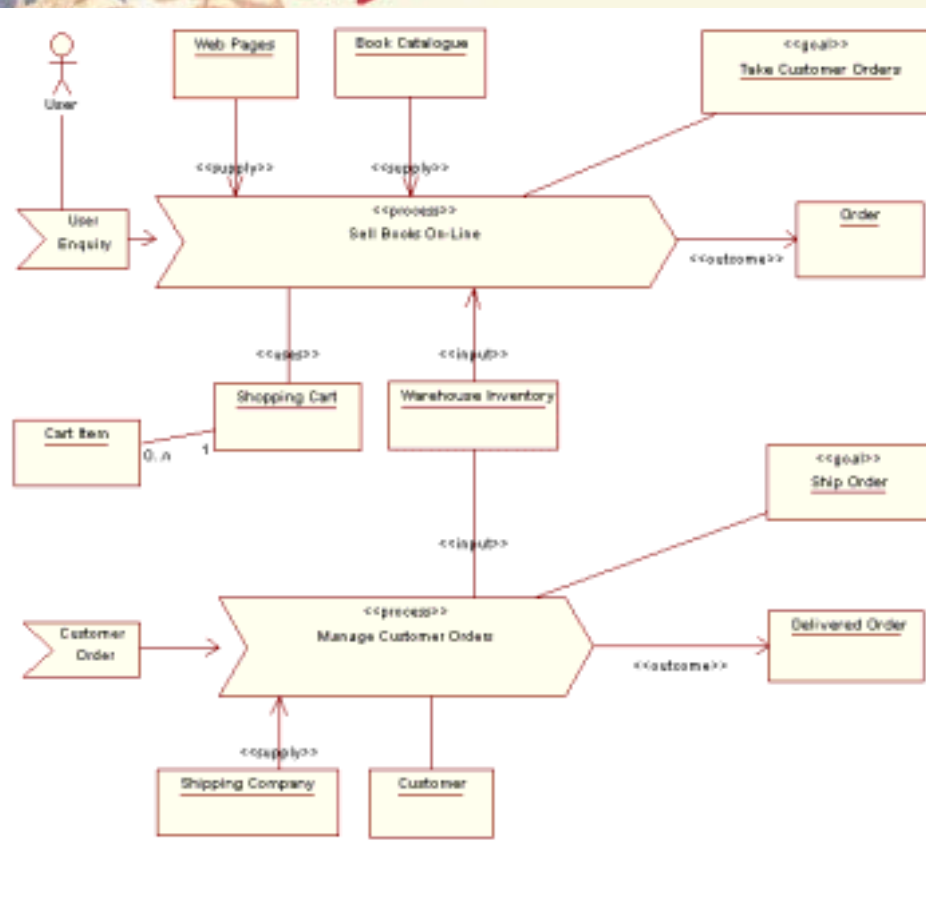
**Analysis Diagram**



## State Diagram







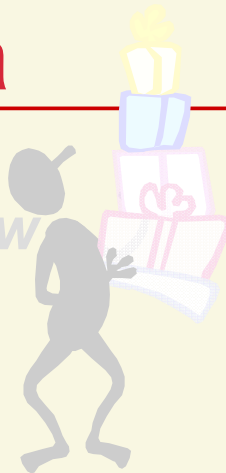
This diagram illustrates the traceability from the Business Process "Manage Customer Orders" to the use cases in the "Order Status" and "Manage Order" packages. A realization link is used to model the implementation relationship - providing the business justification for having the related use cases in the model.

**SIoux**



# Agenda

Overview

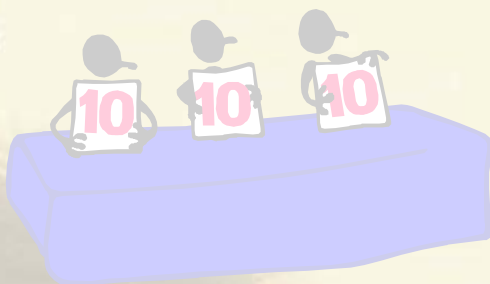


Some examples

Final Word(s)



**ENTERPRISE  
ARCHITECT**



Best bargain?



**Architect(-ing) tool?**

# Architect tool?



## ■ A case tool supports the architect in:

### ■ Enforcing vision:

- Concepts, Components, Interfaces
- Profiles & Patterns
- Enforce OO rules; Guiding analysis & design

### ■ Communication:

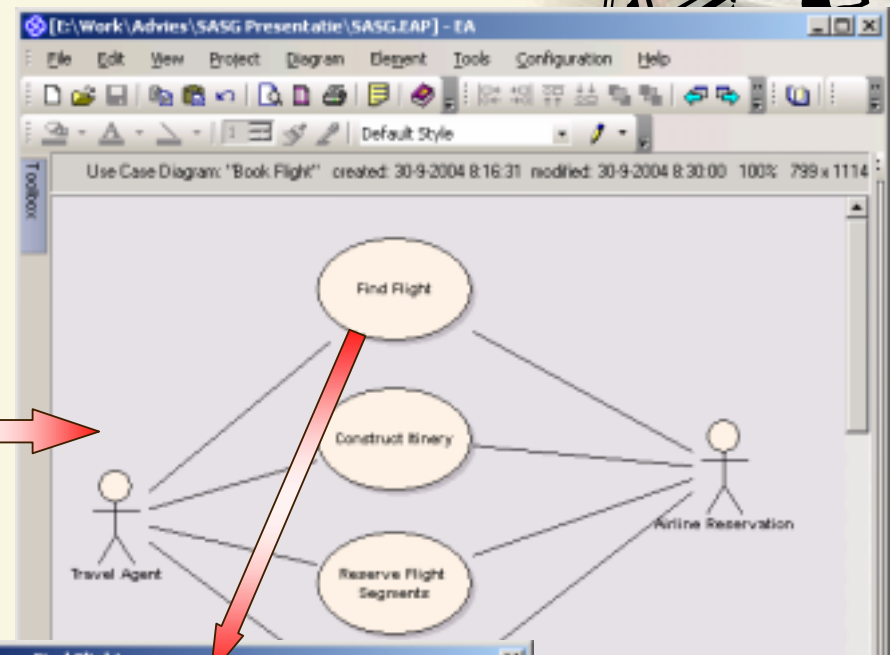
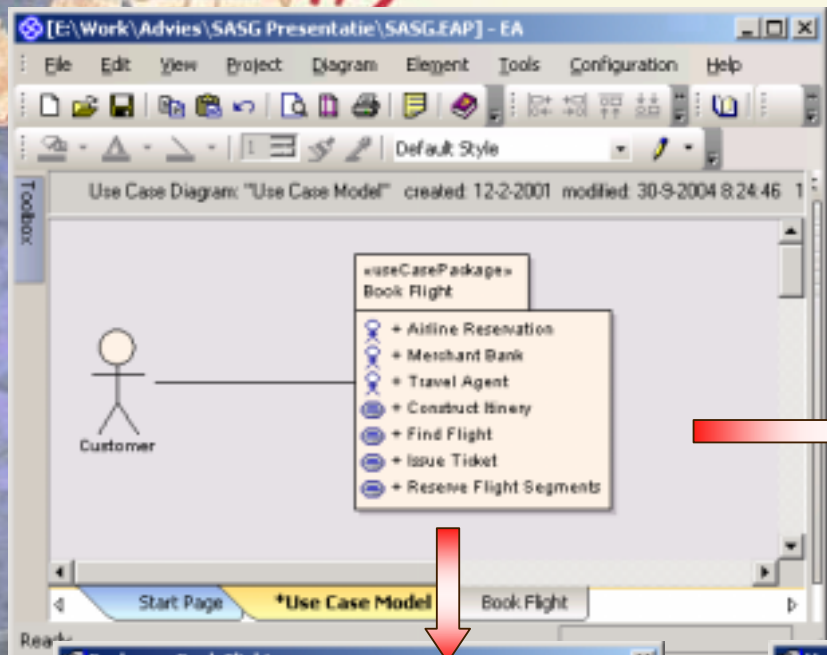
- One design tool for all development platforms
- One drawing technique only
- One modeling method only
- Share knowledge

### ■ Team-based development

- Single repository
- User security
- Documentation

### ■ Many other things...





**Package : Book Flight**

General | Require | Constraints | Link | Scenario | Tags | Files

Scenario: Main Success Scenario Type: Basic Path

1. This use case begins when a customer calls and requests a flight.
2. The customer describes her flight needs by specifying her origination, destination, travel dates, and preferred departure times.
3. The system looks up all flights that match the customer's travel preferences and presents the travel options to the customer.
4. The customer selects a flight.
5. The system builds a flight itinerary for the customer.
6. The system reserves the flight for the customer.
7. The customer provides credit card number and charges the price of

Scenarios: [New] [Save] [Delete]

Name	Type
Main Success Scenario	Basic Path

**UseCase : Find Flight**

General | Require | Constraints | Link | Scenario | Tags | Files

Scenario: Main Success Scenario Type: Basic Path

1. The use case begins when a customer contacts the travel agency and requests a flight.
2. The travel agent captures the customer's trip origin and destination.
3. The travel agent looks up the airport codes for the origin and destination.
4. The travel agent captures the preferred departure times for the customer.
5. The travel agent captures the customer's preferred class of service.
6. The travel agent confirms that the customer's preferences are

Scenarios: [New] [Save] [Delete]

Name	Type
Main Success Scenario	Basic Path

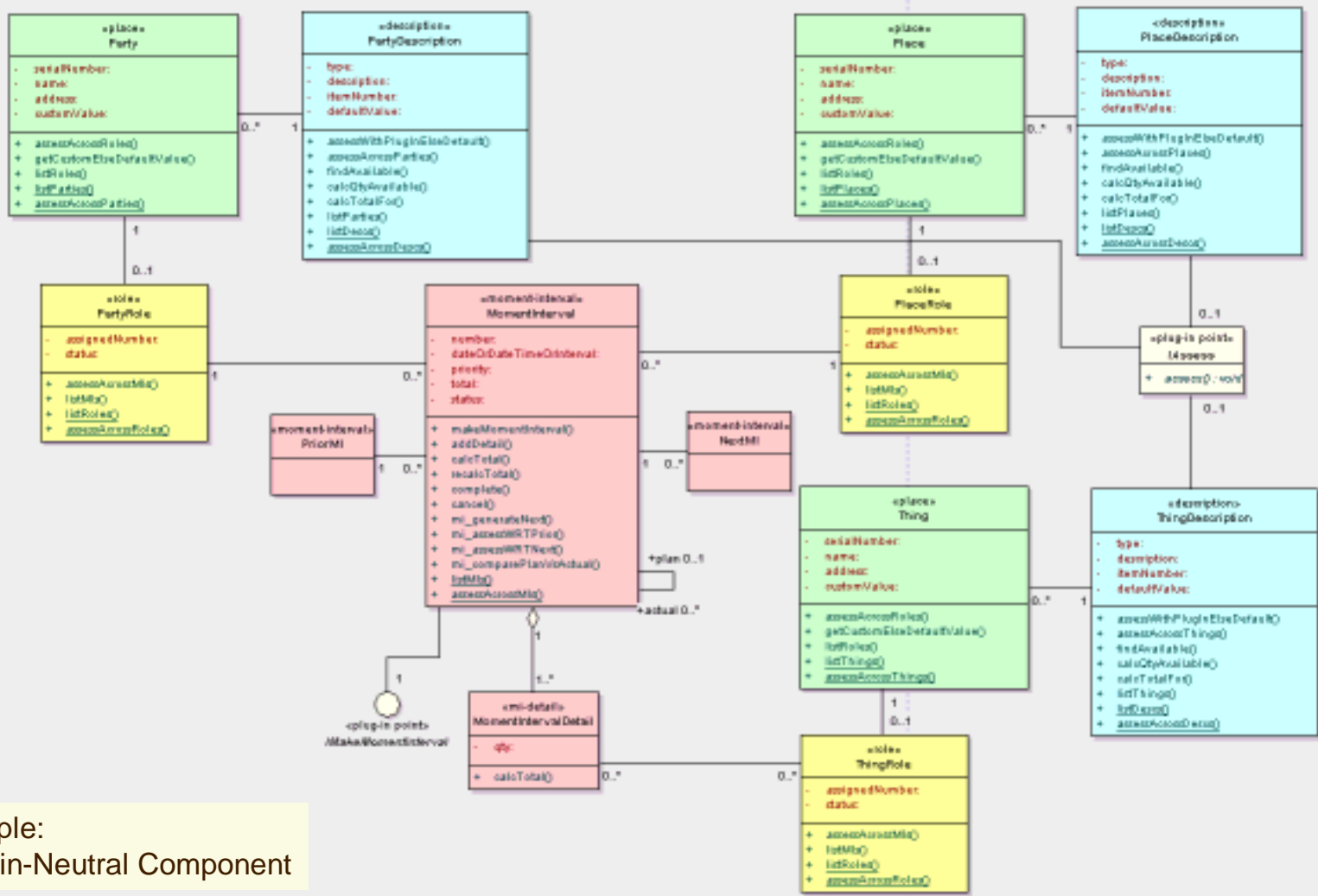




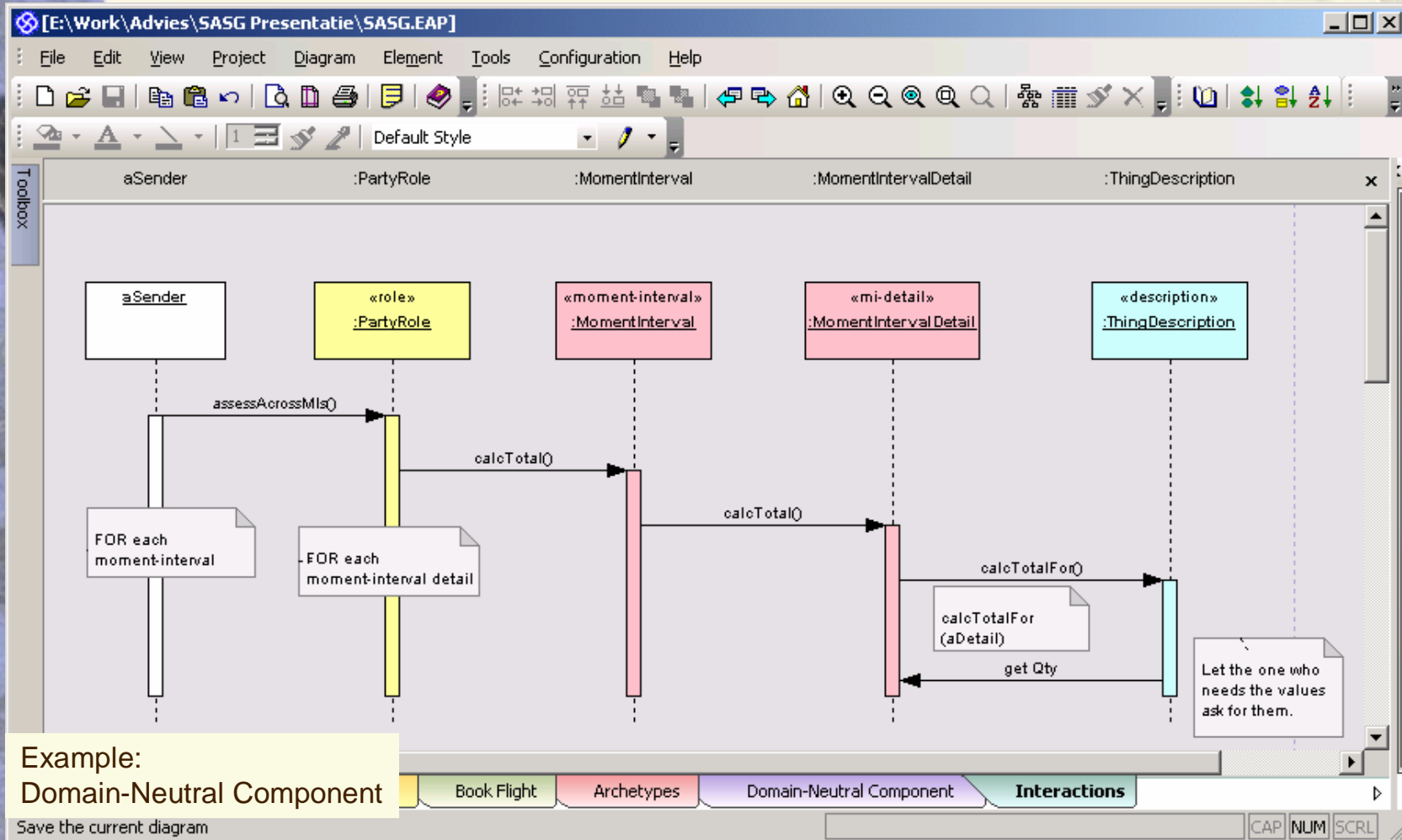
# Class Diagram



Class Diagram: "Domain-Neutral Component" created: 30-9-2004 8:58:41 modified: 30-9-2004 10:07:49 100% 799 x 1114



Example:  
Domain-Neutral Component



Example:  
Domain-Neutral Component

Book Flight

Archetypes

Domain-Neutral Component

Interactions

Save the current diagram

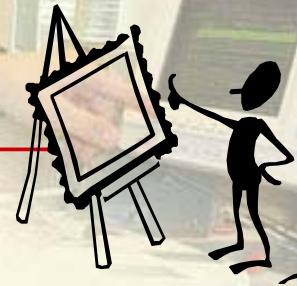
30/09/2005

Copyright © Sioux Technische Software Ontwikkeling 2004

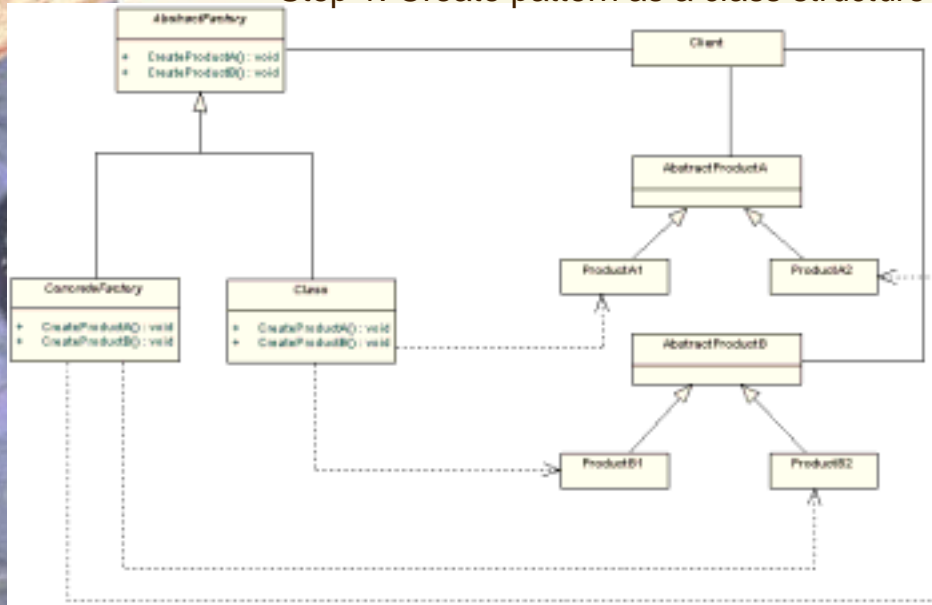
CAP NUM SCRL



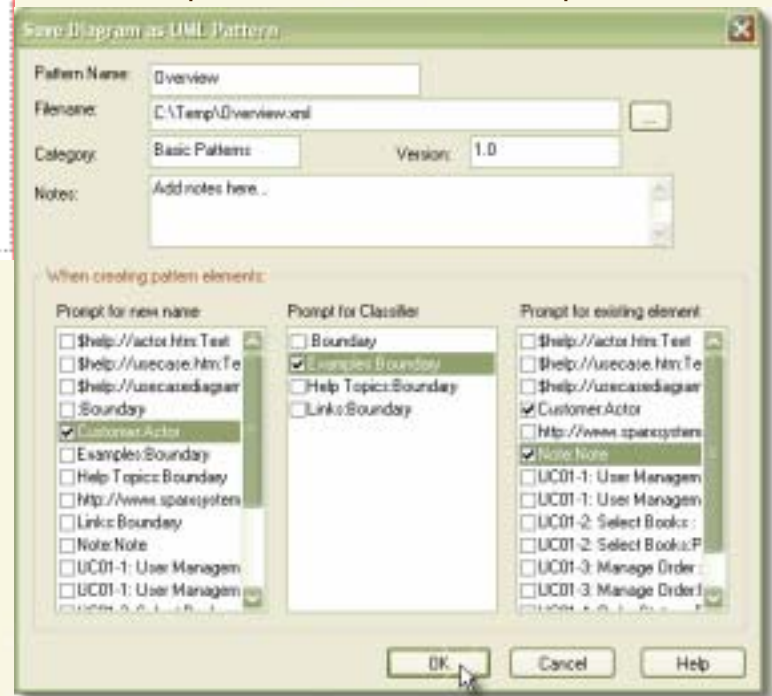
# UML Patterns: Define



Step 1: Create pattern as a class structure



Step 2: Save structure as a pattern

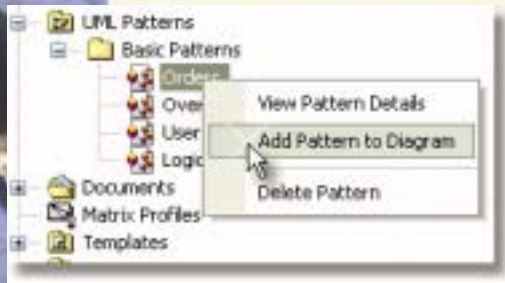


# UML Patterns: Use



Step 1: Import pattern

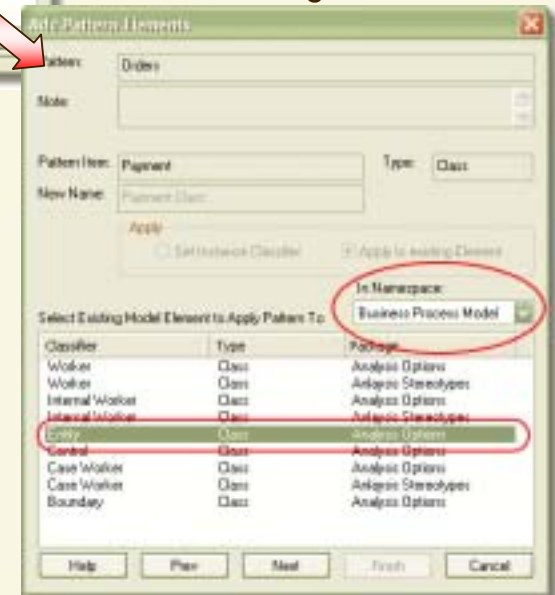
Step 2: Use pattern



Step 3: Fill in variable part of all elements participating in patterns;

Step 3b: Forces to select from existing elements

Step 3a: Forces to give a new name





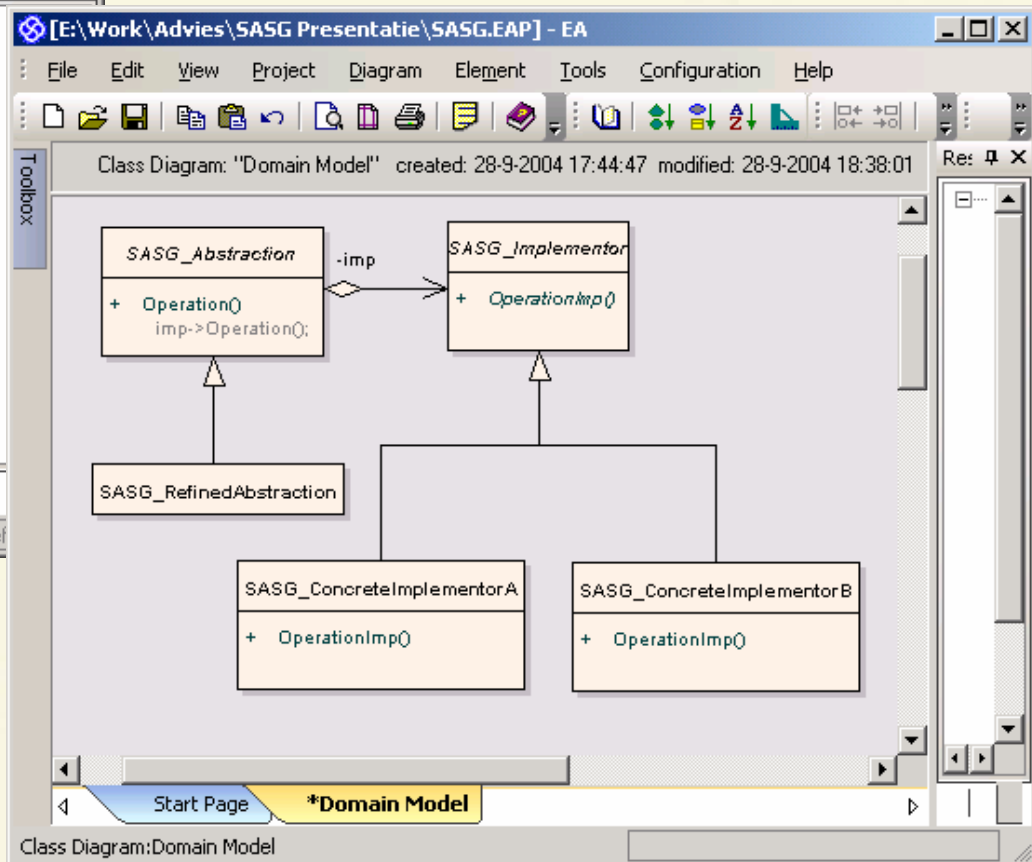
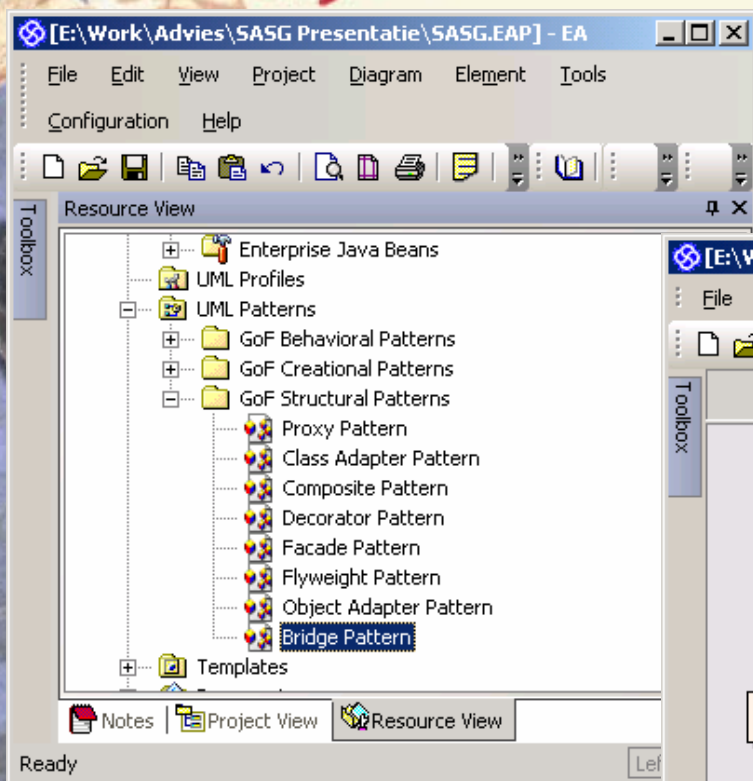


# UML Patterns: Example

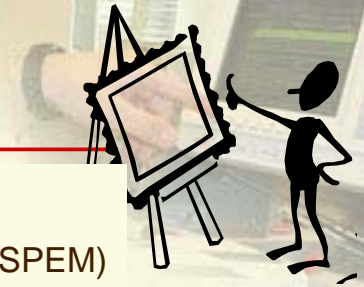


Example: Use of a Bridge-pattern

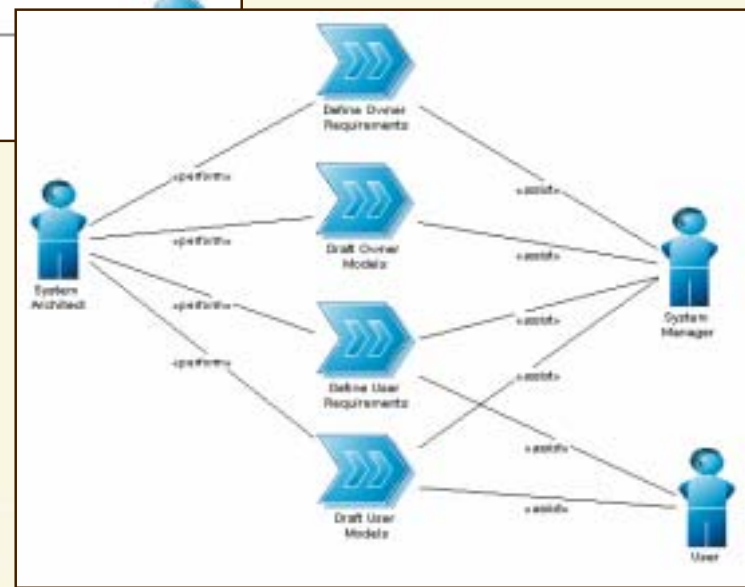
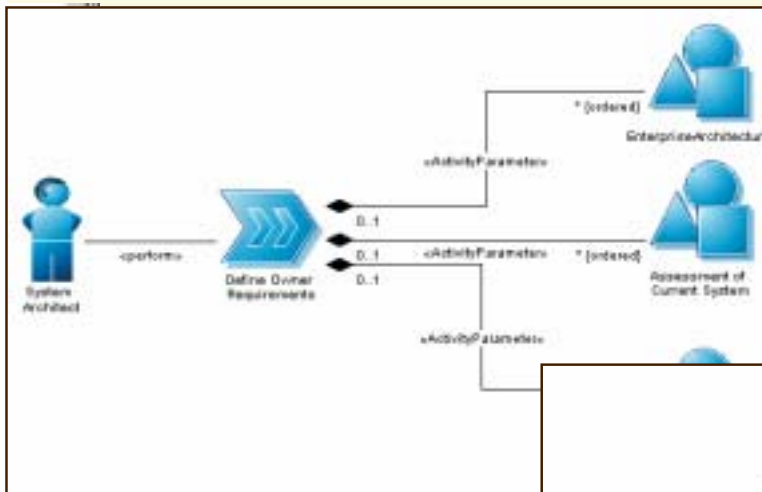
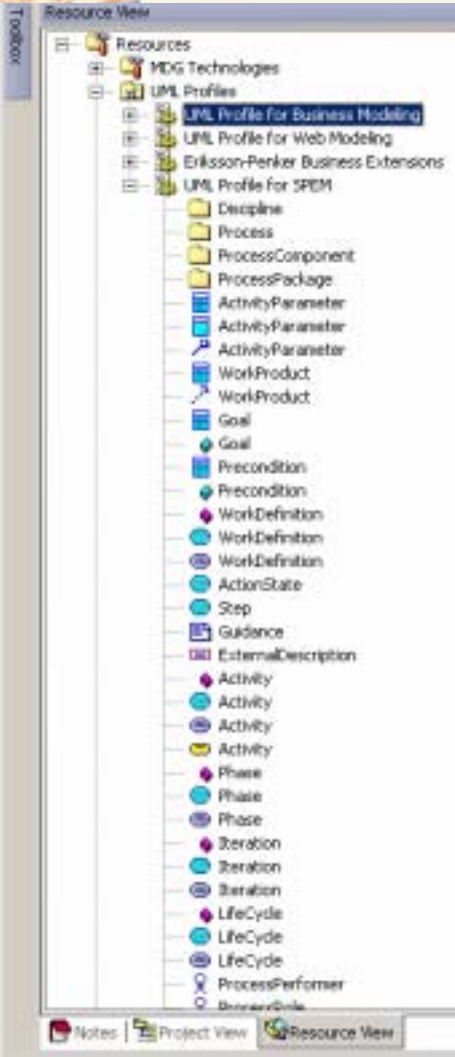
Resulting class structure after inserting the pattern



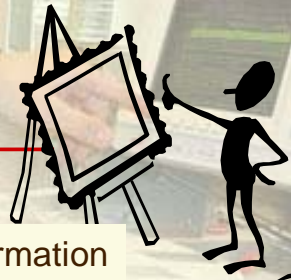
# UML Profiles



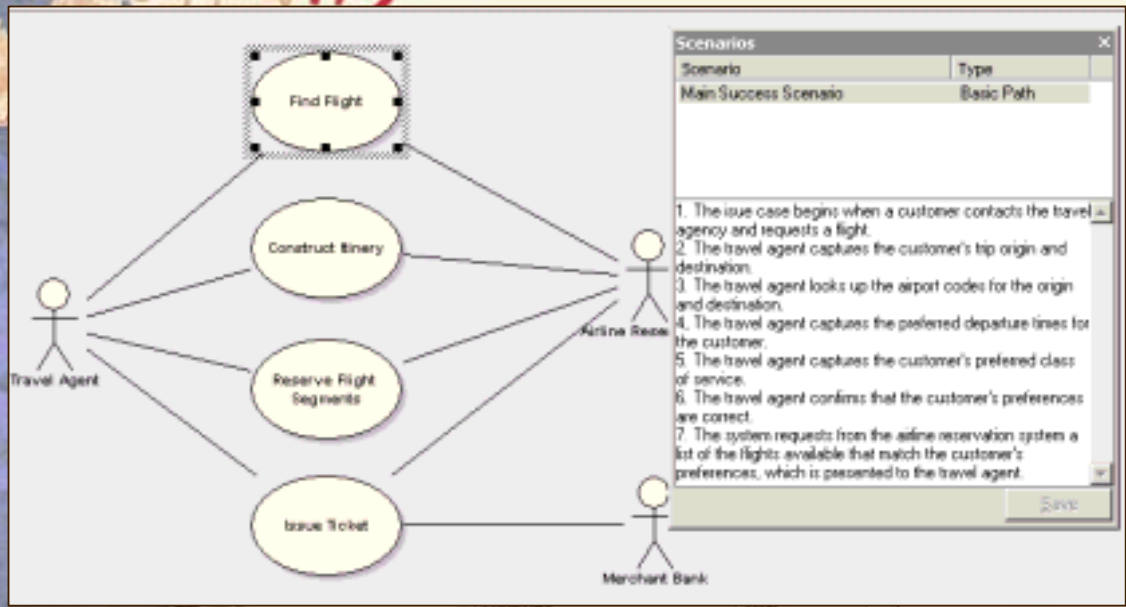
Example: Dedicated profile based on:  
 Software Process Engineering Metamodel Specification (SPEM)  
 (OMG Modeling UML extension)



# Additional Windows



Show specific information of selected element on the fly



Scenarios

Scenario	Type
Main Success Scenario	Basic Path

1. The use case begins when a customer contacts the travel agency and requests a flight.
2. The travel agent captures the customer's trip origin and destination.
3. The travel agent looks up the airport codes for the origin and destination.
4. The travel agent captures the preferred departure times for the customer.
5. The travel agent captures the customer's preferred class of service.
6. The travel agent confirms that the customer's preferences are correct.
7. The system requests from the airline reservation system a list of the flights available that match the customer's preferences, which is presented to the travel agent.

Save

Scenarios

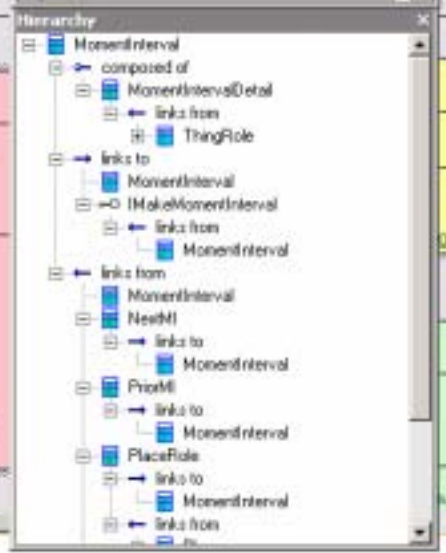
Scenario	Type

```

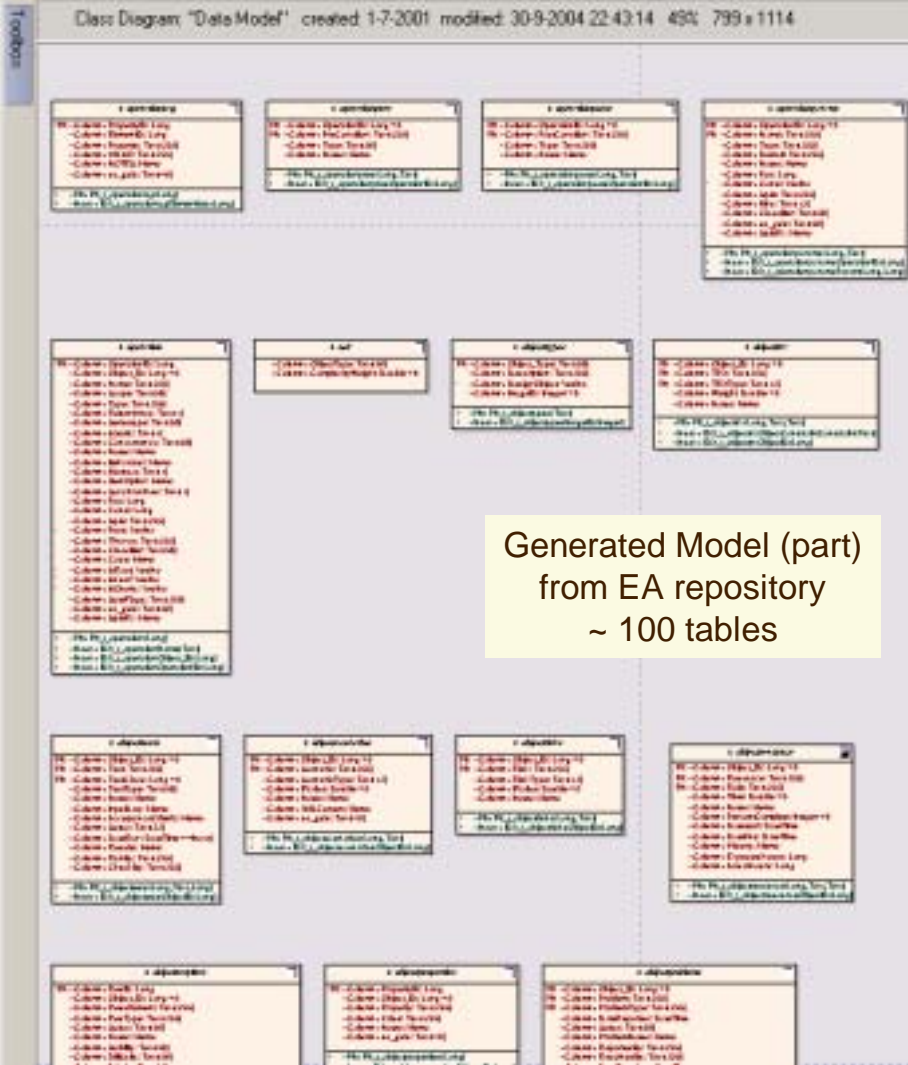
    classDiagram
        class MomentInterval {
            +start: date
            +dateOfDate: TimeOfInterval
            +priority:
            +total:
            +status:
            +makeMomentInterval()
            +addDetail()
            +calcTotal()
            +issueTotal()
            +complete()
            +cancel()
            +mi_generateNext()
            +mi_associateRTPlan()
            +mi_associateRTHead()
            +mi_composePlan/NoActual()
            +setMin()
            +assocAccessories()
        }
        class TimeInterval {
            +start:
            +end:
        }
        MomentInterval "0..*" -- "1" TimeInterval
    
```

Relationships

Relation	Target	Type	Visible
Association	MakeMomentInterval	Interface	X
Association	Priority	Class	X
Association	PlaceRole	Class	X
Association	PartyRole	Class	X
Association	NextMI	Class	X
Association	MomentInterval	Class	X
Aggregation	MomentInterval	Class	X







Generated Model (part)  
from EA repository  
~ 100 tables

```
CREATE TABLE t_refuser (
  refID Text(255),
  ToolID Text(50),
  Name Text(255),
  Type Text(255),
  Visibility Text(255),
  Namespace Text(255),
  Requirement Text(255),
  Constraint Text(255),
  Behavior Text(255),
  Partition Text(255),
  Description Memo,
  Client Text(255),
  Supplier Text(255),
  Link Text(255)
)

CREATE TABLE usys_system (
  Property Text(50),
  Value Text(50)
)

CREATE TABLE usysoldtables (
  TableName Text(50),
  NewName Text(50),
  ReOrder Long,
  FixCode YESNO NOT NULL
)

CREATE TABLE usysqueries (
  QueryName Text(50),
  NewName Text(50),
  FixCode YESNO NOT NULL
)

CREATE TABLE usysTables (
  TableName Text(50),
  ReOrder Long,
  DisplayName Text(50),
  Fromver Text(50),
  ToVer Text(50)
)

ALTER TABLE t_attribute ADD CONSTRAINT PK_t_attribute
PRIMARY KEY (ID)
|
ALTER TABLE t_attributeconstraints ADD CONSTRAINT PK_t_attributeconstraints
PRIMARY KEY (Constraint, ID)
|
ALTER TABLE t_attributetag ADD CONSTRAINT PK_t_attributetag
PRIMARY KEY (PropertyID)
|
ALTER TABLE t_authors ADD CONSTRAINT PK_t_authors
PRIMARY KEY (AuthorName)
|
ALTER TABLE t_cardinality ADD CONSTRAINT PK_t_cardinality
PRIMARY KEY (Cardinality)
|
ALTER TABLE t_category ADD CONSTRAINT PK_t_category
```

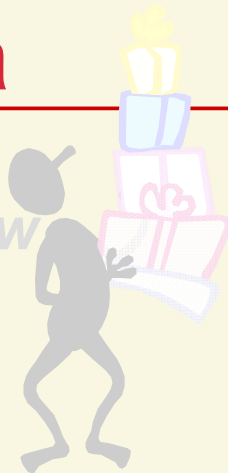
Generated SQL  
from Data Model  
(part)





# Agenda

Overview

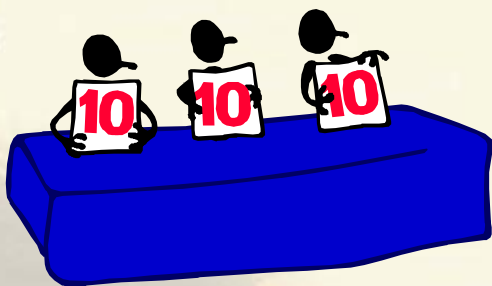


Some examples

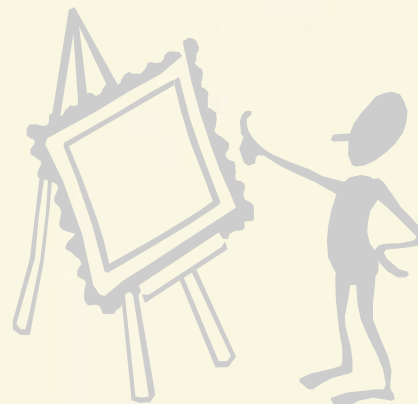
Final Word(s)



**ENTERPRISE  
ARCHITECT**



*Best bargain?*

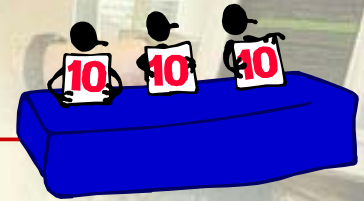


Architect(-ing) tool?





# Best bargain?



License (+1 year support):

Corporate Edition: US \$225 - US \$175

Professional Edition: US \$189 - US \$150

Desktop Edition: US \$125 - US \$85



Add-ons:

Zicom Visual UML Dictionary: US \$50 - US \$40

Visual Studio.Net MDG Link: US \$95 - US \$80

*Is it a bargain?*

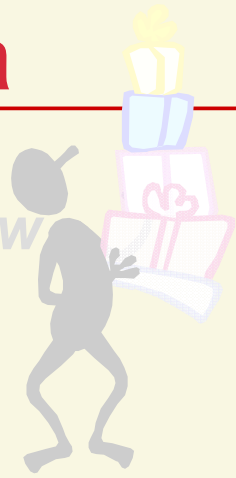
*Well,.... Depends...*

*But the rich feature-set, the ease of use and a price that's 1/20 of other products makes it certainly worthwhile to investigate it.*



# Agenda

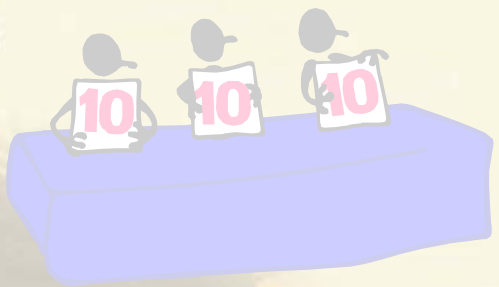
Overview



*Final Word(s)*



*Some examples*



*Best bargain?*



*Architect(-ing) tool?*



# Final Word(s)



- Useful as an architect tool? --> Yes
- Where/when helped EA us:
  - Get/keep overview
  - Knowledge transfer
  - Enforce architecture/design decisions:
    - patterns, profiles, rules
  - Everybody has the same tool
  - Multi-user: one repository
  - ...
- Lessons learned:
  - Start small; extend use along the way
  - Allow time for learning (give assignments)
    - Focus on using the tool, UML notation as well as OO thinking.
  - Tools are important, skills are essential
  - Define project/department wide standard of using UML
  - Reverse (Roundtrip) engineering requires strict way of working
  - ...



# Where to find more

---



## ■ Links:

- <http://www.sparxsystems.com.au/ea.htm>
- <http://www.xpdian.com/>
- [http://consulting.dthomas.co.uk/software\\_architecture\\_consulting/articles\\_resources.htm](http://consulting.dthomas.co.uk/software_architecture_consulting/articles_resources.htm)
- <http://www.sparxsystems.com.au/zm/ZicomMentor.htm>
- <http://pcoad.com/>
- <http://www.step-10.com/notes/index.html>
- [http://www.omg.org/technology/documents/spec\\_summary.htm](http://www.omg.org/technology/documents/spec_summary.htm)

## ■ Books:

- Patterns for Effective Use Cases, ISBN 0-201-72184-8
- Better Software Faster, ISBN 0-13-008752-1
- Streamlined Object Modeling (Patterns, Rules, and Implementation), ISBN 0-13-066839-7
- Java Modeling in Color with UML, ISBN 0-13-011510-X



Any questions ?

Thank you  
for your attention