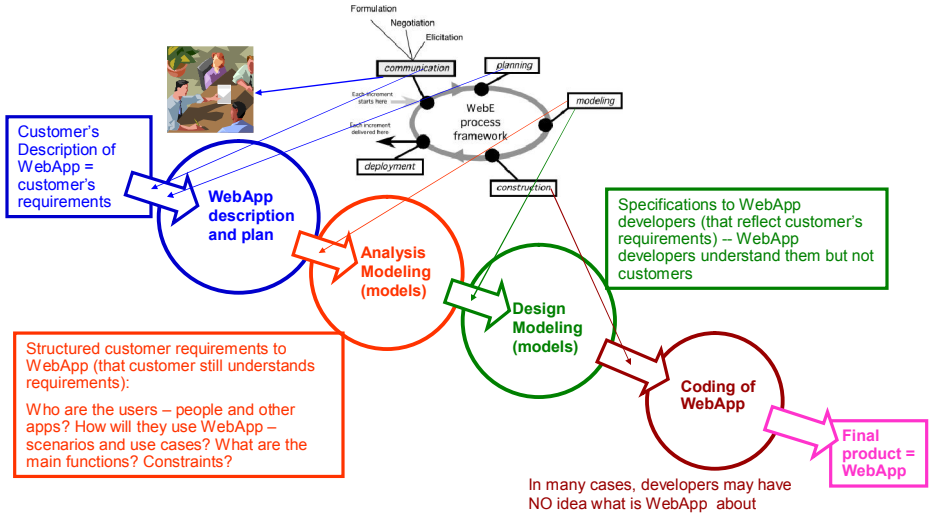
Topic 06

Web Engineering:
Analysis Modeling Phase

Why Do We Need Modeling (Analysis and Design) of WebApp?



- **Analysis Modeling** helps you to understand the nature of customer's requirements or the problem being addressed and the "shape" of the WebApp that will allow you to address that problem.
- **Design Modeling** is about understanding the internal structure of the WebApp to-be-developed and how this creates the shape of the WebApp that was identified by the analysis model.

Importance of Analysis Modeling

- It is **impossible** to begin the design and development (D&D) of a WebApp increment if you have no understanding of what is required.
- ***Analysis Modeling*** helps you to understand **customer's requirements in details --** that will allow Web Engineer to satisfy better user requirements (needs).
- ***Analysis Models*** provide **technical details** regarding
  1) content of WebApp (content model),
  2) interaction with or GUI of WebApp (interaction model),
  3) main functions and behavior of WebApp (functional model), and
  4) configuration or composition of WebApp (configuration model).
- **How much Analysis Modeling needed**? It depends on:
  - Size and complexity of the WebApp increment
  - Number of stakeholders (analysis can help to identify conflicting requirements coming from different sources)
  - Size of the WebE team
  - Degree to which members of the WebE team have worked together before (analysis can help develop a common understanding of the project)
  - Degree to which the organization's success is directly dependent on the success of the WebApp

Analysis Modeling = FUNCTION (size, complexity, team specs, experience, knowledge, etc.)

3

WebE Analysis Models: Goals and Outputs/Diagrams

WebE Analysis Models	Goal	Diagrams to be used to reflect analysis models	Tools to be used
Interaction Model	Describes the manner in which users interact with the WebApp.	■ Use Cases (UCs) ■ User interface prototypes	■ UMLet ■ MS Expression Studio ■ Visual Paradigm
Information model (or, Content Model)	Identifies the full spectrum of content to be provided by the WebApp. Content includes text, graphics and images, and video and audio data.	■ Content Objects (including, Data Objects) ■ Data Flow Diagrams (DFDs) ■ Content Model Trees (CMTs)	■ Microsoft Visio ■ IBM Rational Software ■ SmartDraw
Functional Model	Defines the operations that will be applied to WebApp content and describes other processing functions that are independent of content but necessary to the end user.	■ State Transition Diagrams (STDs) ■ Activity Diagrams (ADs) ■ Sequence Diagrams (SDs) ■ SwimLane Diagrams (SLDs)	■ UMLet or SmartDraw ■ UMLet ■ SmartDraw ■ SmartDraw
Configuration Model	Describes the environment and infrastructure in which the WebApp resides.	Components: ■ Hardware, operating systems ■ Software ■ Internet, browsers ■ Data Protocols ■ Security considerations etc.	■ UMLet or SmartDraw ■ MS Visio ■ IBM Rational Software ■ SmartDraw

4

Diagrams for the Interaction Modeling

- Use Cases (UCs)
- Graphic User Interface (GUI) prototypes

5

WebE Analysis Models: Goals and Outputs/Diagrams

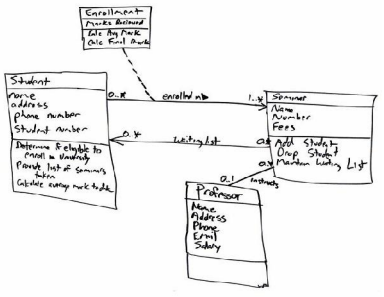
WebE Analysis Models	Goal	Diagrams to be used to reflect analysis models	Tools to be used
Interaction Model	Describes the manner in which users interact with the WebApp.	■ Use Cases (UCs) ■ User interface prototypes	■ UMLet ■ MS Expression Studio ■ Visual Paradigm
Information model (or, Content Model)	Identifies the full spectrum of content to be provided by the WebApp. Content includes text, graphics and images, and video and audio data.	■ Content Objects (including, Data Objects) ■ Data Flow Diagrams (DFDs) ■ Content Model Trees (CMTs)	■ Microsoft Visio ■ IBM Rational Software ■ SmartDraw
Functional Model	Defines the operations that will be applied to WebApp content and describes other processing functions that are independent of content but necessary to the end user.	■ State Transition Diagrams (STDs) ■ Activity Diagrams (ADs) ■ Sequence Diagrams (SDs) ■ SwimLane Diagrams (SLDs)	■ UMLet or SmartDraw ■ UMLet ■ SmartDraw ■ SmartDraw
Configuration Model	Describes the environment and infrastructure in which the WebApp resides.	Components: ■ Hardware, operating systems ■ Software ■ Internet, browsers ■ Data Protocols ■ Security considerations etc.	■ UMLet or SmartDraw ■ MS Visio ■ IBM Rational Software ■ SmartDraw

6

1. Use-Cases and Use-Case Scenarios

- It crucial for WebApp designers and developers to **understand users** and **create users' profiles.**
- **Use-Case is a collection of user scenarios that describe the thread of usage of a system by various users.**
- **Each SINGLE scenario is described from the point-of-view of an "actor" — a person or device that interacts with the software in some way**
- **Each SINGLE scenario should clearly answer the following questions:**
  - Who is the primary actor (user), the secondary actor (s)?
  - What are the actor's goals?
  - What preconditions should exist before the story begins?
  - What main tasks or functions are performed by the actor?
  - What extensions might be considered as the story is described?
  - What variations in the actor's interaction are possible?
  - What system information will the actor acquire, produce, or change?
  - Will the actor have to inform the system about changes in the external environment?
  - What information does the actor desire from the system?
  - Does the actor wish to be informed about unexpected changes?

Old days: Conceptual Sketches done manually



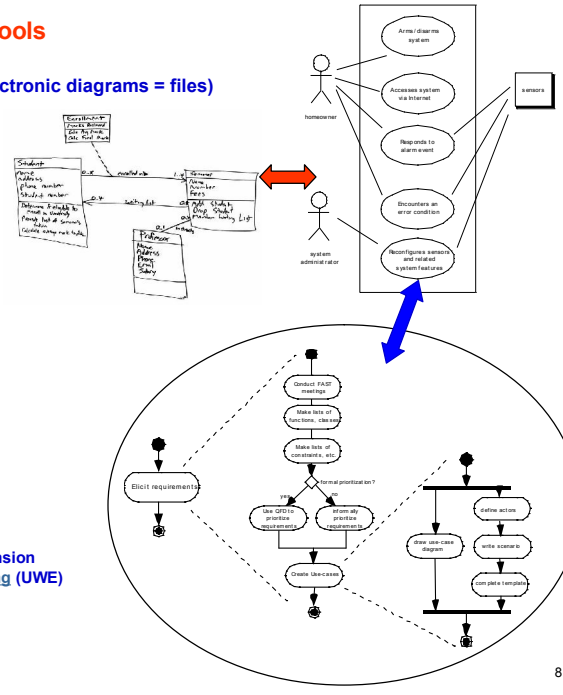
7

These days: Active Use of Tools

Modeling Languages

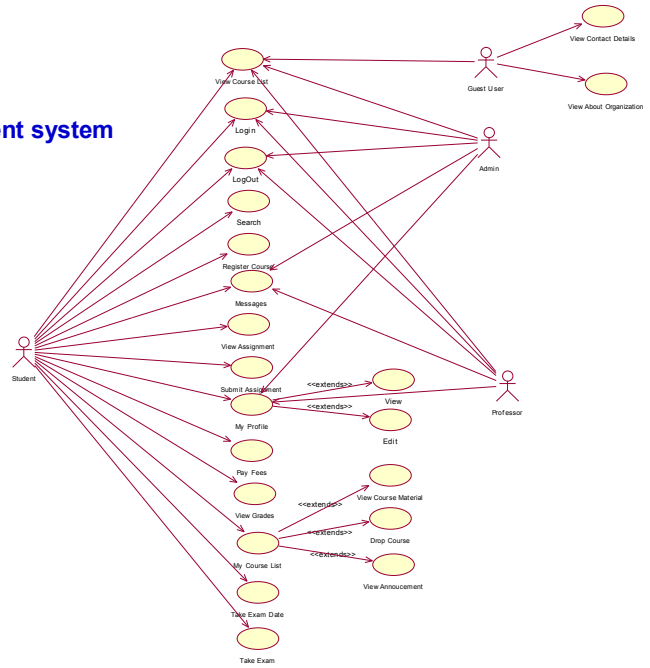
(to convert manual sketches into electronic diagrams = files)

- A ***modeling language*** (ML) incorporates a set of symbols notations, and/or terms, as well as the rules for establishing associations between them.
- A ***modeling language*** often has a formally structured representation as well as a set of graphical elements.
- Some MLs are general purpose (e.g., UML) and others are more specific (e.g., WebML)
  - **HDM - W2000**
  - **RMM**
  - **OOHDM**
  - **ARANEUS**
  - **STRUDEL**
  - **TIRAMISU**
  - **WebML**
  - **Hera**
  - **UML Web Application Extension**
  - **UML-based Web Engineering (UWE)**
  - **ACE**
  - **WebArchitect**
  - **OO-H**



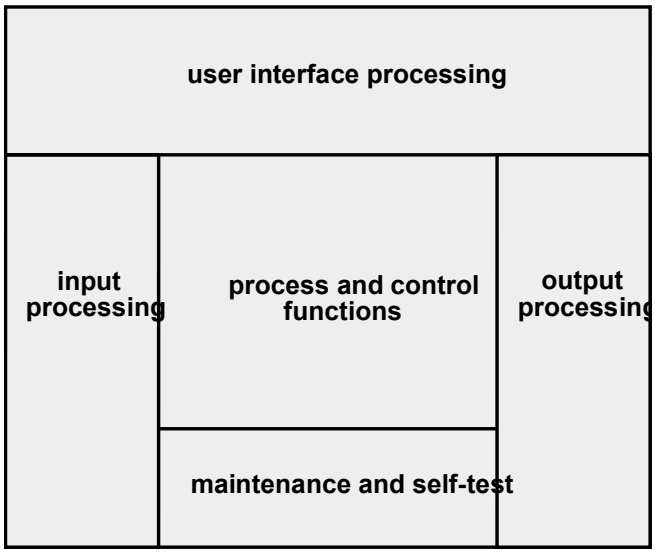
8

Use-Case Diagram: An example of Sakai (BlackBoard) learning management system (LMS)



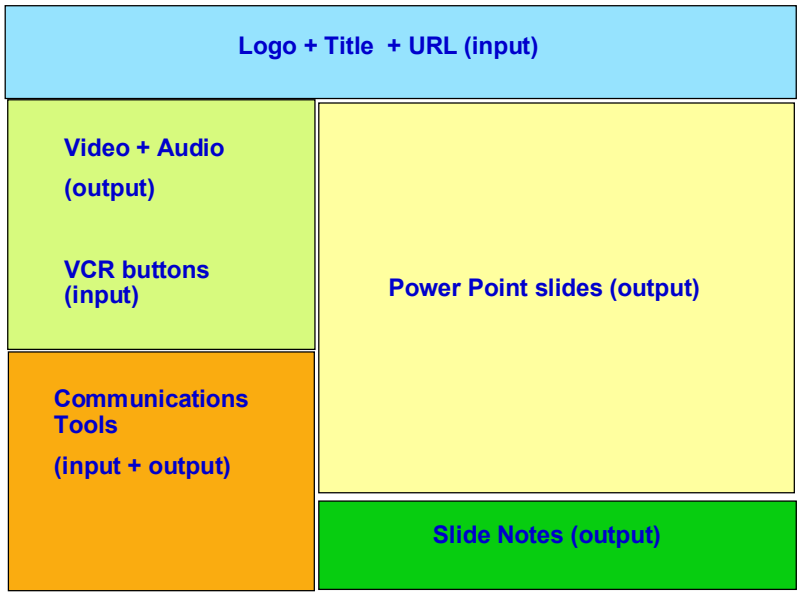
9

2. User Interface (GUI) Prototyping: A Template-Based Approach (Dr. Uskov's NSF CCLI project in 2000-2004)



10

GUI Template for e-Learning System (an example)



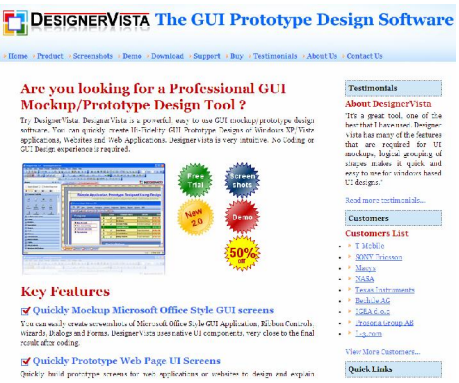
11

GUI of the InterLabs -- Advanced Web-Lecturing System with Streaming Media (Video, Audio, Data) at Bradley University (D&D 2002-2009)

12

Tools for User Interface (GUI) Prototyping: Design Software-Based Approach

http://www.microsoft.com/expression/default.aspx



http://www.designervista.com/index.html

13

16 User Interface Prototyping Tools

	Balsamiq	Pencil	OmniGraffle Pro	Visio Pro
One liner	Cool!	This can't really be a Firefox addon!	Amazingly simple and powerful drawing	Practical and business-like
Ramp up time	none	minuscule	small	small
Other expertise required	none	none	none	none
Supports interactivity	nope	nope	nope	nope
Full lifecycle product (rough to spec)	rough only	rough only	yes	somewhat
Fun to use	absolutely!	yes	yes	somewhat
Ease of creating a mockup of tree navigation	drag and drop and populate	no tree stencil!	had to download Yahoo stencils	straight-forward
Time to build mockup of tree navigation	15 min	1 hr	1 hr	30 min
Quirky?	nope	somewhat	nope	nope
Cost	$79	free	$199.95	$559.95

	Photoshop CS4	Illustrator CS4	Fireworks CS4	Dreamweaver CS4
One liner	Gold standard for bitmap tools	Gold standard for vector tools	Bitmap + Vector but I still don't get it...	Yes, there are countless HTML toolkits out there, but I still love it
Ramp up time	very large	very large	medium	large
Other expertise required	knowledge of bitmap tools	knowledge of vector tools	none (for this)	html, javascript, css
Supports interactivity	nope	nope	nope	yes
Full lifecycle product (rough to spec)	professional	professional	yes	yes
Fun to use	somewhat	somewhat	rendering is poor	yes
Ease of creating a mockup of tree navigation	from scratch	from scratch	no default tree!	had to find javascript tree control and figure out how to use
Time to build mockup of tree navigation	1 hr	1 hr	gave up	gave up
Quirky?	difference between Illustrator and PS can be confusing	difference between Illustrator and PS can be confusing	average	nope
Cost	$699	$599	$299	$399

	Flash CS4	Flex 3 Pro	Thermo (codename)	Powerpoint 2007
One liner	You can do whatever you want – total interactive flexibility	With some programming experience this rocks!	Not released yet – but from experience on the beta I have high hopes	Great prototyping tool for the earnest product manager
Ramp up time	huge	medium	large	negligible
Other expertise required	programming	programming	none	none
Supports interactivity	oh yes!	oh yes!	yes!	okay... yes
Full lifecycle product (rough to spec)	yes	rough – not really a visual tool	sort of	rough only
Fun to use	yes	subjective, but for me yes	yes!	no
Ease of creating a mockup of tree navigation	no built-in tree	easier than flash because of built-in tree, but still not automatic	medium	no tree
Time to build mockup of tree navigation	gave up	1 hr	1 hr	2 hr
Quirky?	not really	nope	nope	nope
Cost	$699	$699	?	$299

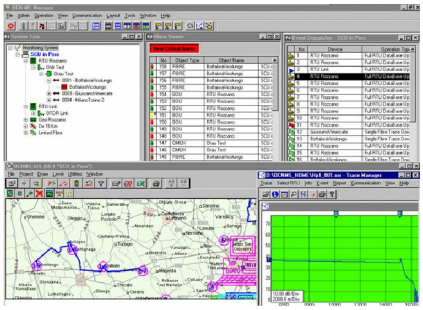
	Axure	iRise Pro	Microsoft Expression Blend 2	Microsoft Expression Design 2
One liner	Why?	Think process, large teams, ... but why?	Microsoft's Flash	Microsoft's Fireworks
Ramp up time	very large	very large	very large	large
Other expertise required	none	none	a little more than Flash	a little more than Fireworks
Supports interactivity	I think	I think	yes	nope
Full lifecycle product (rough to spec)	yes	yes	yes	professional
Fun to use	no	no	no	no
Ease of creating a mockup of tree navigation	didn't find tree immediately	didn't find tree immediately	didn't try – too tired to go on – opened the interface and started clicking around and just closed it	Stopped with Blend 2 – I hope the argument for these tools is not 'MS Office integration
Time to build mockup of tree navigation	gave up	gave up	gave up	didn't try
Quirky?	not really	not really	don't know	don't know
Cost	$589	$6,995	$499	$699 (for Studio – couldn't find individual)

Available at http://www.dexodesign.com/2008/11/07/review-16-user-interface-prototyping-tools/

14

Innovative Technology: Active Interface Prototype (examples)

- A prototype shows the layout of the user interface, the content, interaction mechanisms and overall aesthetic
- Supports validation with the client of the requirements and analysis



15

Diagrams for the Information (content) Modeling

- Content Objects (including, Data Objects)
- Data Flow Diagrams (DFDs)
- Content Model Trees (CMTs)

16

Analysis Models: Goals and Outputs/Diagrams

Analysis Models	Goal	Models or Diagrams
Interaction model	Describes the manner in which users interact with the WebApp.	■ Use Cases (UCs) ■ User interface prototypes
Information model	Identifies the full spectrum of content to be provided by the WebApp. Content includes text, graphics and images, and video and audio data.	■ Content Objects (including, Data Objects) ■ Data Flow Diagrams (DFDs) ■ Content Model Trees (CMTs)
Functional model	Defines the operations that will be applied to WebApp content and describes other processing functions that are independent of content but necessary to the end user.	■ State Transition Diagrams (STDs) ■ Activity Diagrams (ADs) ■ Swim Lane Diagrams (SLDs)
Configuration model	Describes the environment and infrastructure in which the WebApp resides.	Components: ○ Hardware, operating systems ○ Software ○ Internet, browsers ○ Data Protocols ○ Security considerations etc.

17

1. Content Objects (particularly, Data Objects), Attributes and Methods

A content object contains a set of attributes that act as an aspect, quality, characteristic, or descriptor of the object

A content object might be:
- a textual description of a product,
- an article describing news event,
- an action photograph,
- a user's response on a forum,
- a video file,
- an audio file,
- a collection of PPT slides,

etc.

Content Object Name:
Student

Attributes:
- ***firsrt_name***
- ***last_name***
- ***year of admission***
- ***major***
- ***courses taken***
- ***credits obtained***
- ***home_address***

etc.

Methods:
- ***to add to a course***
- ***to drop a course***
- ***to change major***

etc.

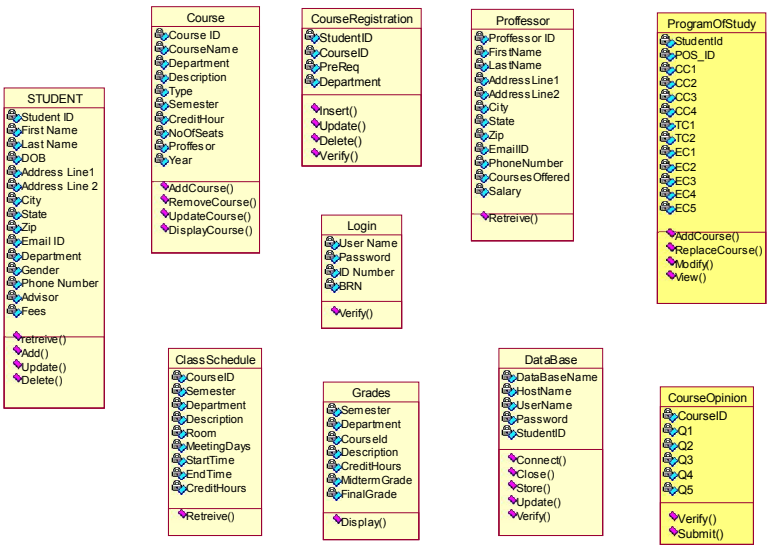
18

Content Objects = Components of the Content Model

- ***External entities*** (e.g., other systems, databases, people) that produce or consume information to be used by the WebApp
- ***Things*** (e.g., reports, displays, video images) that are part of the information domain for the problem
- ***Occurrences or events*** (e.g., a quote or an order) that occur within the context of a user's interaction with a WebApp
- ***Roles*** (e.g., retail purchasers, customer support, salesperson) played by people who interact with the WebApp
- ***Organizational units*** (e.g., division, group, team) that are relevant to an application
- ***Places*** (e.g., manufacturing floor or loading dock) that establish the context of the problem and the overall function of the WebApp
- ***Structures*** (e.g., sensors, monitoring devices) that define a class of objects or related classes of objects

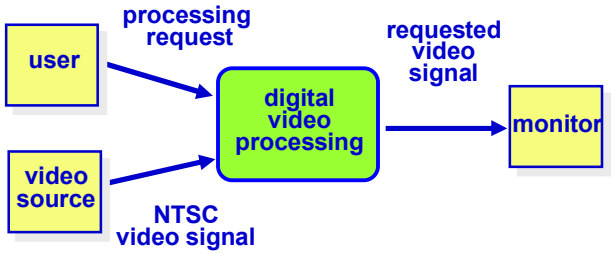
19

Content Objects (particularly, Data Objects), Attributes, and Methods



20

2. Data Flow Diagrams: A Context Diagram (with a SINGLE procession unit)

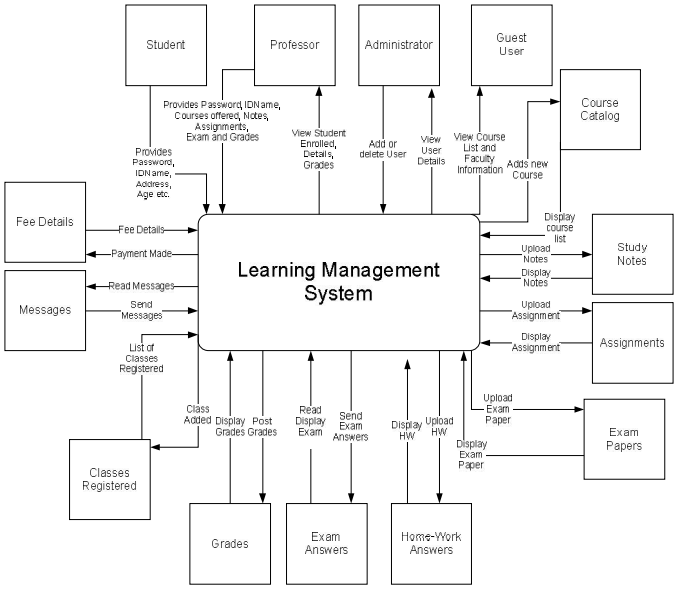


Important notes:

- A single processing unit
- No data storage units
- Multiple data sources (providers of data, receivers of data)

21

An Example of Data Flow Diagram: Context Diagram of a Learning Management System (like Blackboard or Sakai)

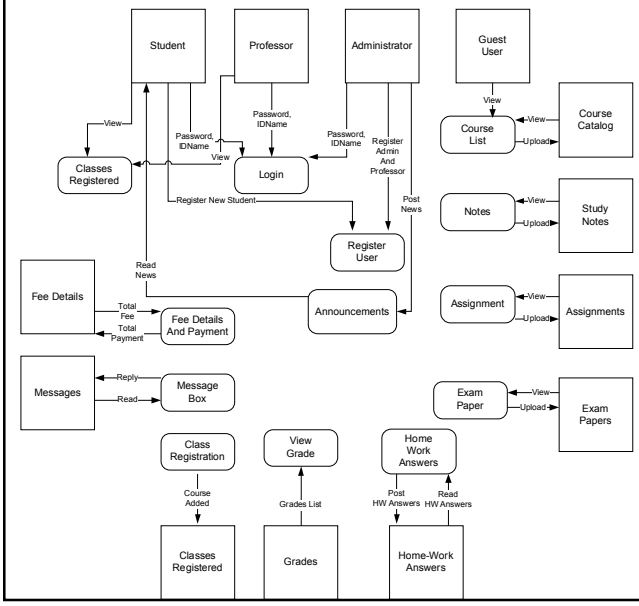


Data Flow Diagrams' Manuals:

- Course web site
- online

22

An Example of Data Flow Diagram: Level-0 Diagram of a Learning Management System (like Blackboard or Sakai)



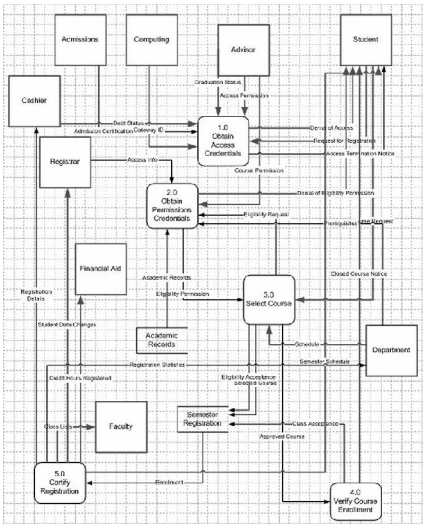
A single processing unit has been divided into 2…9 main subsystems (or, systems' functions)

Data storage units are allowed

Multiple data sources (providers of data, receivers of data)

23

An Example of Data Flow Diagram: Level-0 Data Flow Diagram of the "University of Missouri – St. Louis Student Registration System"

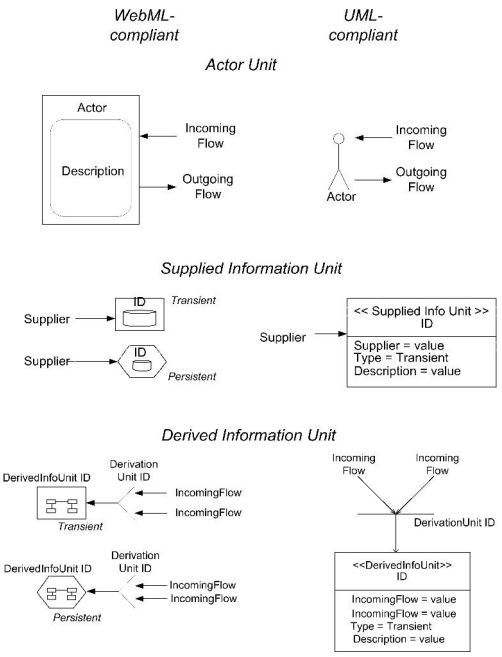


- A single processing unit has been divided into main sub systems (or, systems' functions)
- Data storage units are allowed
- Multiple data sources (providers of data, receivers of data)

http://www.umsl.edu/~sauterv/analysis/dfd/dfd_homework.html

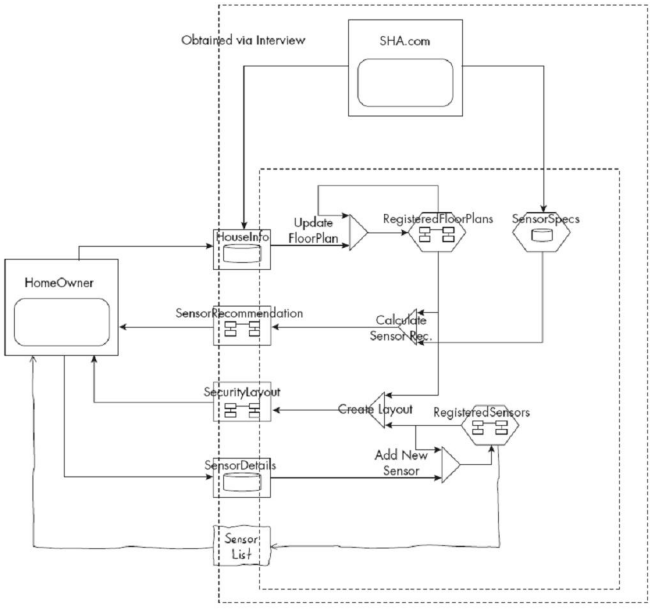
24

Web Information Exchange – Notation (in WebML and UML formats)



25

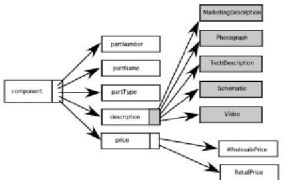
Web Information Exchange – An Example

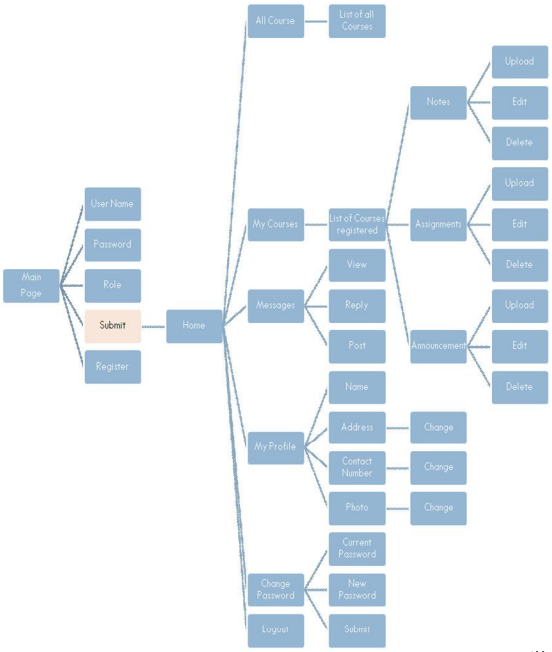


26

3. Data Trees

- In some cases, the interac and information model ma from a richer analysis usir based approach
- ***Data trees*** depict the rela among content objects an hierarchy of content maint WebApp.
- ***Menu trees*** depict the hie relationships among menu and/or the hierarchy of ob of a WebApp.





27

Diagrams for the Functional Modeling (or, Behavioral Modeling)

28

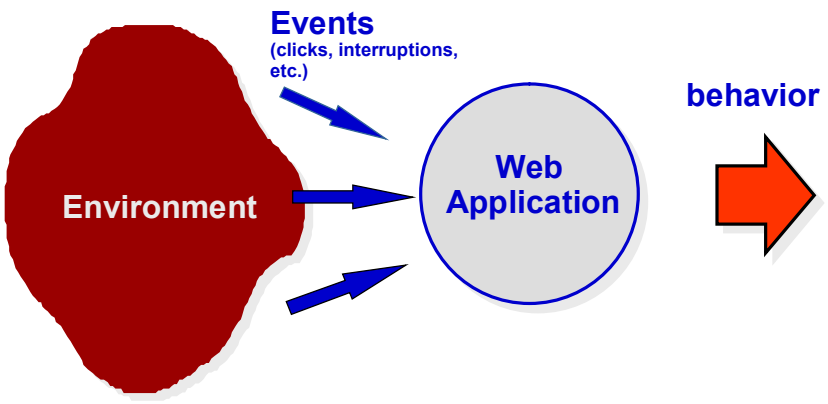
Analysis Models: Goals and Outputs/Diagrams

Analysis Models	Goal	Models or Diagrams
Interaction model	Describes the manner in which users interact with the WebApp.	■ Use Cases (UCs) ■ User interface prototypes
Information model	Identifies the full spectrum of content to be provided by the WebApp. Content includes text, graphics and images, and video and audio data.	■ Content Objects (including, Data Objects) ■ Data Flow Diagrams (DFDs) ■ Content Model Trees (CMTs)
Functional (behavioral) model	Defines the operations that will be applied to WebApp content and describes other processing functions that are independent of content but necessary to the end user.	■ State Transition Diagrams (STDs) ■ Activity Diagrams (ADs) ■ Sequence Diagrams (SDs) ■ SwimLane Diagrams (SLDs)
Configuration model	Describes the environment and infrastructure in which the WebApp resides.	Components: ○ Hardware, operating systems ○ Software ○ Internet, browsers ○ Data Protocols ○ Security considerations etc.

29

Behavioral Modeling

The **behavioral model** indicates **how WebApp will respond to external events.**



- State transition diagrams (STD) – most popular means of representing a software behavioral model.
- State diagrams are most useful when a user interaction triggers a change in the state of the WebApp—and hence changes the way in which it might react to a user.
- UML *state diagrams* describe dynamic behaviour of the WebApp as an interaction occurs.

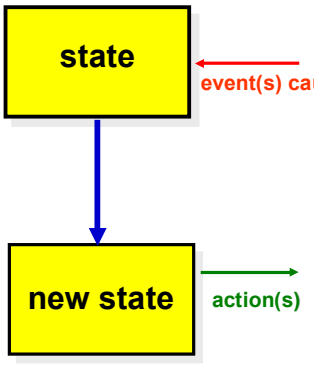
30

Behavioral Modeling

- The **behavioral model** indicates **how software will respond to external events.**
- To create the model, the analyst must perform the following steps:
  - **Identify events** that drive the interaction sequence and understand how these events relate to specific objects.
  - **Create a sequence of events** for each use-case.
  - **Build a state diagram** for the system.
  - **Review the behavioral model** to verify accuracy and consistency.

31

The States of a Web App



State - a set of observable circumstances that characterizes the behavior of a WebApp at a given time

State transition - the movement of WebApp from one state to another

Event - an occurrence that causes WebApp to exhibit some predictable form of behavior

Action - process that occurs as a consequence of making a transition

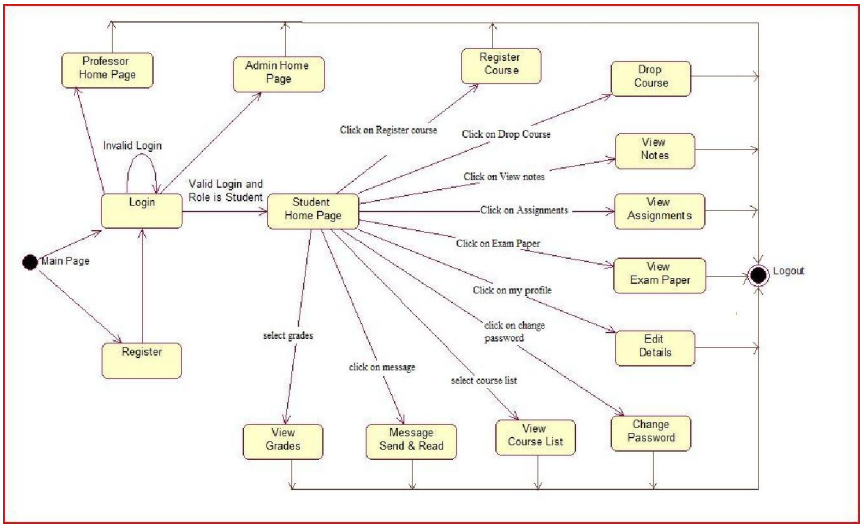
32

Creating Behavior Models

- **Evaluate all use-cases**
  to understand the sequence of interaction within the WebApp
- **Identify events**
  that drive the interaction sequence and how these events relate to specific objects
- **Create a sequence of events and list of corresponding states**
  or event-trace for each use-case
- **Build a state transition diagram**
  for the WebApp
- **Review the behavior model**
  to verify accuracy and consistency

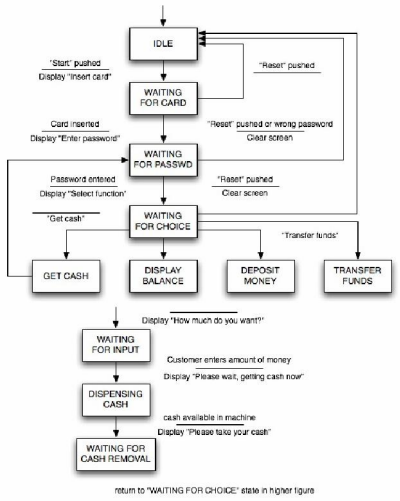
33

1. State Transition Diagram: An Example



34

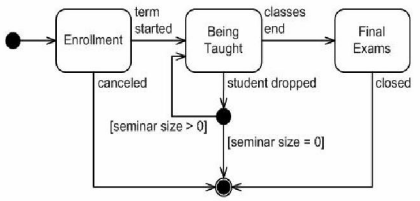
An Example: STD of Automated Teller Machine



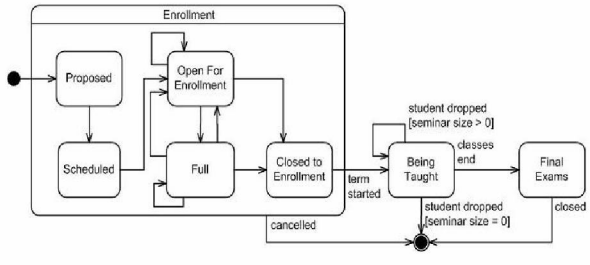
Source: http://yourdon.com/strucanalysis/wiki/index.php?title=Chapter_13

An Example: Multi-level STD

Top-level STD in UML format



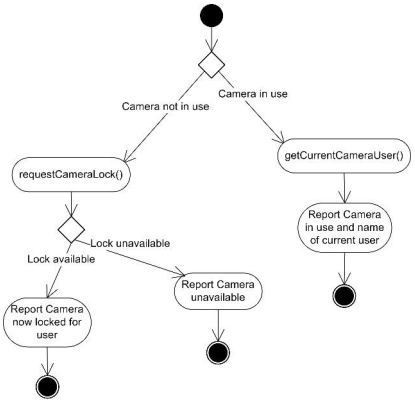
An STD with details for the "Enrollment" state in UML format



Source:
http://www.agilemodeling.com/artifacts/stateMachineDiagram.htm

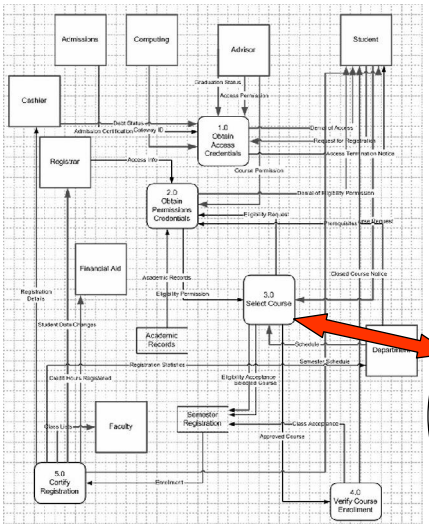
2. Activity Diagrams

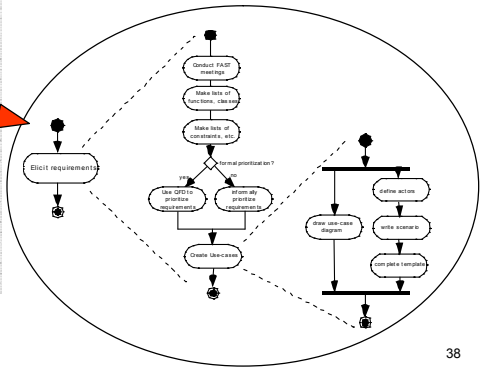
- Illustrates the processing flow and logical decisions (algorithms) within the flow.
- The construction details indicate how these operations are invoked, and the interface details for each operation are not considered until WebApp design commences.



37

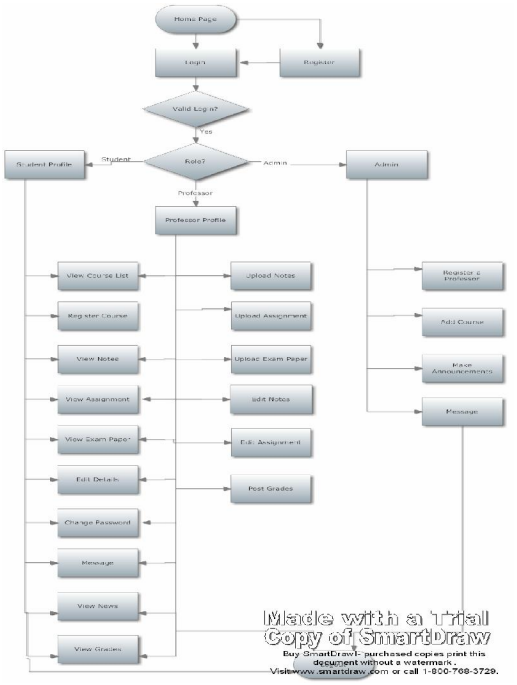
Modeling Logic Inside Processing Units in DFDs





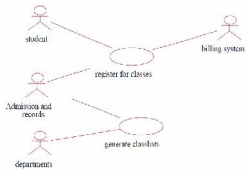
38

Activity Diagrams

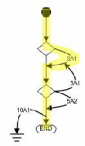


39

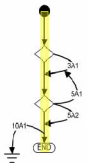
3. Processing of Use Cases: Sequence Diagrams or Swim Lane Diagrams

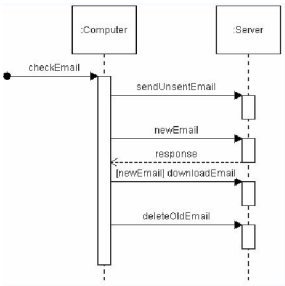


Another possible scenario would be to follow alternate flow "3A1" following it.

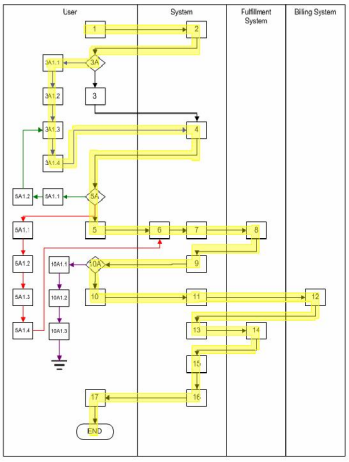


A use case scenario is a single path through the diagram.





Sequence diagrams describe how user actions collaborate with analysis classes (the structural elements of a system).



Source: http://tynerblain.com/blog/2007/04/10/what-are-use-case-scenarios/

40

Configuration Modeling

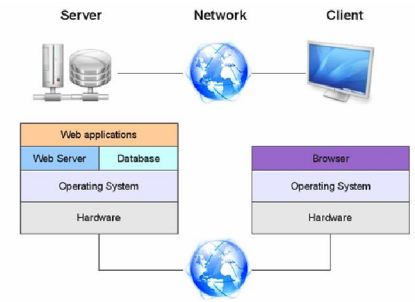
41

Analysis Models: Goals and Outputs/Diagrams

Analysis Models	Goal	Models or Diagrams
Interaction model	Describes the manner in which users interact with the WebApp.	■ Use Cases (UCs) ■ User interface prototypes
Information model	Identifies the full spectrum of content to be provided by the WebApp. Content includes text, graphics and images, and video and audio data.	■ Content Objects (including, Data Objects) ■ Data Flow Diagrams (DFDs) ■ Content Model Trees (CMTs)
Functional (behavioral) model	Defines the operations that will be applied to WebApp content and describes other processing functions that are independent of content but necessary to the end user.	■ State Transition Diagrams (STDs) ■ Activity Diagrams (ADs) ■ Sequence Diagrams (SDs) ■ SwimLane Diagrams (SLDs)
Configuration model *)	Describes the environment and infrastructure in which the WebApp resides.	Components: ■ Hardware, operating systems ■ Software ■ Internet, browsers ■ Data Protocols ■ Security considerations etc.

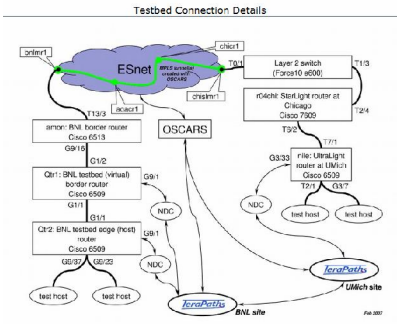
42

The Configuration (Implementation) Model



- On Server side:
  - operating system environments,
  - hardware,
  - Internet,
  - Web servers, data servers,
  - security considerations,
  - access to databases,
  - data protocols,
  - etc.
- On the client side:
  - Local OS
  - Browser software
  - Client hardware variations

Network Configuration Model: An Example (from U of Michigan)



Interface naming scheme SXX/YY:

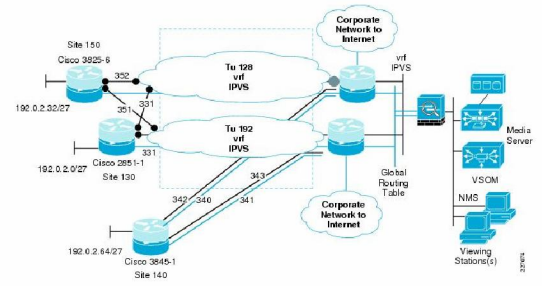
Router Configuration

NOTE: Cisco IOS requires qos be explicitly enabled with the following command:

Core QoS configuration of Qtr2/Qtr1/rile (additions to existing configuration):

Available at https://www.racf.bnl.gov/terapaths/software/network-hardware/configuration/hardware-configuration-examples/terapaths-bnl-umich-phase-3-testbed-router-configuration

Network Configuration Model: An Example (from CISCO company)



Router and Firewall Configurations

In this section the running configuration files from the routers shown in the previous topology diagram are included as reference.

vpn-jk7-7206-1

This configuration is for the upper WAN aggregation router shown in the topology diagram.

```
!
! Last configuration change at 13:06:43 edt Tue Aug 1 2006
! NVRAM config last updated at 13:07:50 edt Tue Aug 1 2006
!
upgrade fpd auto
version 12.4
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
!
hostname vpn-jk2-7206-1
!
boot-start-marker
boot system flash disk0:c7200-advsecurityk9-mz.124-15.T4
boot-end-marker
!
logging buffered 2000000
enable secret 5 [removed]
!
no aaa new-model
clock timezone est -5
clock summer-time edt recurring
```

Available at http://www.cisco.com/en/US/docs/solutions/Enterprise/Video/IPVS/IPVS_DG/IPVSchap7.html#wp1069779

Lab # 1:
Diagramming Tools for Analysis Modeling of Web-Based Systems and Design Modeling

4 REQUIRED tools to learn:

- ♦ UMLet or Visual Paradigm,
- ♦ IBM Rational Software,
- ♦ Microsoft Visio,
- ♦ Visual Paradigm,

OPTIONAL: ♦ SmartDraw – (7-day trial version)

CS593 Midterm Assignment
(see detailed assignment on course web site)

Learn the assigned existing Web system.

Create 8 diagrams for WebE Analysis and Design Models using 4 different diagramming tools, including
- **UMLet or Visual Paradigm,**
- **IBM Rational Software,**
- **Microsoft Visio,**
- **Visual Paradigm**

OPTIONAL: ♦ **SmartDraw – (7-day trial version)**

47

Why do we need to know all those diagrams and models, and tools?

48

(In the middle of Fall semester, this student was hired for a full-time position starting)

Dr. Uskov :

I recently had a round of interviews for a position as a Software Engineer. Prior to taking your class, I had no experience in Software Engineering. My expectation for the start of my career was to be a programmer for a few years to gain experience. Going into the interviews for a Software Engineering position, I thought I was probably getting in a little over my head.

Throughout all of the interviews, I was repeatedly asked questions that I was able to answer from what I have learned in your class.

I was not asked a single question about coding. Everything I was asked involved the analysis and design process and I was able to apply my knowledge of Use Cases, Class Diagrams, Data Flow Diagrams, State Transition Diagrams, Entity Relationship Diagrams, etc.

The guys that I interviewed with understood that I'm still a student and I'm no expert, but I was able to speak their language and display my ability to learn and understand what I will need to know as a Software Engineer.

I have since been offered a position as an entry level Software Engineer with the company and I believe that is due in large part to my experiences in your class.

Matt Gihring

49