

---

## Viewpoint-oriented requirements methods

---

---

---

---

---

---

---

---

---

## Objectives

- ◆ To explain the notion of viewpoints in RE
- ◆ To explain the notion of viewpoints in structured analysis
- ◆ To introduce emerging viewpoint approaches in RE

---

---

---

---

---

---

---

---

---

## Viewpoints-oriented requirements engineering

- ◆ RE involves the capture, analysis and resolution of many ideas, perspectives and relationships at varying levels of detail
- ◆ Methods based on rigid global schemes do not adequately address the diversity of issues presented by RE problems
- ◆ Methods based on the notion of *viewpoints* evolved to address the problem

---

---

---

---

---

---

---

---

## Example

- ◆ Consider the requirements for a system to be installed on a train which will automatically bring the train to a halt when it wrongly goes through a danger signal
- ◆ Some examples of viewpoints for this system and the requirements they encapsulate might be:
  - *Driver* Requirements from the train driver on the system
  - *Trackside equipment* Requirements from trackside equipment which must interface with the system to be installed
  - *Safety engineer* Safety requirements for the system
  - *Existing on-board systems* Compatibility requirements
  - *Braking characteristics* Requirements which are derived from the braking characteristics of a train.

---

---

---

---

---

---

---

---

## Advantages of viewpoint-oriented approaches

- ◆ They explicitly recognise the diversity of sources of requirements
- ◆ They provide a mechanism for organising and structuring this diverse information
- ◆ They imparts a sense of thoroughness (completeness)
- ◆ They provide a means for requirements sources or stakeholders to identify and check their contribution to the requirements

---

---

---

---

---

---

---

---

## SADT viewpoints

- ◆ Structured analysis and design technique (SADT) was developed in the late 1970s
- ◆ The notation consists of a rectangle representing some system activity and a set of four arrows
- ◆ SADT does not have an explicit notion of viewpoints  
Instead viewpoints are an intuitive extension of its modelling technique
- ◆ SADT “viewpoints” are sources and sinks of data
- ◆ In example “viewpoints” are shown in square brackets

---

---

---

---

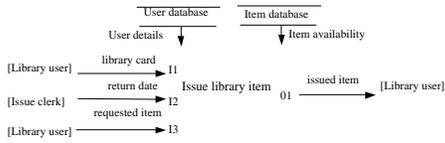
---

---

---

---

## Library example



---

---

---

---

---

---

---

---

## Controlled requirements expression (CORE)

- ◆ CORE was developed for the British Aerospace in the late 1970s by System Designers
- ◆ The CORE method is based on functional decomposition approach
- ◆ CORE is explicitly based on viewpoints
- ◆ Viewpoints defines two types of viewpoint:
  - *Defining viewpoints* Sub-processes of the system, viewed in a top-down manner
  - *Bounding viewpoints* Entities that interact indirectly with the intended system

---

---

---

---

---

---

---

---

## CORE method steps

- ◆ The CORE method is made up of 7 iterative steps:
  - Viewpoint identification
  - Viewpoint structuring
  - Tabular collection
  - Data structuring
  - Single viewpoint modelling
  - Combined viewpoint modelling
  - Constraint analysis

---

---

---

---

---

---

---

---

## Library example

### Step 1-Identifying viewpoints

- ◆ The first step involves identifying possible viewpoints
- ◆ From this initial list, defining and bounding viewpoints are identified
- ◆ There are no hard and fast rules for identifying relevant viewpoints

---

---

---

---

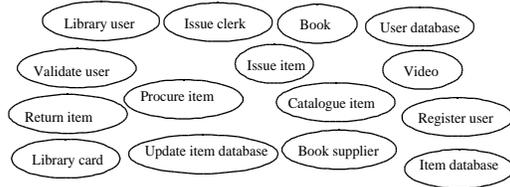
---

---

---

---

## Initial viewpoints



---

---

---

---

---

---

---

---

## Step 1 - Pruning viewpoints

- ◆ The last stage in viewpoint identification involves pruning the identified viewpoints into a set of bounding and defining viewpoints as shown in
- ◆ Each bubble represents the most abstract form of the viewpoint

---

---

---

---

---

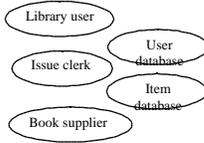
---

---

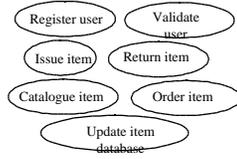
---

## Bounding and defining viewpoints

### Bounding Viewpoints



### Defining Viewpoints




---

---

---

---

---

---

---

---

---

---

## Step 2 - Viewpoint structuring

- ◆ Involves iteratively decomposing the 'target system' into a hierarchy of functional sub-systems
- ◆ Structurally bounding viewpoints are placed at the same level as the target system
- ◆ Each functional subsystem constitutes a viewpoint

---

---

---

---

---

---

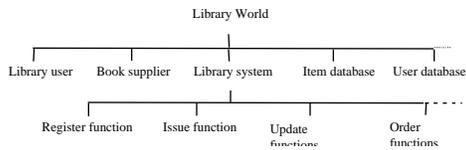
---

---

---

---

## Library system- viewpoint structuring




---

---

---

---

---

---

---

---

---

---

### Step 3 - Tabular collection

- ◆ A mechanism for gathering information about a viewpoint
- ◆ Each viewpoint is considered in turn with respect to the action it performs
  - Data used for these actions, the output data derived, the source of the data and the destination of the data
- ◆ Tabular collections serve the purpose of exposing omissions and conflicts in the information flow across viewpoints

---

---

---

---

---

---

---

---

---

---

### Library system- tabular collection

Source	Input	Action	Output	Destination
Library user	requested item	check item	issued item	Library user
			error message	Issue clerk
Library user	library card	validate user	loan default message	Issue clerk

---

---

---

---

---

---

---

---

---

---

### Steps 4-7

- ◆ The data structuring step involves decomposing data items into constituent parts and creating a data dictionary
- ◆ Step 5 and 6 involve modelling viewpoint actions using action diagrams
- ◆ An action diagram is similar in notation to an SADT diagram
- ◆ The final step in CORE involves performing constraint analysis on the system as a whole

---

---

---

---

---

---

---

---

---

---

## Problems with CORE

---

- ◆ Poorly defined notion of a viewpoint
  - Difficult to say what is and what is not a valid viewpoint
- ◆ Analysis focuses on internal perspectives - defining viewpoints
  - Bounding viewpoints not analysed beyond been seen as sinks and sources of data
- ◆ Difficult to integrate other requirements methods

---

---

---

---

---

---

---

---

## Viewpoint-oriented system engineering

---

- ◆ Developed at Imperial College, London in the early 1990s
- ◆ Viewpoint-oriented system engineering is a framework for integrating development methods
- ◆ Viewpoints used viewpoints to partition and distribute the activities and knowledge of the participants in software development
- ◆ Viewpoints capture the role and responsibility of a participant at a particular time

---

---

---

---

---

---

---

---

## Viewpoint template

---

- ◆ A Viewpoint can be thought of as a template describing:
  - Style or representative scheme what it sees
  - Domain
  - Specification
  - Work plan
  - Work record

---

---

---

---

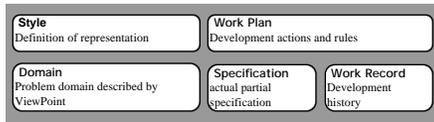
---

---

---

---

## Standard viewpoint template slots



---

---

---

---

---

---

---

---

## Viewpoint configurations

- ◆ Viewpoints can be organised into configurations
- ◆ A configuration may consist of
  - Templates with different styles, 'viewing' the same partition of the problem domain, or
  - Templates with the same style 'viewing' different partitions of the problem domain

---

---

---

---

---

---

---

---

## Library example

- ◆ Consider a library item presented the user at the issue desk for borrowing, returning or reserving
- ◆ 'Library world' can be partitioned into the domains of the issue desk and the library user
- ◆ Data-flow and state transition schemes are used to model the library item from point of view each domain

---

---

---

---

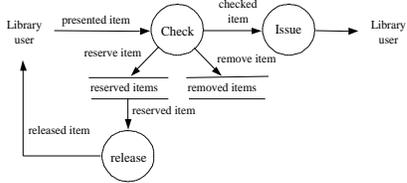
---

---

---

---

## Data-flow model -Issue desk domain




---

---

---

---

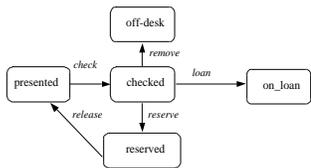
---

---

---

---

## State transition model -Issue Desk Domain




---

---

---

---

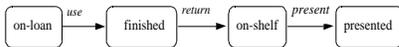
---

---

---

---

## State transition - Library user domain




---

---

---

---

---

---

---

---

## Conflict resolution

- ◆ Important to ensure that consistency between different representations of the domains
- ◆ For similar styles conflicts are resolved by checking for the loss of continuity between the models
- ◆ For different styles the correspondences between representation schemes need to be identified to facilitate consistency checking

---

---

---

---

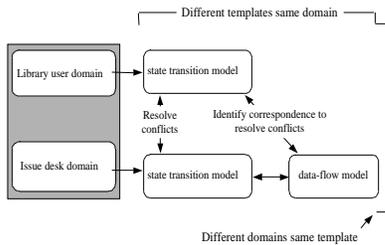
---

---

---

---

## Consistency checking




---

---

---

---

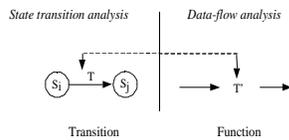
---

---

---

---

## Correspondence between transition and function




---

---

---

---

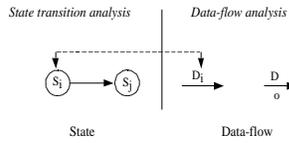
---

---

---

---

## Correspondence between state and data




---

---

---

---

---

---

---

---

---

---

## Mapping on different styles same domain

Issue desk DFD	Issue desk ST
check	check
issue	loan
release	release

---

---

---

---

---

---

---

---

---

---

## Mapping on different domains same style

Issue desk ST	Library user ST
presented	presented
on-loan	on-loan

---

---

---

---

---

---

---

---

---

---

## Viewpoint-oriented requirements definition

- ◆ Developed at the University of Lancaster
- ◆ Mainly intended for specifying interactive systems
- ◆ Based viewpoints that focus on user issues and organisational concerns
- ◆ The uses a service oriented model for viewpoints
- ◆ VORD defines two main types of viewpoints; direct and indirect

---

---

---

---

---

---

---

---

## Direct and indirect viewpoints

- ◆ Direct viewpoints
  - Interact directly with the intended system
  - Correspond directly to clients in that they receive services the system and provide control information
  - Include operators/users or other sub-systems interfaced to the system being analysed
- ◆ Indirect viewpoints
  - Do not interact directly with the intended system
  - Indirect viewpoints have an 'interest' in some or all the services which are delivered by the system
  - Generate requirements which constrain the services delivered to direct viewpoints
  - Includes organisation, environment, engineering and regulatory viewpoints

---

---

---

---

---

---

---

---

## Examples of direct and indirect viewpoints

- ◆ A systems planning viewpoint which is concerned with future delivery of library services (indirect)
- ◆ A library user viewpoint which is concerned with accessing the system services through the internet (direct)
- ◆ A trade-union viewpoint which is concerned with the effects of system introduction on staffing levels and library staff duties (indirect)

---

---

---

---

---

---

---

---

## Viewpoint-oriented requirements validation

- ◆ Uses viewpoints to support early requirements validation
- ◆ Objective of the approach is identify and classify problems related to completeness and correctness

---

---

---

---

---

---

---

---

## Viewpoints, perspectives and views

- ◆ Viewpoint is defined as a standing position used by an individual when examining a universe of discourse
- ◆ A perspective is defined as a set of facts observed and modelled according to a particular aspect of reality
- ◆ A view is defined as an integration of these perspectives
- ◆ A viewpoint language is used to represent the viewpoints

---

---

---

---

---

---

---

---

## Method steps

- ◆ Involves at least two analysts (viewpoints) using VWPL
- ◆ A view is constructed by describing the problem using three perspectives; data, process and actor perspectives
  - Analysts use the is-a and part-of hierarchies to improve their own view
- ◆ Perspectives and hierarchies are analysed and a 'list of discrepancies' and 'types of discrepancies' produced
- ◆ Perspectives are integrated into a view
  - Expressed in the process perspective together with the hierarchies
- ◆ After two views are available compare the different viewpoints for correctness and completeness

---

---

---

---

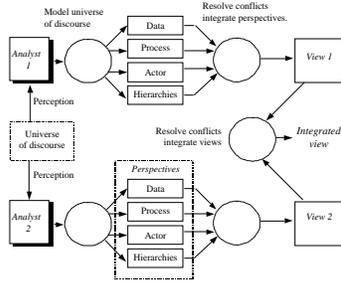
---

---

---

---

## Viewpoint-based requirements validation process



---

---

---

---

---

---

---

---

## Key points

- ◆ Requirements engineering is a distributed process involving many participants with different interests
- ◆ A viewpoint is a collection of information about a system or related problem, environment or domain which is collective from a particular perspective
- ◆ Structured analysis techniques do not have explicitly defined viewpoints

---

---

---

---

---

---

---

---