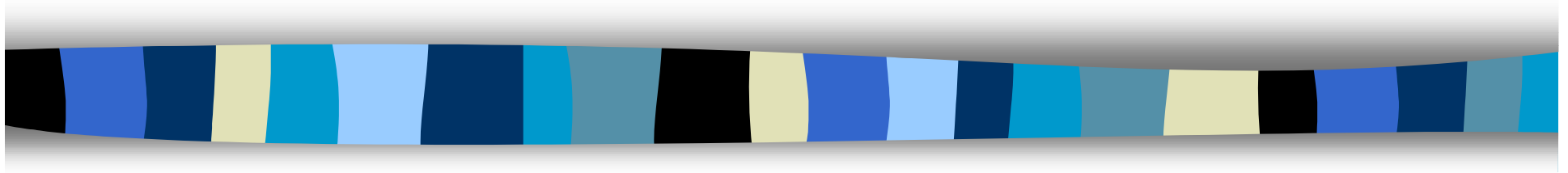


Use Case Maps as a Feature Description Notation



Daniel Amyot

SITE, University of Ottawa, Canada

damyot@site.uottawa.ca

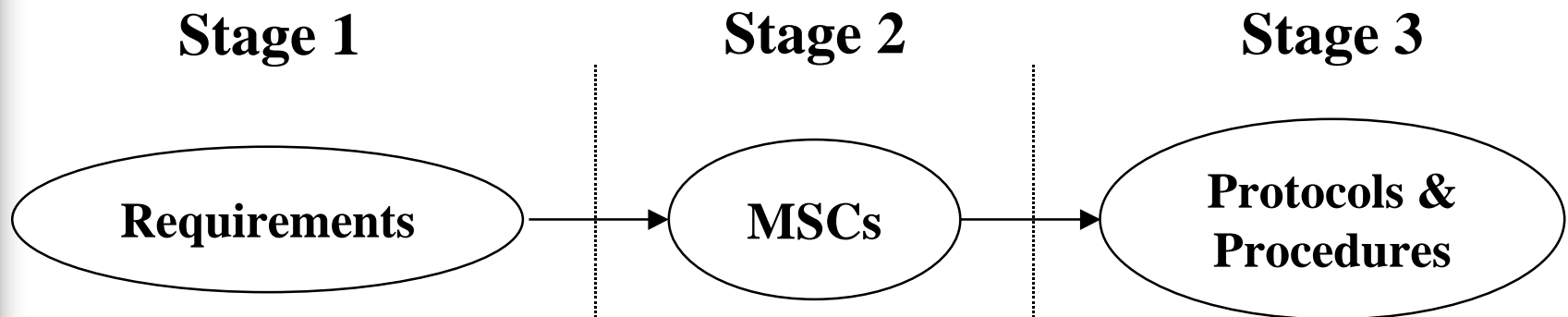
FIREworks'00, Glasgow, May 15, 2000



1. Introduction

- Telecommunications systems face difficult challenges:
 - Increasingly complex functionalities, architectures and protocols
 - New dimensions: mobility, IP, policy-driven negotiations, evolution at run-time
- Pressure is on industry and on standardisation bodies...

Common Methodology



Stage 1: Informal Service Descriptions

Stage 2: Message Sequence Information (Scenarios)

Stage 3: Protocol/Procedure Specifications

- Already uses scenarios
- Needs improvement to cope with new realities



ITU-T Question for Study

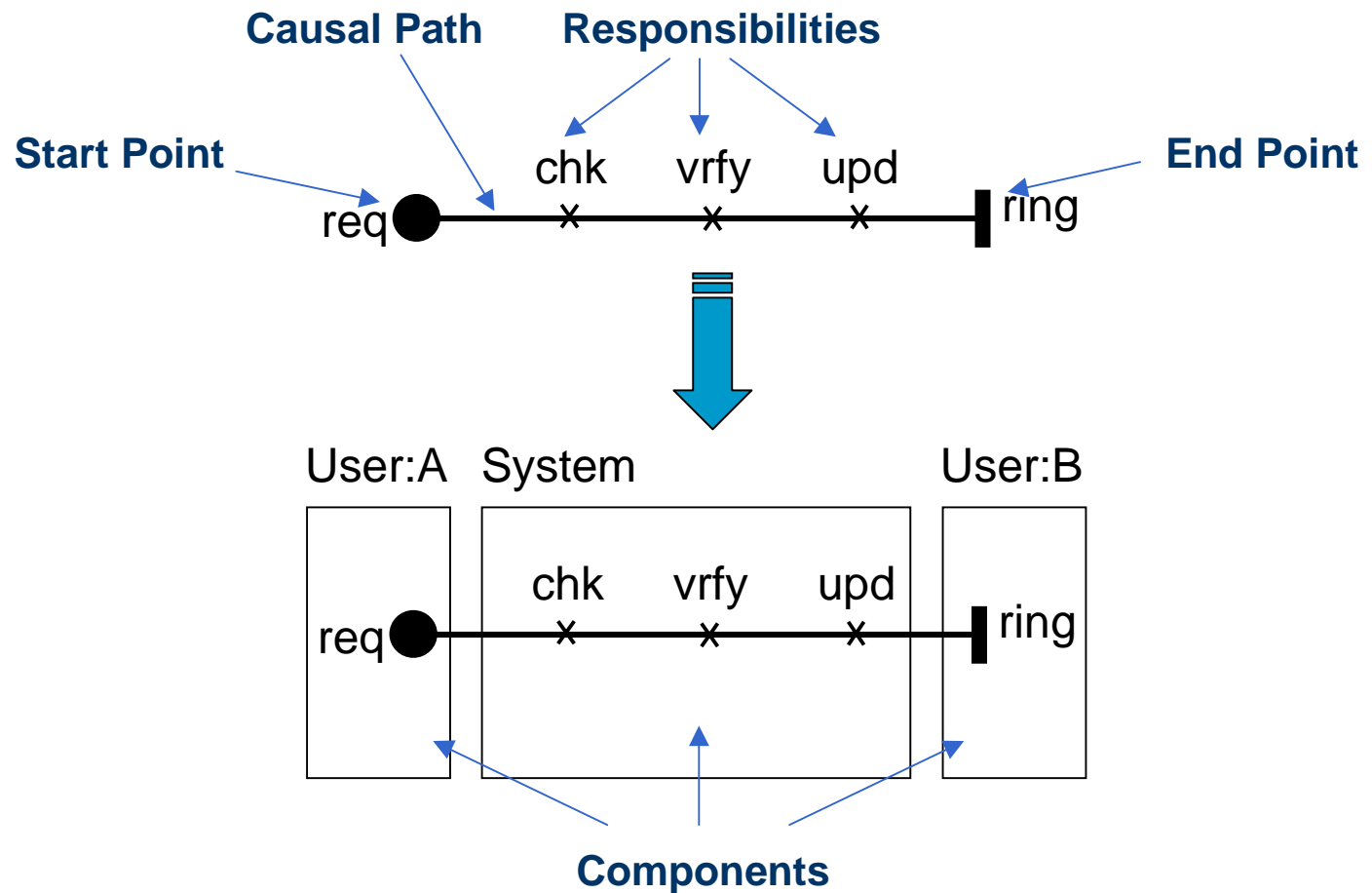
- **User Requirements Notation (URN)**, to complement existing notations such as MSC, SDL, and UML
- Recently approved by SG-10
- New Recommendation by 2003
- Focus on early stages of design, with **scenarios**
- No states, messages, or components required
- Reusability and allocation of scenarios
- Dynamic refinement capabilities
- Modelling of agent systems, early performance analysis, and early detection of interactions



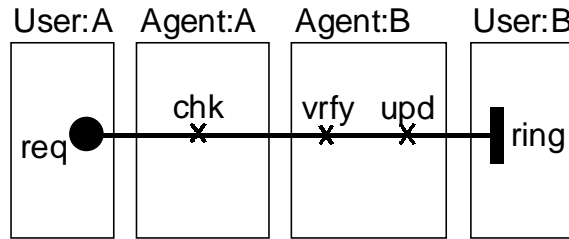
2. Use Case Maps

- Scenarios describing causal relationships between responsibilities
- UCM scenarios can be allocated to abstract components.
- Useful for describing features visually
- Proposed as a solution to SG-10's question.

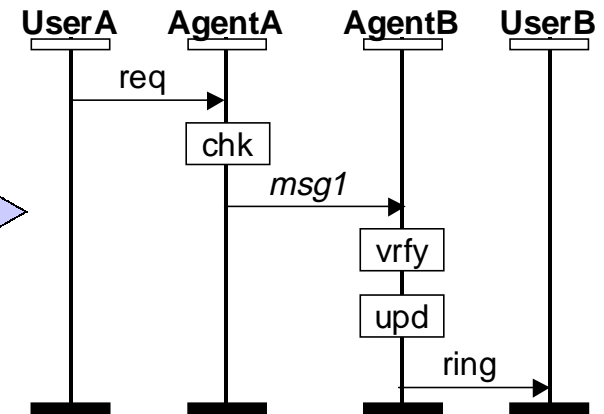
Basic Notational Elements



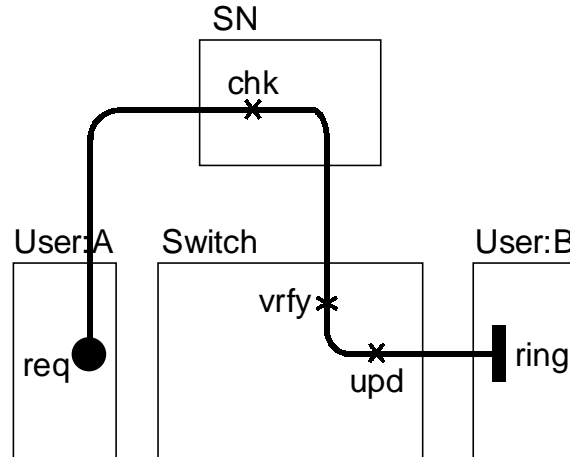
UCMs, Messages, Architectures



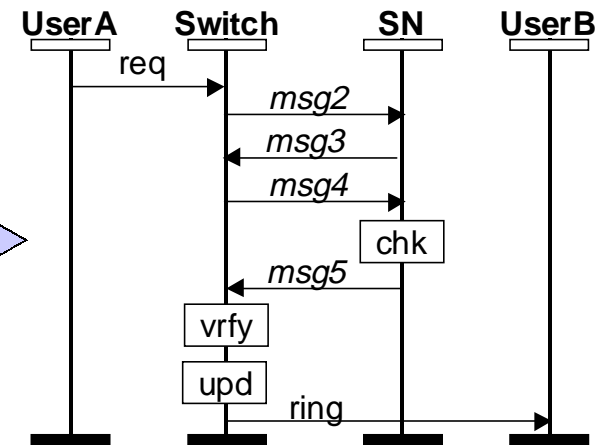
a) UCM path on an agent-based architecture



c) A possible MSC for (a)



b) UCM path on an IN-based architecture



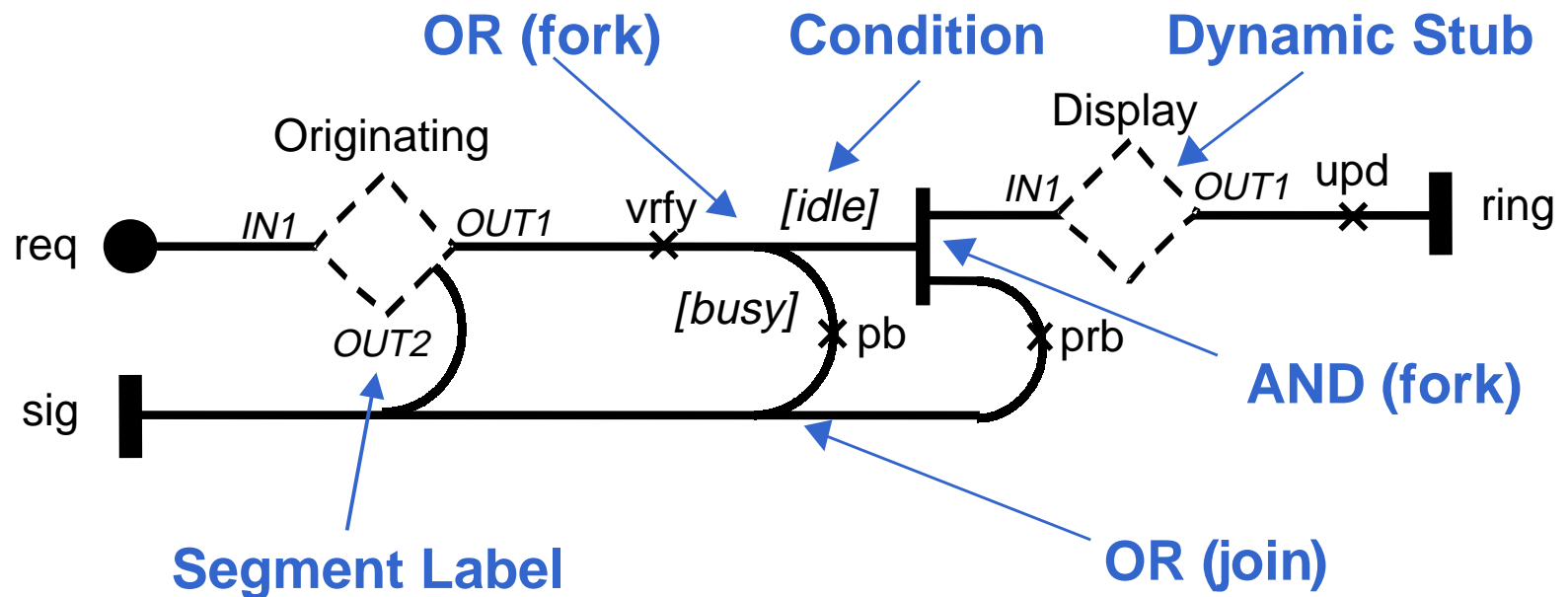
d) A possible MSC for (b)



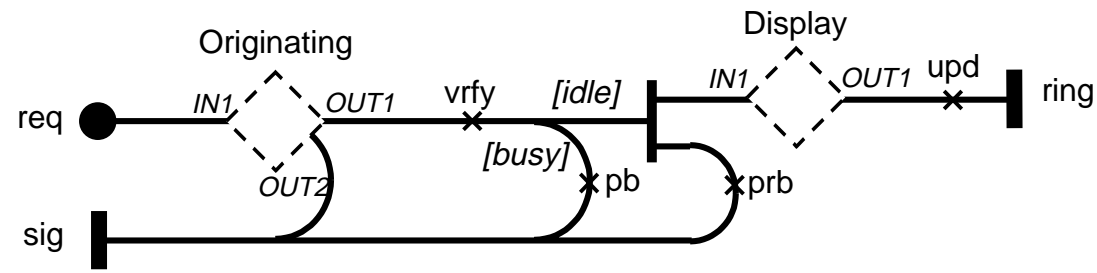
UCMs, Messages, Architectures

- We observed several benefits:
 - Combine behaviour and structure, although components are optional
 - Scenario paths are highly reusable
 - UCM scenarios survive evolutions of architectures better than MSCs
 - No early commitment to messages, protocols, and communication constraints
 - UCM view not clouded by design decisions
 - Focus is on functionalities and capabilities

UCMs and Scenario Integration



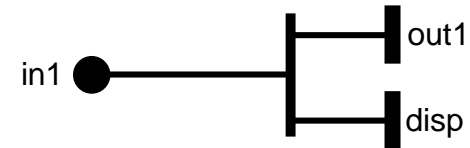
UCMs and Scenario Integration



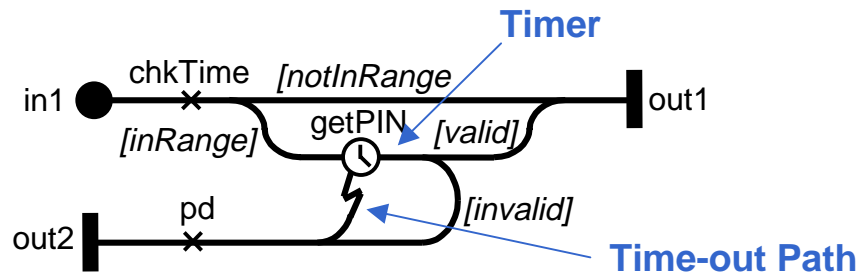
a) Basic Call map



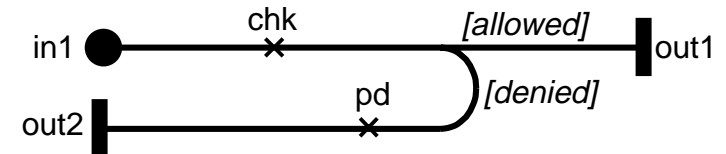
b) *DEFAULT* plug-in map



c) *CND* plug-in map



d) *TEENLINE* plug-in map



e) *OCS* plug-in map



Dynamic Stubs

- Contain *selection policies*
 - if registered to *CND*
then select *CND* plug-in
else select *DEFAULT* plug-in
- Are used to add features
 - Add new/existing plug-in
 - Add new stub with plug-ins
- Localise FI detection

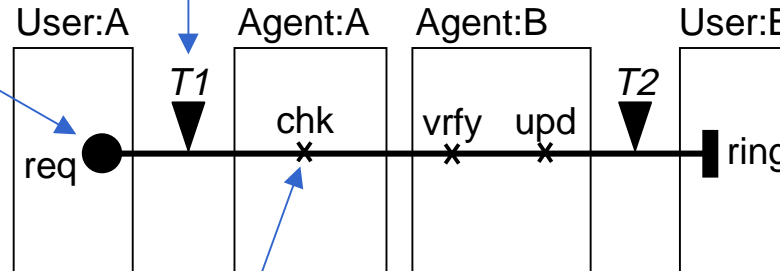
Performance & Agent Annotations

Arrival

Characteristics :

- Exponential, or
- Deterministic, or
- Uniform, or
- Erlang, or
- Other

Timestamp



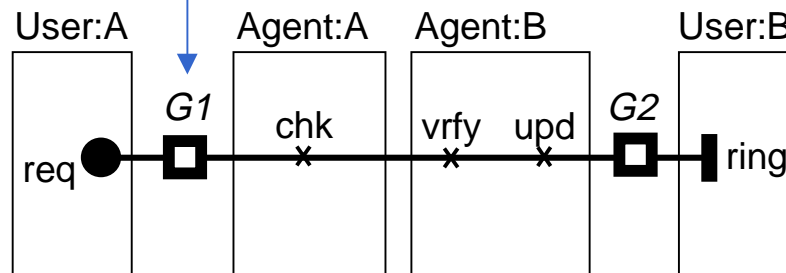
Response Time Requirement :

- From $T1$ to $T2$
- Name
- Response time
- Percentage

Responsibilities :

- Attached to data store
- Perform service requests

Goal Tag



My New Goal :

- From $G1$ to $G2$
- Preconditions
- Postconditions
- Description



Additional Notation Elements

- Component types and attributes
- Dynamic components and dynamic responsibilities
 - Model structural changes in a static way
- Path interactions
- Aborts
- Failure points



3. UCMs as a Feature Description Notation

- Interesting properties for URN:
 - Large-scale issues visible at a glance
 - Scenarios mapped to various architectures
 - Variations of run-time behaviour and of structures
 - Scenarios structured and integrated incrementally
 - Early detection of feature interactions
 - Early performance analysis



Areas of Application

- Not just telecommunications systems:
 - Airline reservation, elevators, railway, agent systems, network management applications, Web applications, graphical user interfaces, drawing packages, multimedia applications, banking applications, object-oriented frameworks, "work patterns" of software engineers, etc.



UCM Navigator Tool

- UCM notation (path and component)
- Maintains various logical bindings
- Flexible GUI (scrollable, resizable, zoomable workspace, align/distribute)
- Export/import in XML (UCM DTD)
- Reports generation (PostScript/PDF)
- Export UCMs in EPS, MIF, CGM
- Many platforms (Linux-Sparc/Intel, Solaris, HP/UX, Windows 9x/NT)

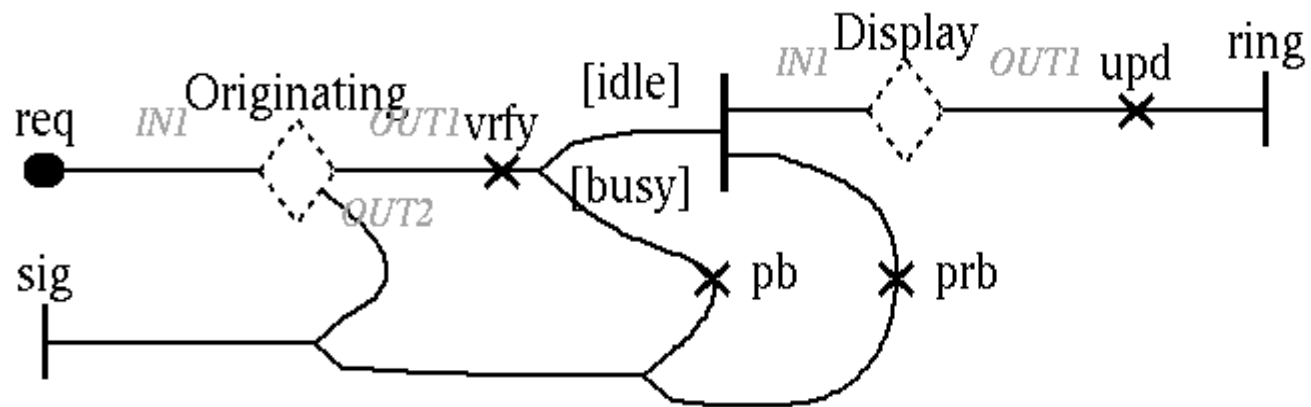
Map Title Basic Call map

0 root

Navigation Mode Full UCM Design Navigation

Component Path Select Scale 208% Editing Mode Full Editing

Decomposition Level



Responsibilities

- pb**
Prepare busy reply.
- prb**
Prepare ringack signal.
- upd**
Update system status.

Edit Responsibility

Empty text area for editing responsibility details.

Add Edit Delete

Empty text area for editing responsibility details.

Add Edit Delete

Description of Map

Simplified call connection scenario.

Example UCM for FIREworks workshop.



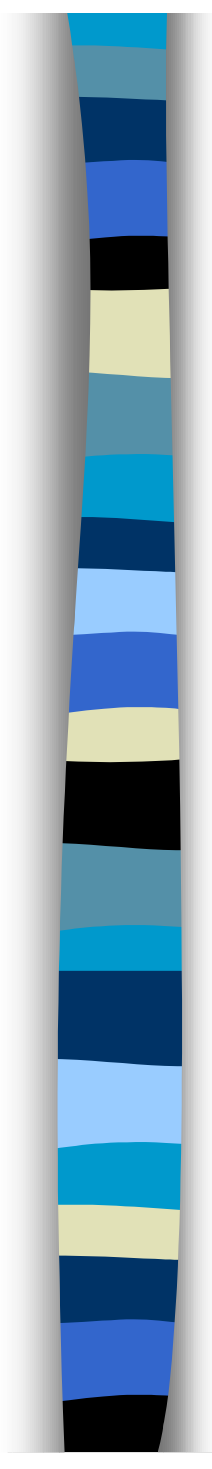
Current UCM Research

- Link UCMs to other languages/notations
 - (H)MSCs or UML sequence diagrams (currently being implemented in UCMNav)
 - Hierarchical state machines (UML, ROOM)
 - LOTOS, for validation and FI detection
 - SDL, for validation
 - LQNs, for performance analysis
 - TTCN, for functional testing



Conclusions on UCMs

- Enthusiastic community of users
 - UCM User Group, UCM Advisory Board
- Success stories in reactive/telecom systems
 - Senior designers can communicate their knowledge, junior designers take care of the specifics
- Satisfy many requirements of URN
- Attractive level of abstraction, concepts and constructs for a full-fledge feature description language
- Have a place in current approaches



How can YOU contribute to Use Case Maps?

- Still lack of formal (dynamic) semantics
- Integration of UCM view with current design notations, languages and methodologies
- Improve automatisisation
- Application of UCMs (features and others)
- Join UCM User Group



Use Case Maps Web Page

- <http://www.UseCaseMaps.org/>
- Prime source of information about the world of UCMs
- Supports the UCM User Group
- UCM Virtual Library (publications)
- Tools (UCM Navigator and others)
- XML Document Type Definition
- UCMs and UML