

From Use Case Maps to SDL  
(Specification and Description  
Language)

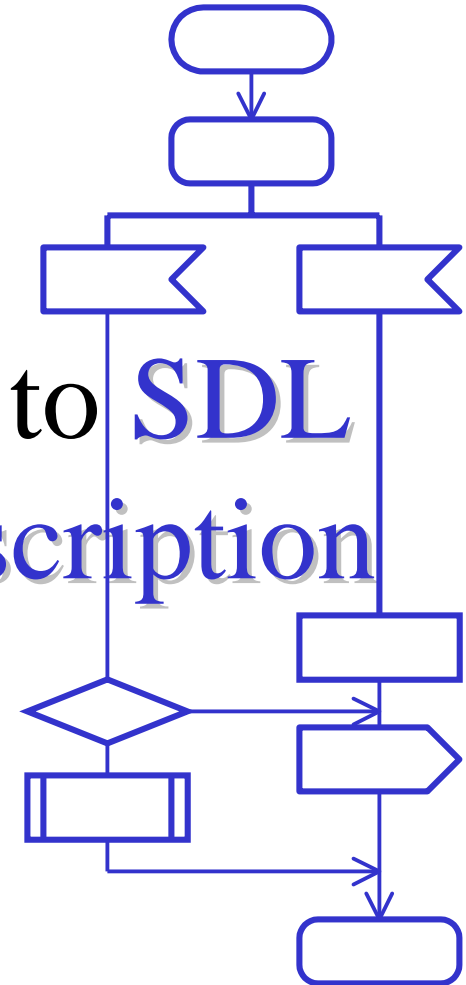
Igor Sales

S.I.T.E.

School of Information Technology and Engineering

University of Ottawa

[isales@site.uottawa.ca](mailto:isales@site.uottawa.ca)



# Outline

- 
- × Motivation
- × Introduction
- × Use Case Maps (UCM)
- × Specification and Description Language (SDL)
- × The Bridging Procedure
- × Conclusions

# Motivation

## ✗ Telecom and Internet Services/Products

Easy to Create



Hard to Implement, Document, and Validate

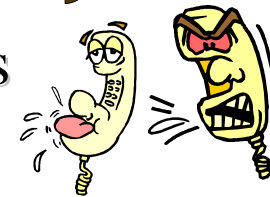


## ✗ Users

Buy & Love the expected behaviour



Hate & Trash unexpected behaviours



## ✗ Companies

Spend Time+Money to Develop Services/Products

Competition Requires Time to Market

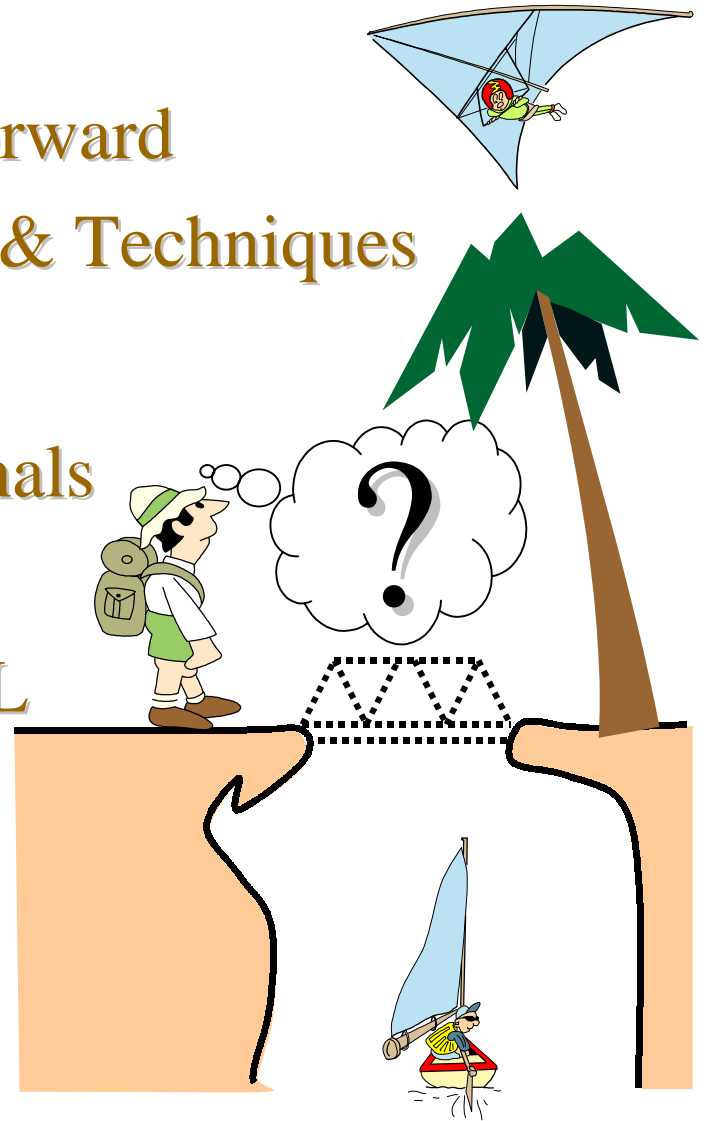
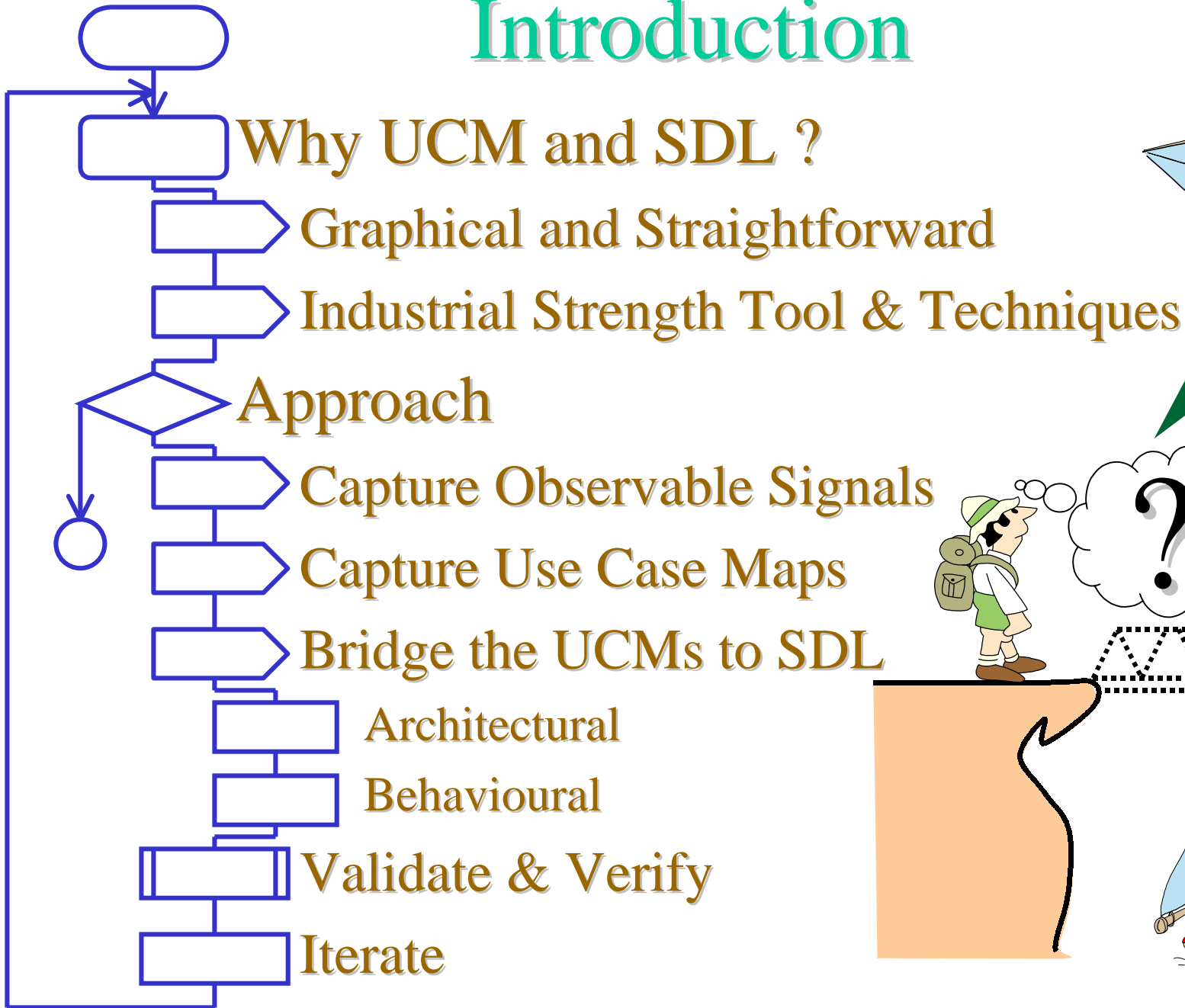


## ✗ **THUS**

UCM Fast High Level Prototyping

SDL Fast Validation/Verification/Simulation

# Introduction



# Use Case Maps (UCM)



Start Point

Thread

✗ Responsibility

◇ Stub

— End Point

Entity Boundary

◇ Static Stub

◇ Dynamic Stub

Triggering event

Time

Action

Different Level UCM

Reaction

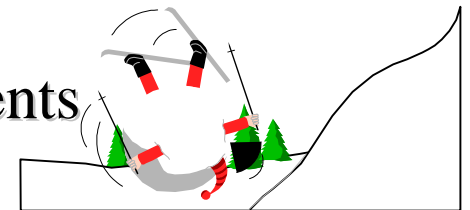
Architecture (Optional)

1 Sub-UCM

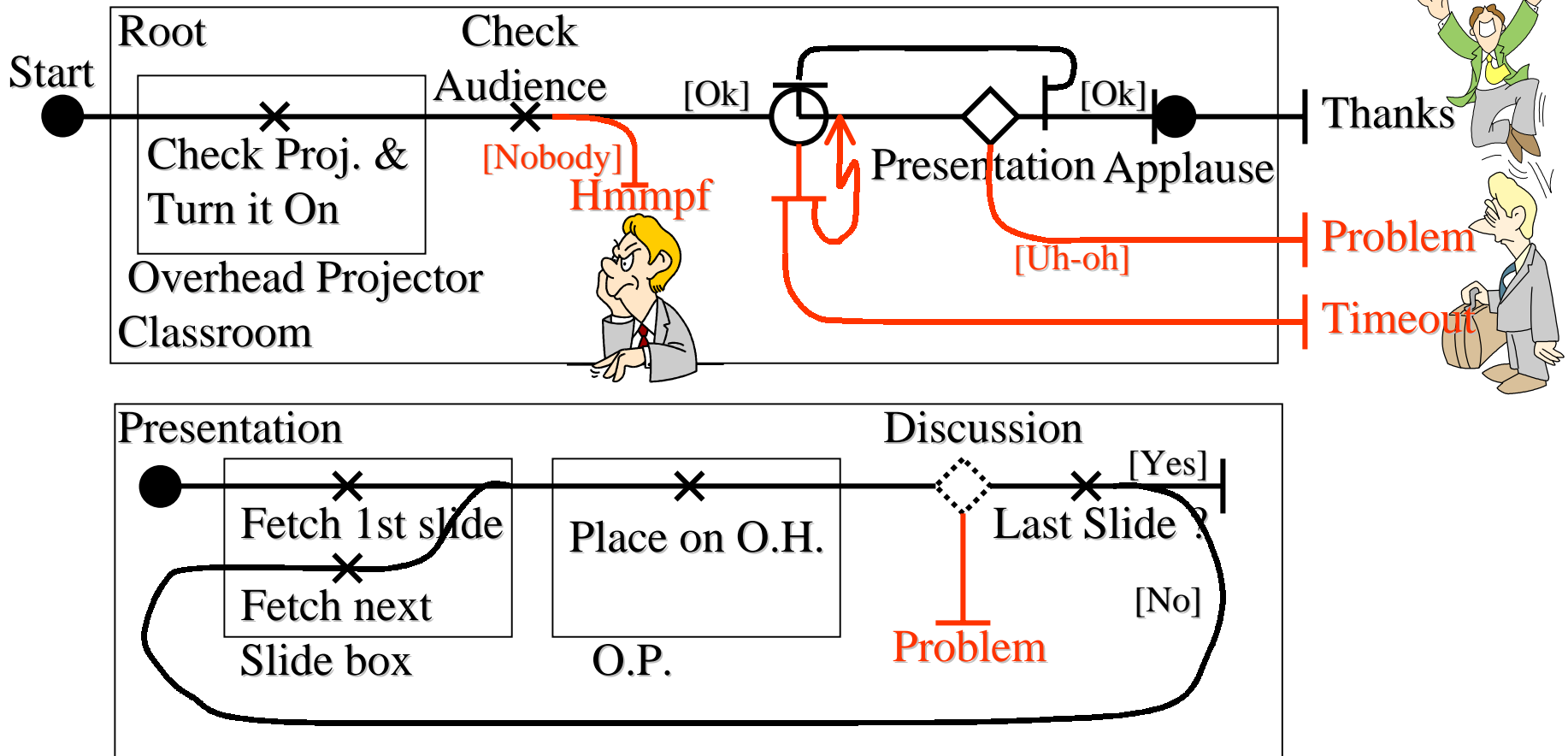
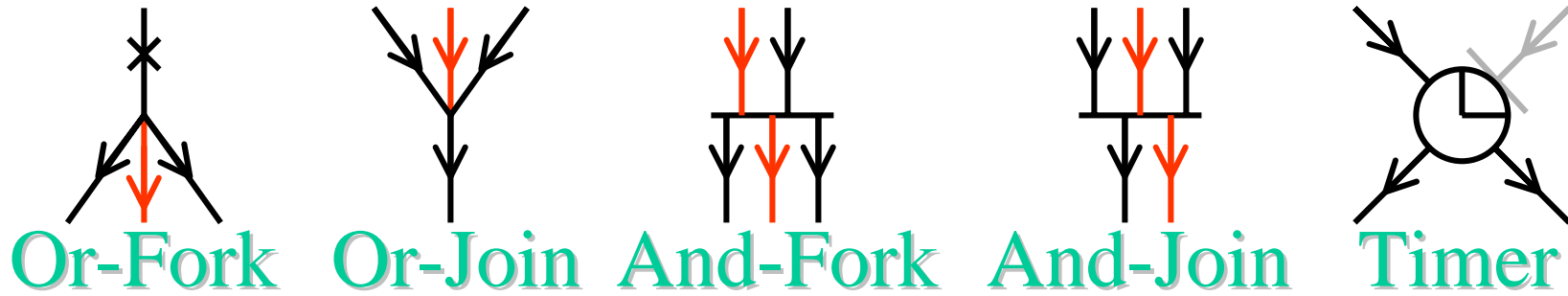
Choose One!



Describe the System Causal Flow of Events

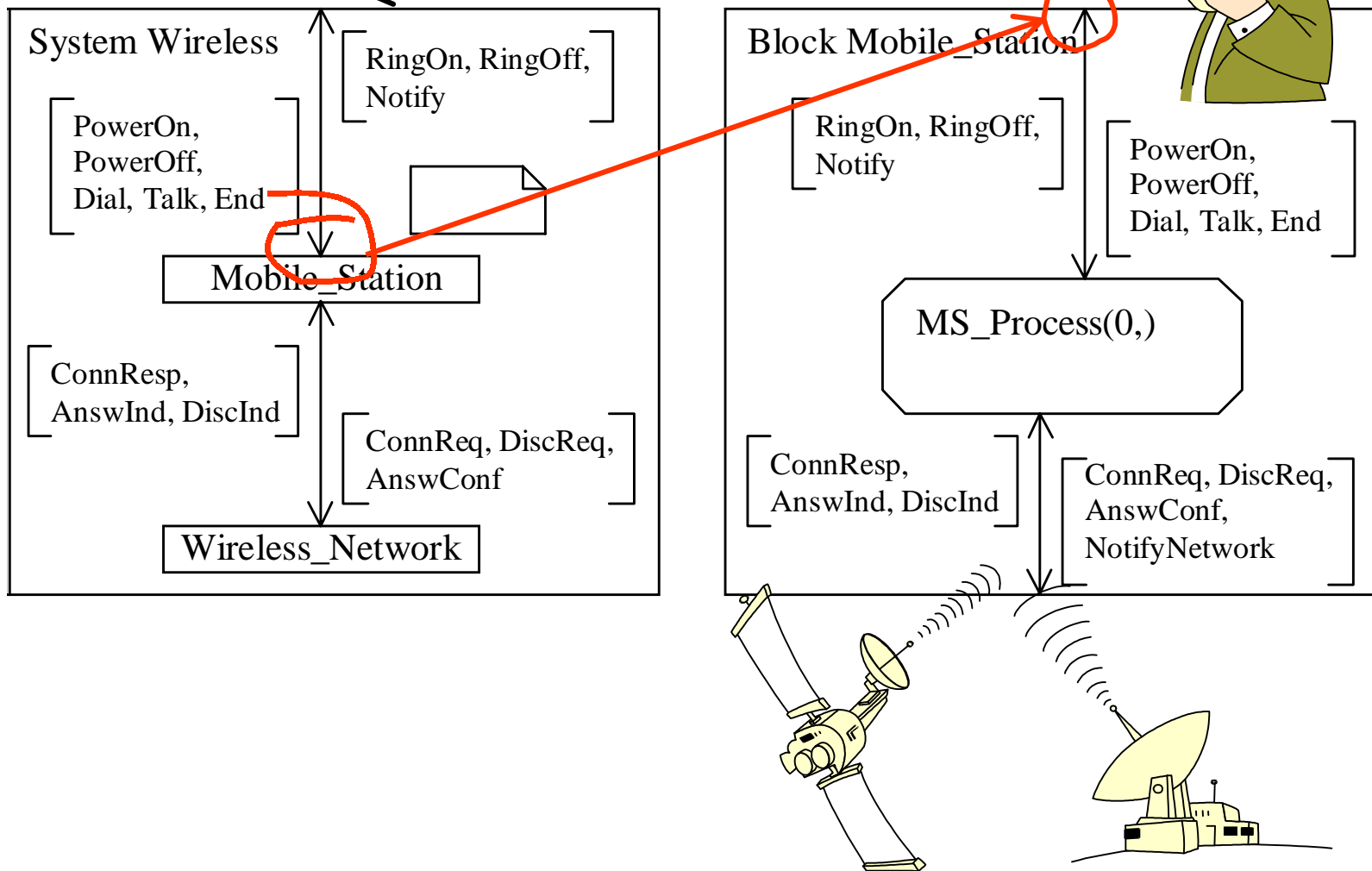


# Use Case Maps (UCM)










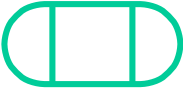





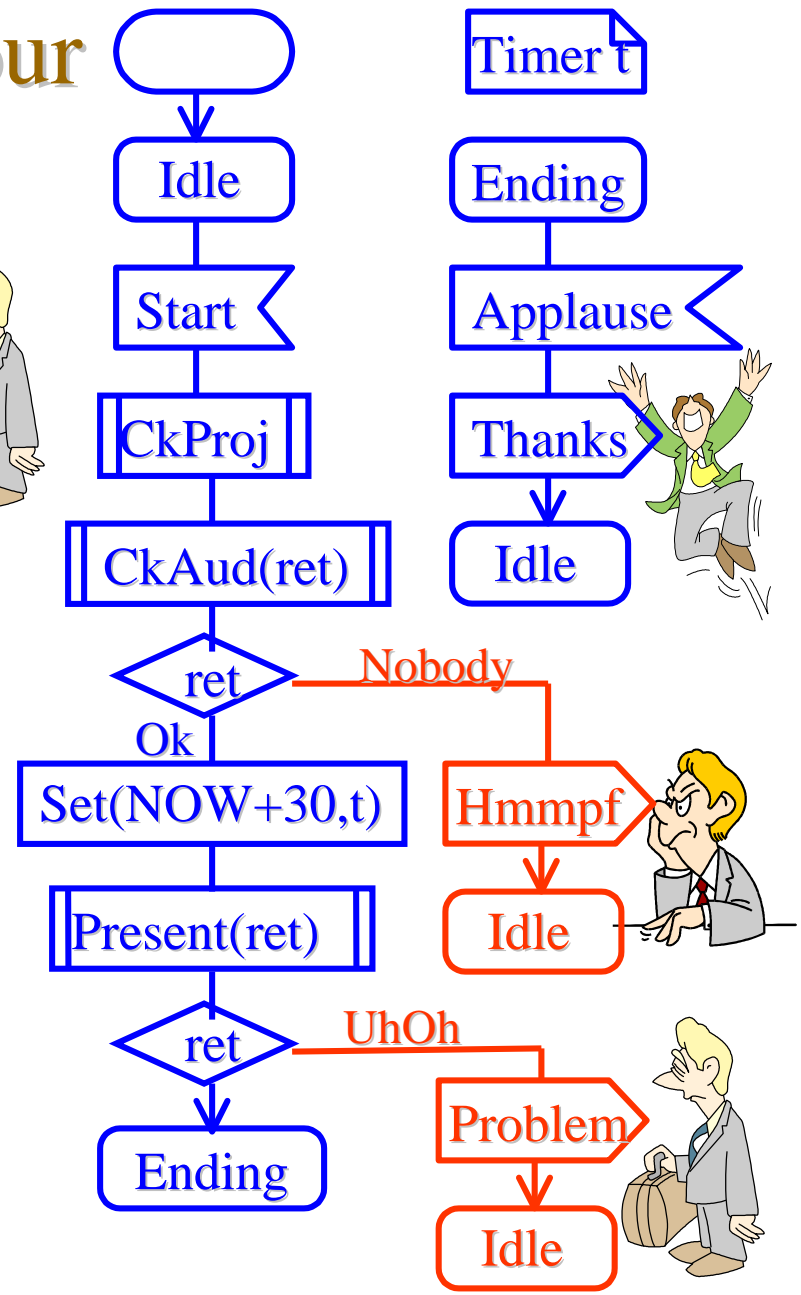
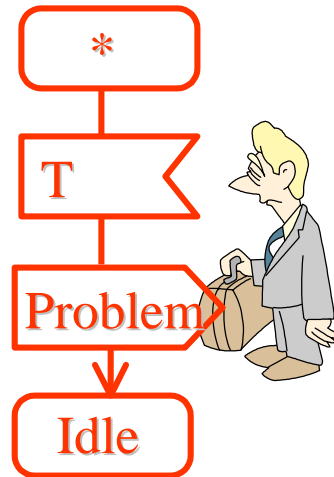
# Specification and Description Language (SDL)

- Architecture



# SDL Behaviour

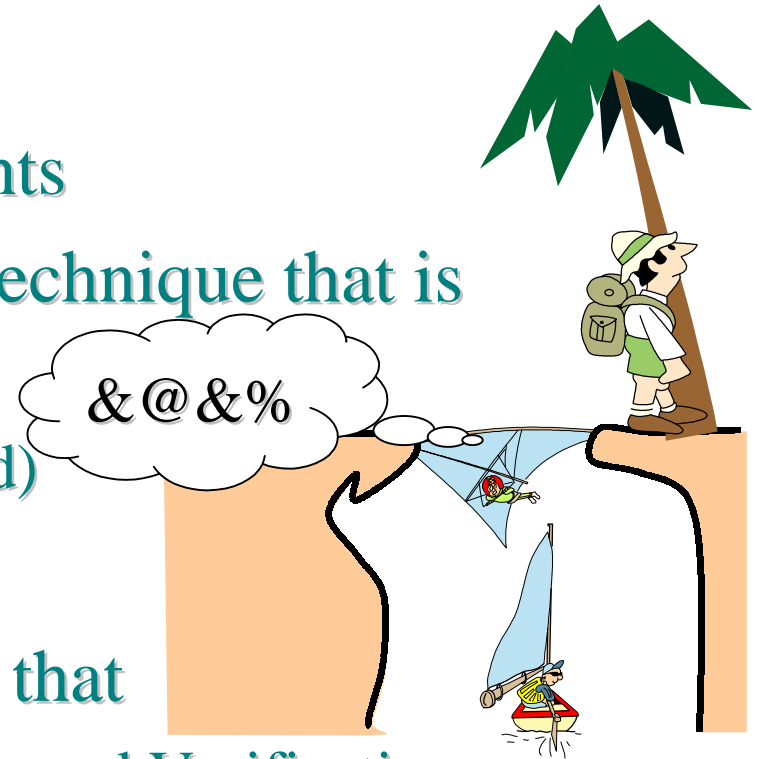
-  Start
-  State
-  Input
-  Output
-  Decision
-  Task
-  Procedure Call
-  Create Request
-  End Process
-  Procedure Start
-  End Procedure
-  Declaration
-  Connection Point





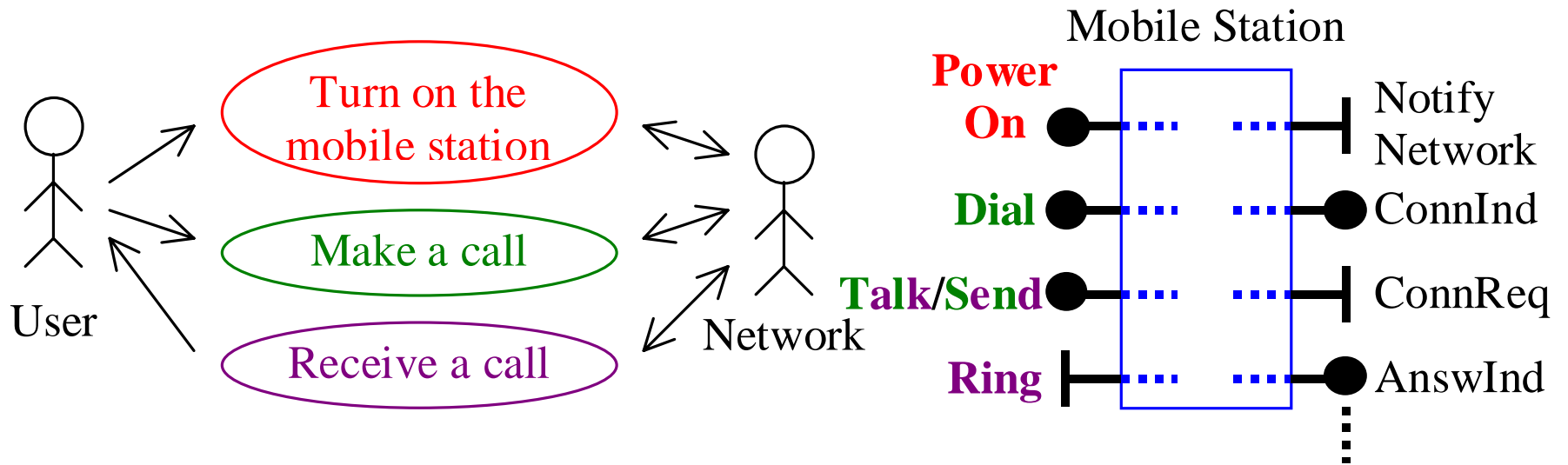
# The Bridging Procedure

- Approach
  - Capture Observable Messages
  - Capture UCM behaviour from Requirements
  - Apply Mapping to construct an SDL model
- In General Terms
  - Capture User Observable Events
  - Model the Abstract Model a Technique that is
    - Graphical (Straightforward)
    - Abstract (Little detail is required)
  - Mix these 2 together
  - Generate a State Based Model that
    - Has Tools that Allow Simulation and Verification
    - Is Straightforward (Little Natural Language Required)

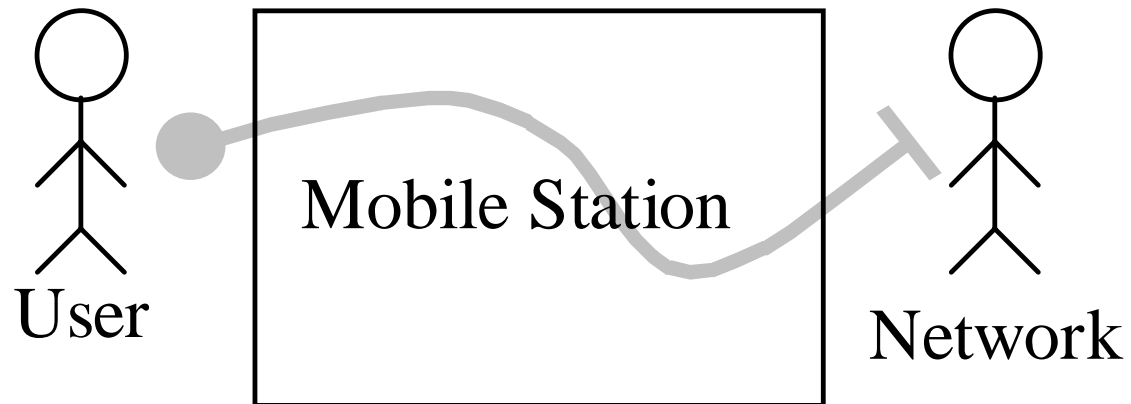


# The Bridging Procedure

## Behaviour from Requirements

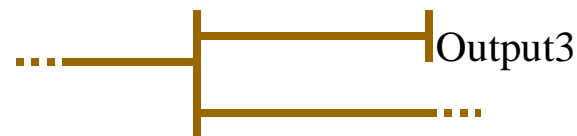
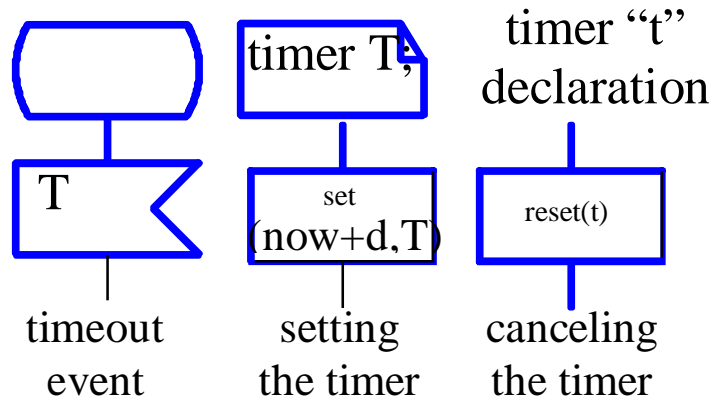
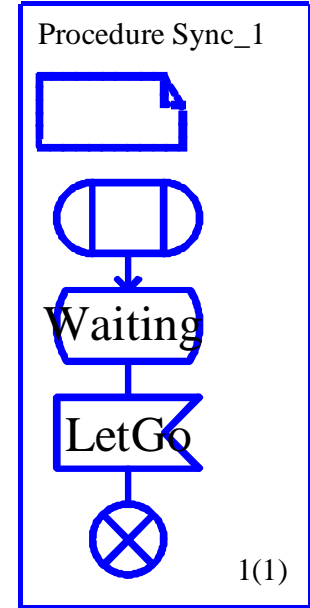
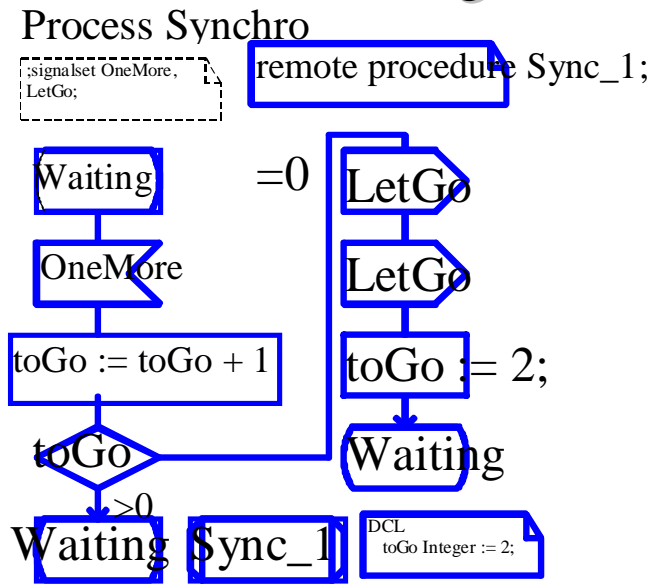
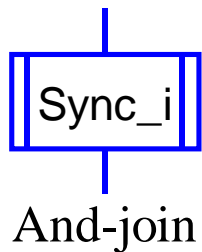
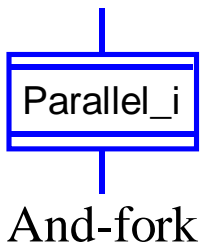
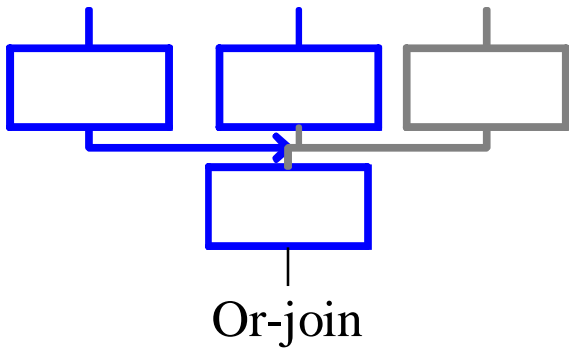
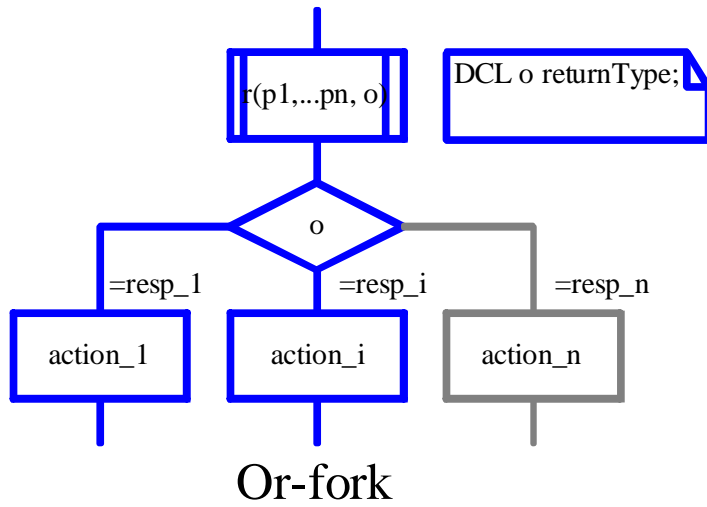


## Structure From Requirements



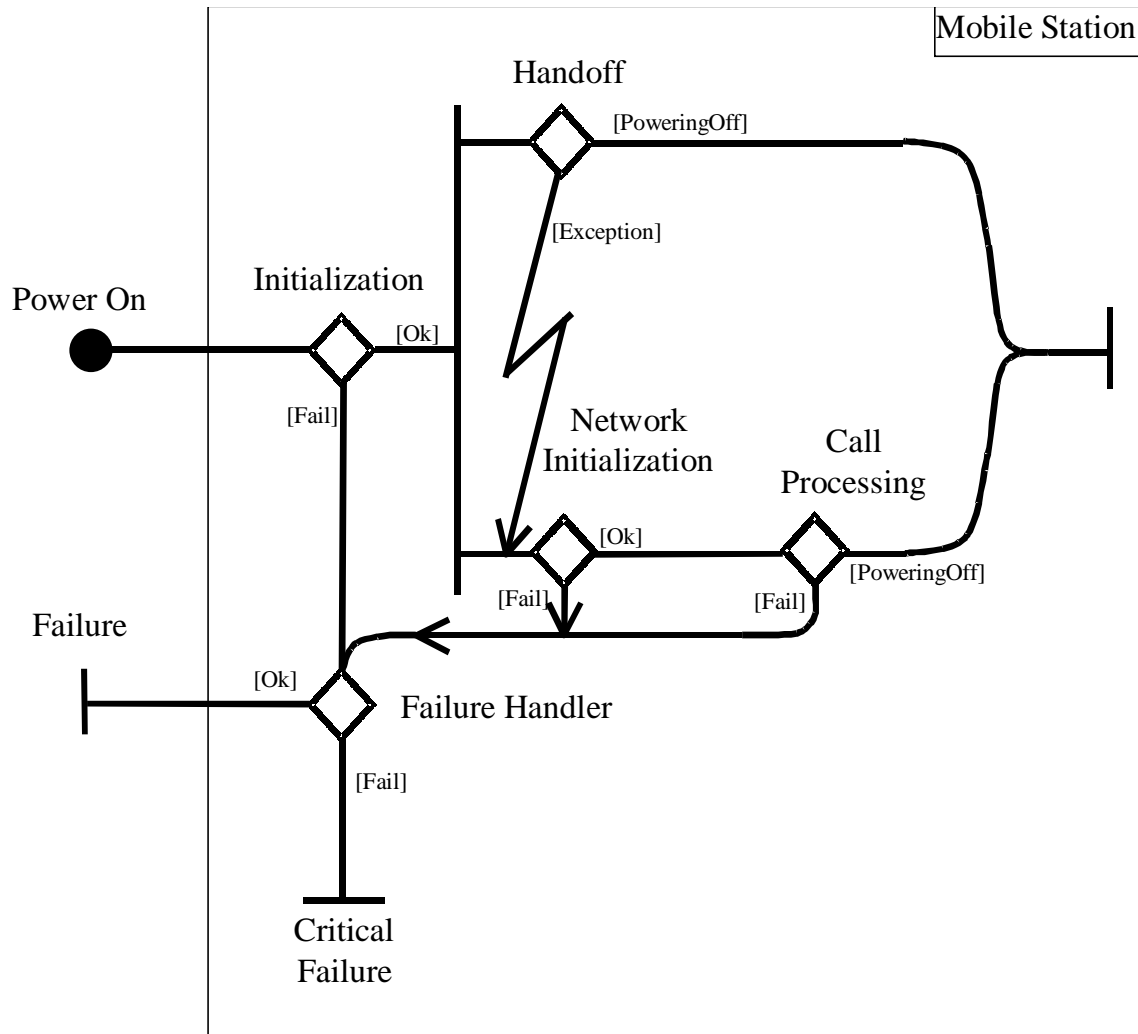
# The Bridging Procedure

## SDL General Bridge Rule from UCMs



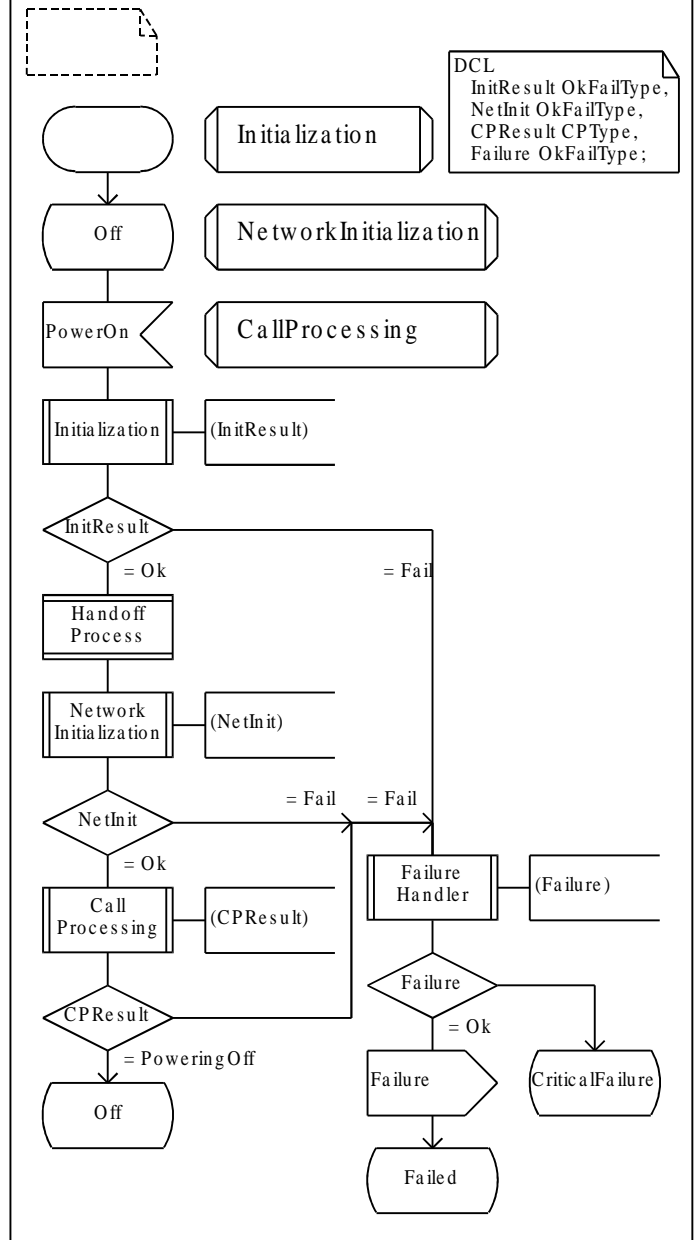
# The Bridging Procedure

An example

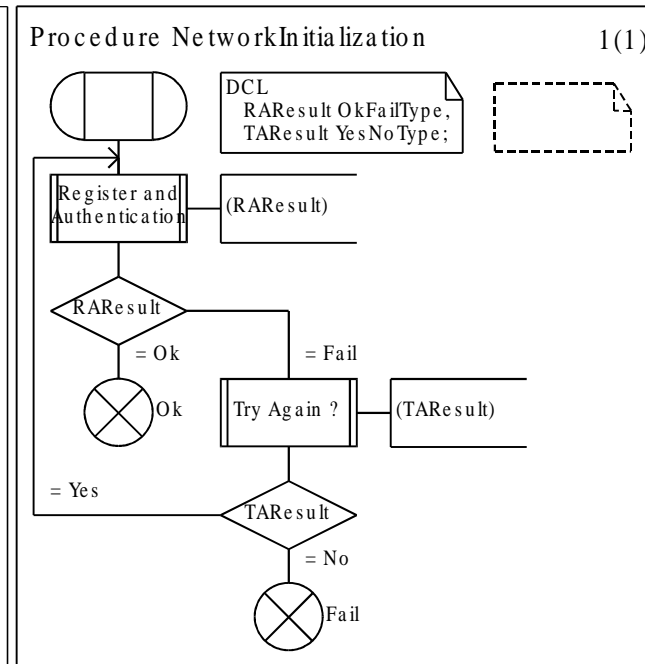
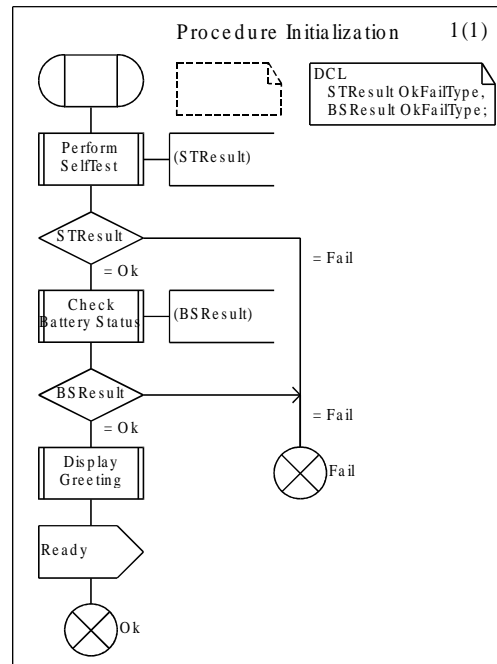
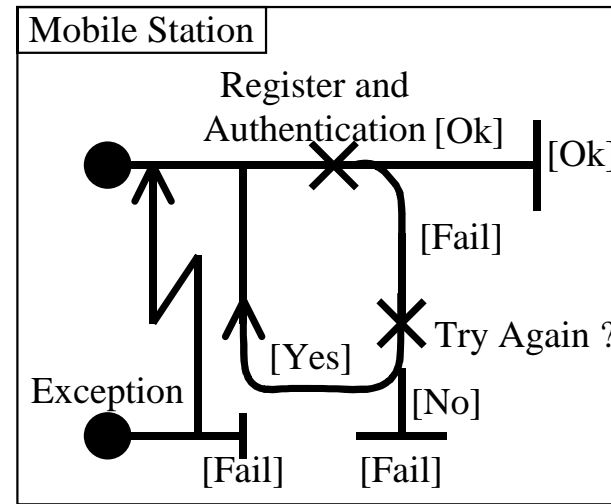
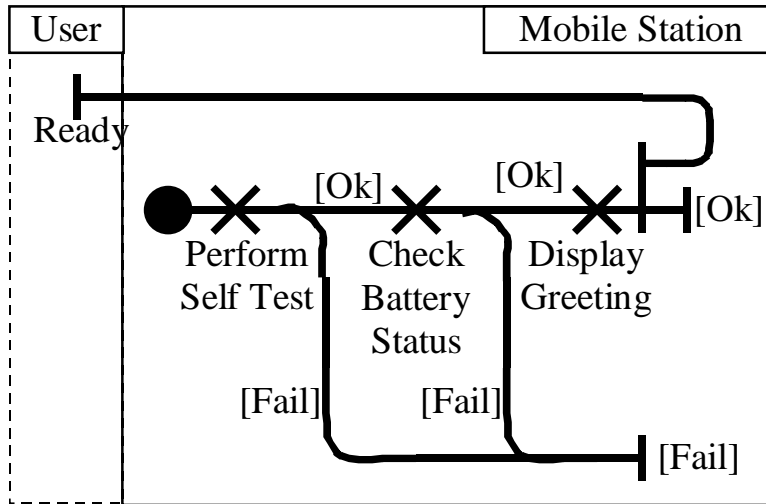


Process MobileStation\_Process

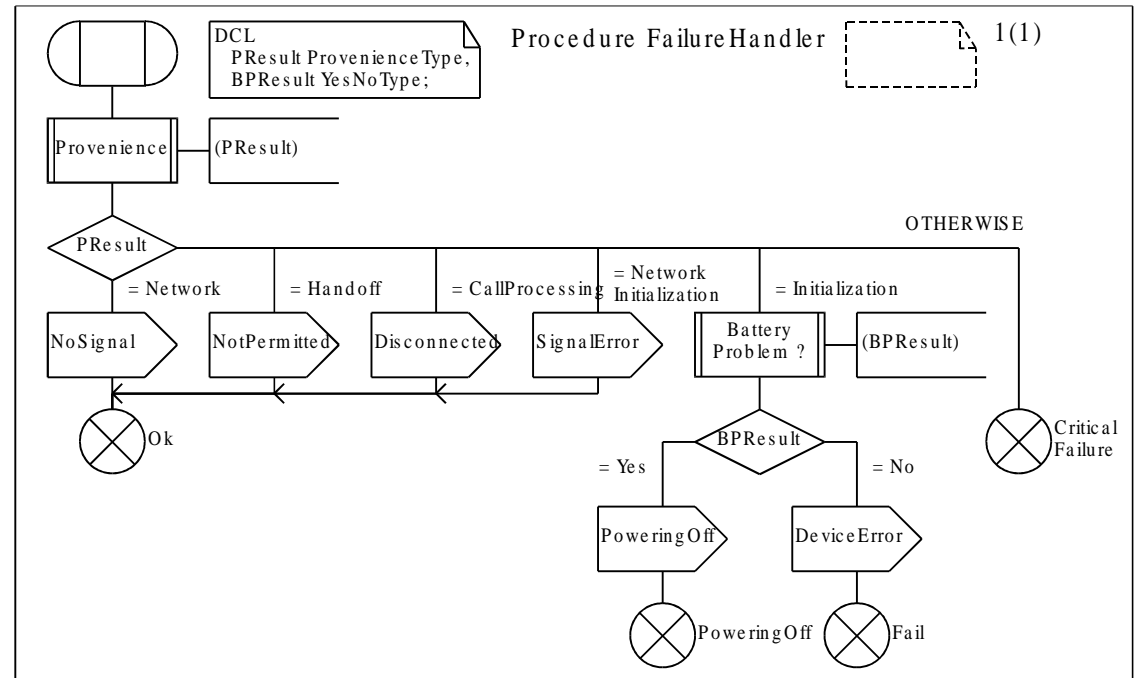
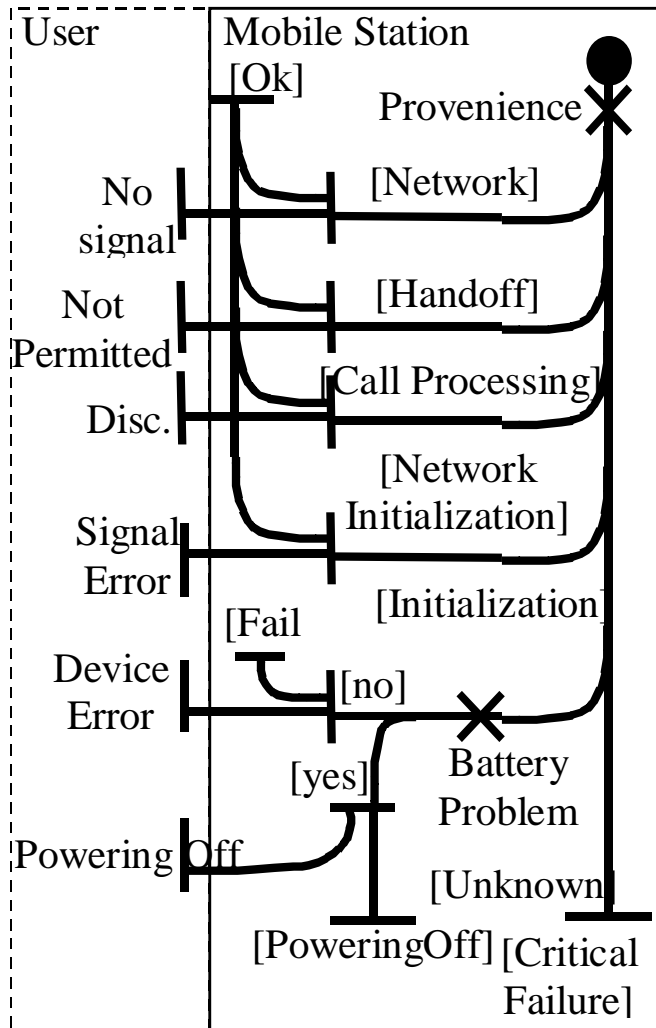
1(1)



# The Bridging Procedure An example (cont'd)

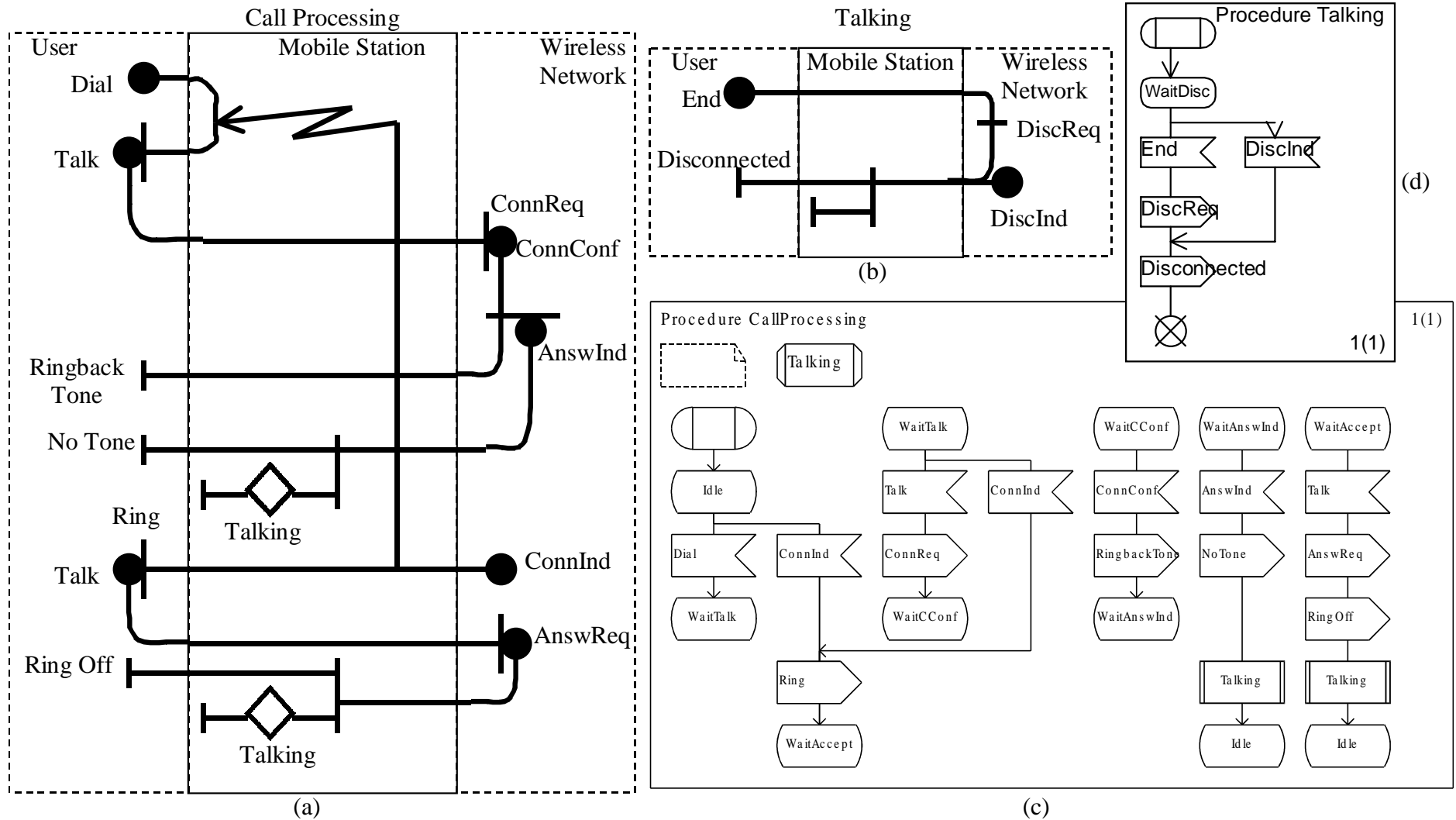


# The Bridging Procedure An example (cont'd)



# The Bridging Procedure

An example (cont'd)



# Conclusions

- EASY

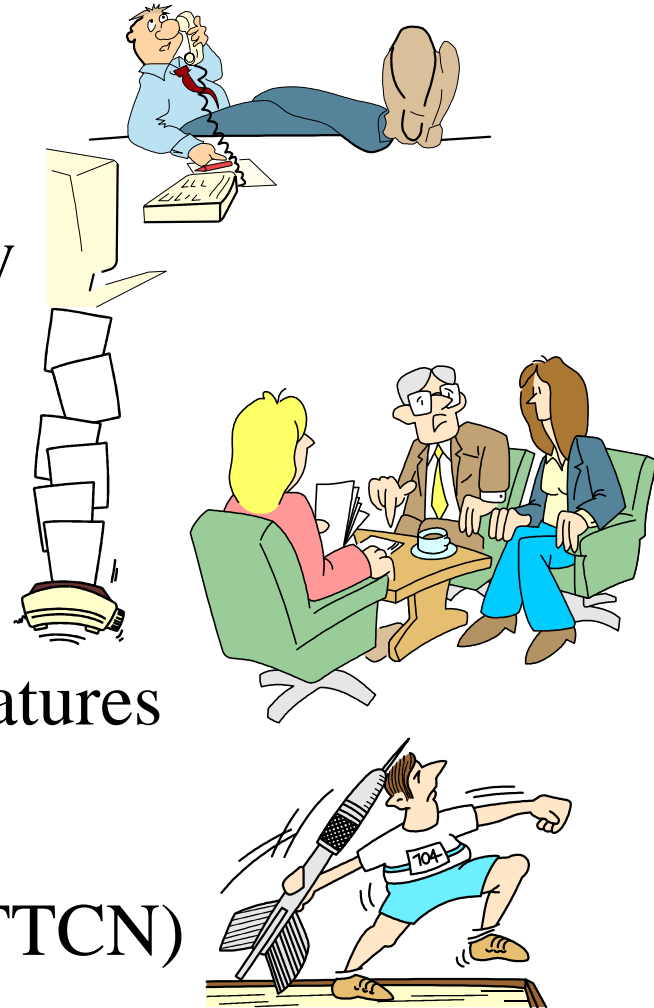
- To Construct UCMs
- To Understand & to Apply

- QUICK

- Prototyping
- Validation & Verification
- Implementation of new features

- Further Proposed Work

- Generation of test cases (TTCN)
- Semi-Automation of Bridging Procedure
- Use Cases  $\Leftrightarrow$  Behaviour Model  $\Leftrightarrow$  State Model
- Multi-Level UCMs





# Main References

- Use Case Maps web page - [www.UseCaseMaps.org](http://www.UseCaseMaps.org)
- ITU Z.100 - Specification and Description Language (SDL)
- ITU Z.120 - Message Sequence Charts (MSC)
- SDL Forum - [www.sdl-forum.org](http://www.sdl-forum.org)
- Telelogic TAU - [www.telelogic.com](http://www.telelogic.com)
- Verilog - [www.verilogusa.com](http://www.verilogusa.com)

# Acknowledgments

- University of Ottawa
  - School of Information Technology and Engineering

- Nortel Networks

- Mitel Corporation

Last but not least!

- SBRC'2000

- UFMG

- Departamento de Ciência da Computação (DCC)