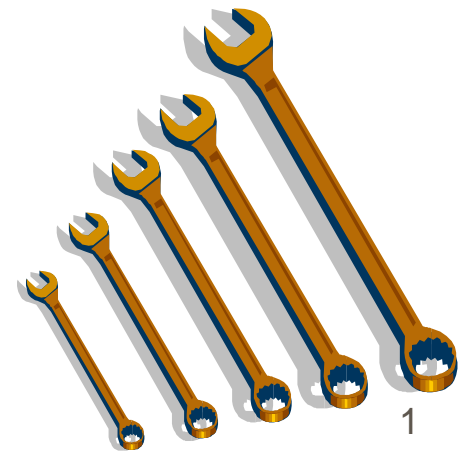


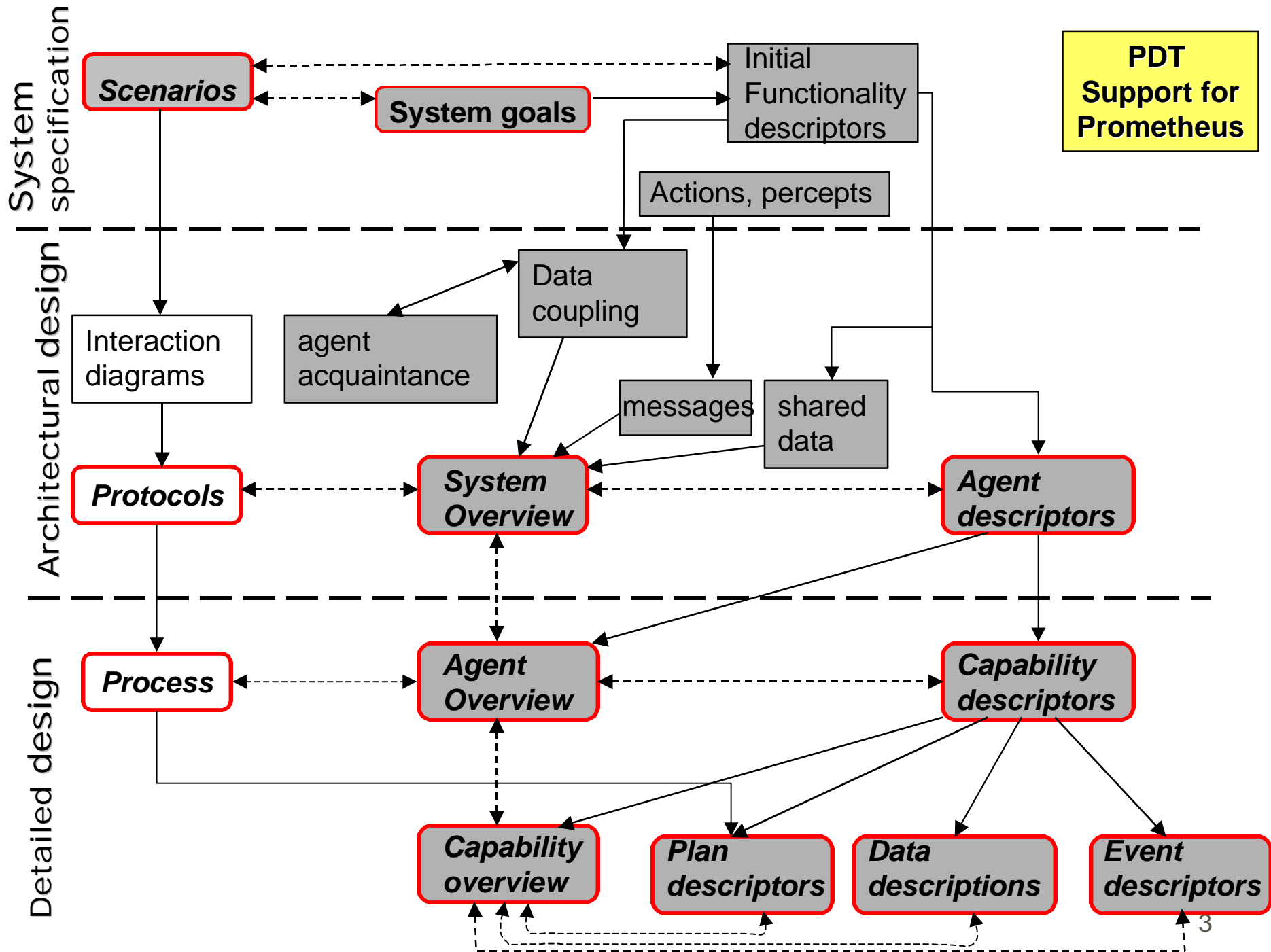
Tool Support for Prometheus

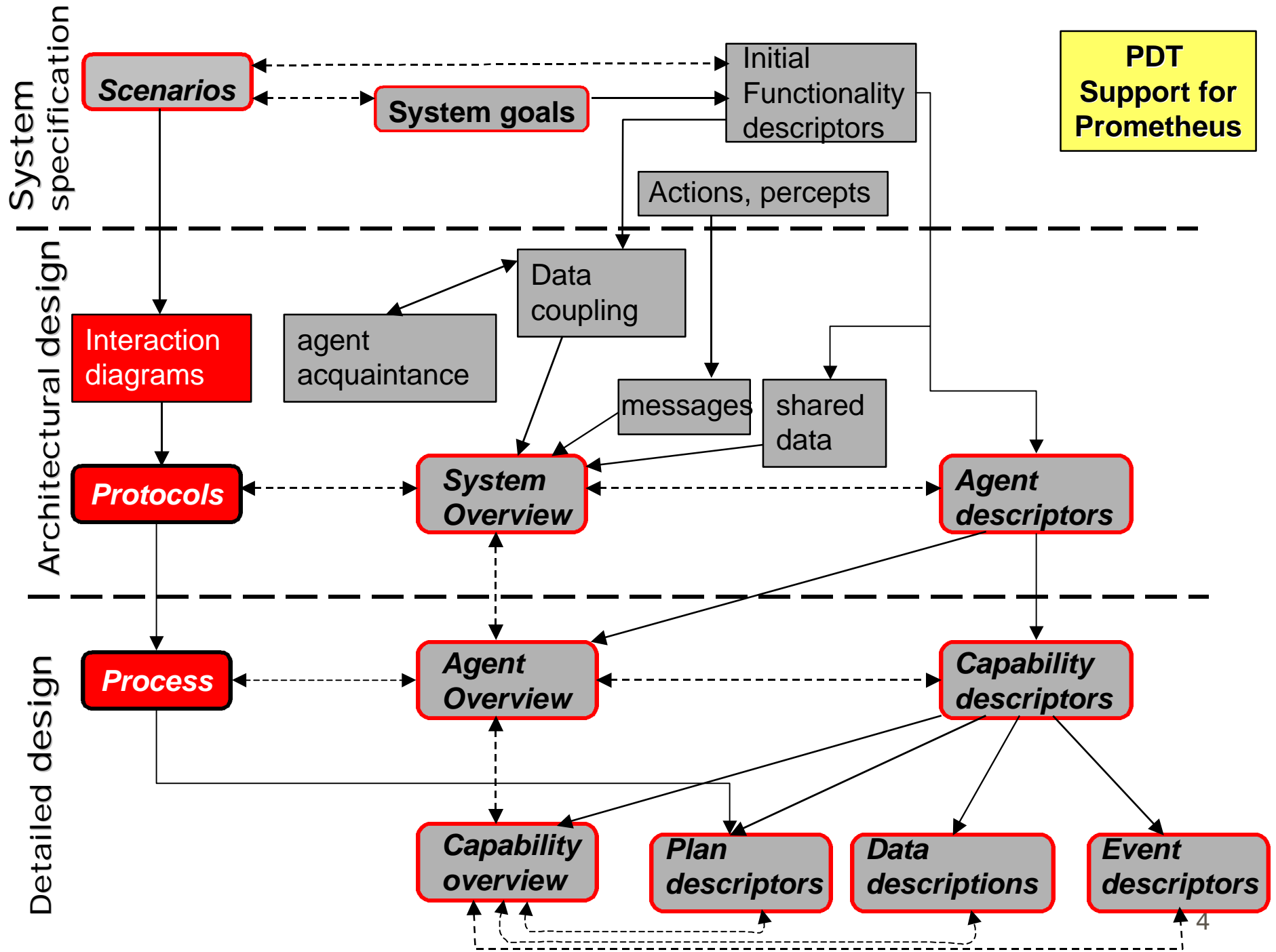
- Jack Development Environment: Design Tool
Developed by Agent Oriented Software
- Prometheus Design Tool (PDT)
 - Developed first half of 2003 by Anna Edberg & Christian Andersson at RMIT
 - Later extended by Claire Hennekam and Jason Khallouf.
- AUML tool
 - Prototype by Michael Winikoff

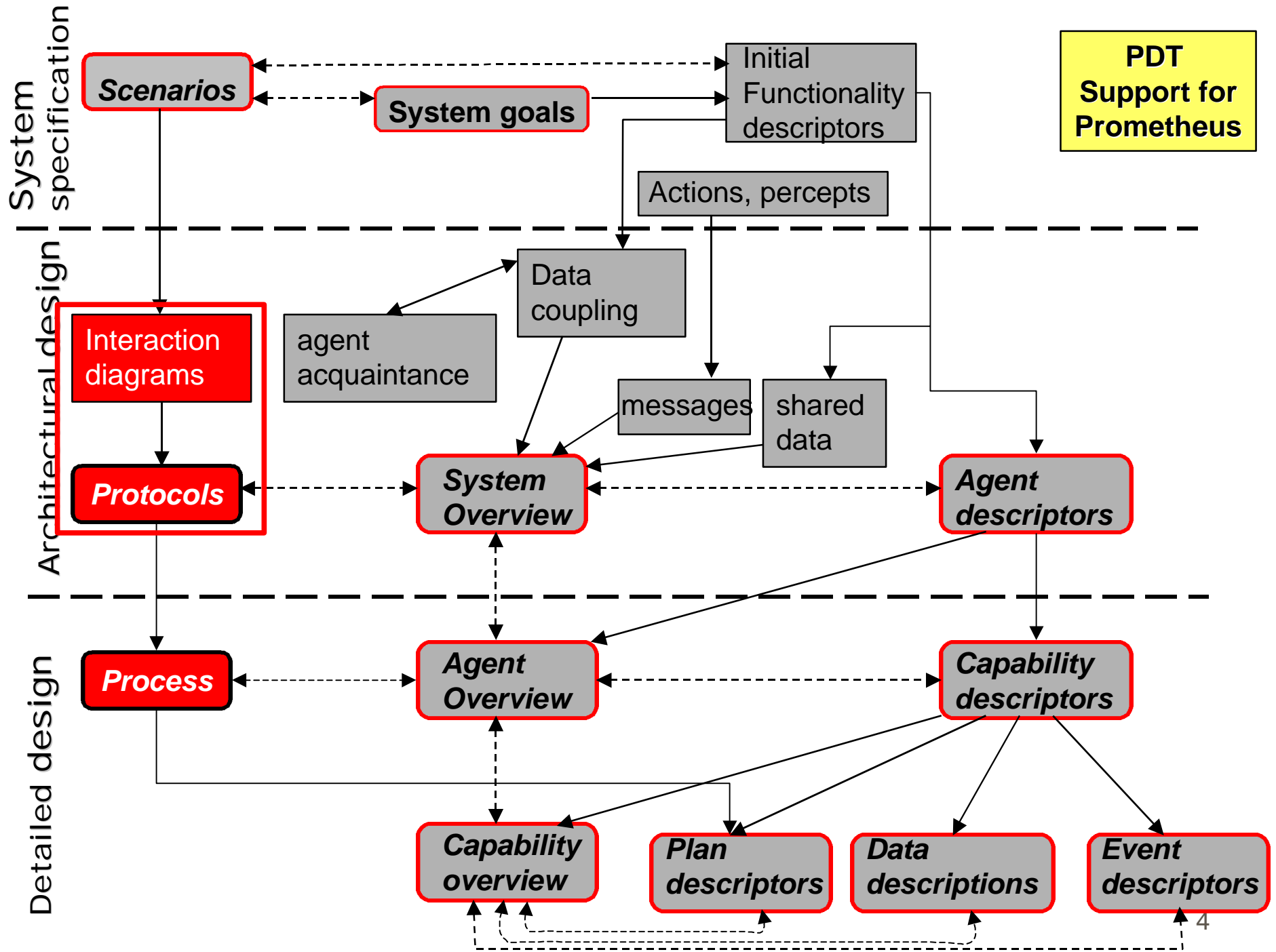


Prometheus Design Tool

- Design tool supporting the Prometheus methodology developed at RMIT
- Features:
 - Enter and edit designs (descriptor forms, diagrams)
 - Cross checking:
 - undefined and unused entities
 - interface consistency
 - ...
 - Generate design report (HTML) + images (PNG)
- Written in Java, freely available on web
www.cs.rmit.edu.au/agents/pdt
- Beta – feedback welcome!
- Demo ...

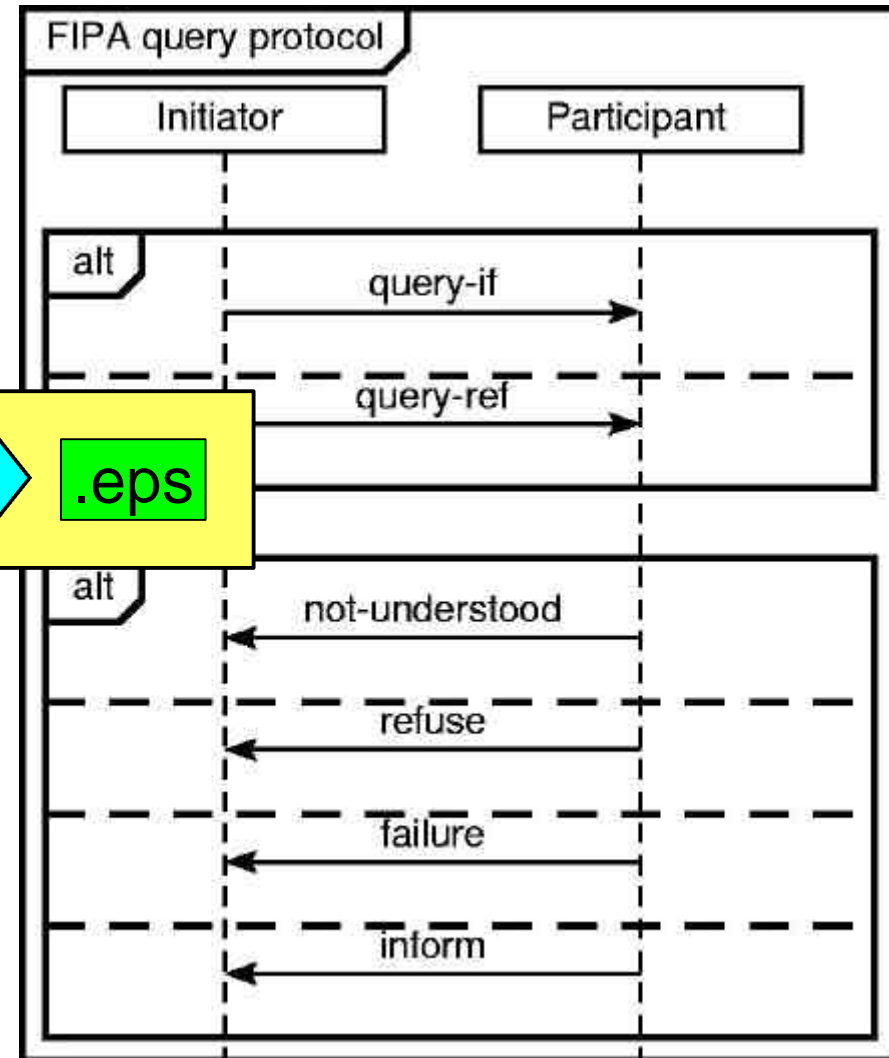
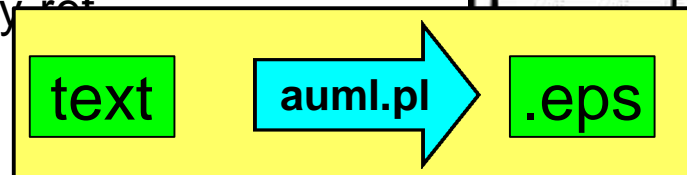






AUML-2 Prototype Tool

```
start FIPA query protocol
agent I Initiator
agent P Participant
box alt
  message I P query-if
next
  message I P query-ref
end alt
box alt
  message P I not-understood
next
  message P I refuse
next
  message P I failure
next
  message P I inform
end alt
finish
```



Additional Slides

(including screenshots of PDT, for those reading the slides after the demo/presentation)

Diagrams

Calendar Agent

▼ System Specification

Goal Overview

Functionalities

Scenarios

System Interface

▼ Architectural Design

Data Coupling

Agent Acquaintance

System Overview

▼ Detailed Design

▶ GUI

Messaging

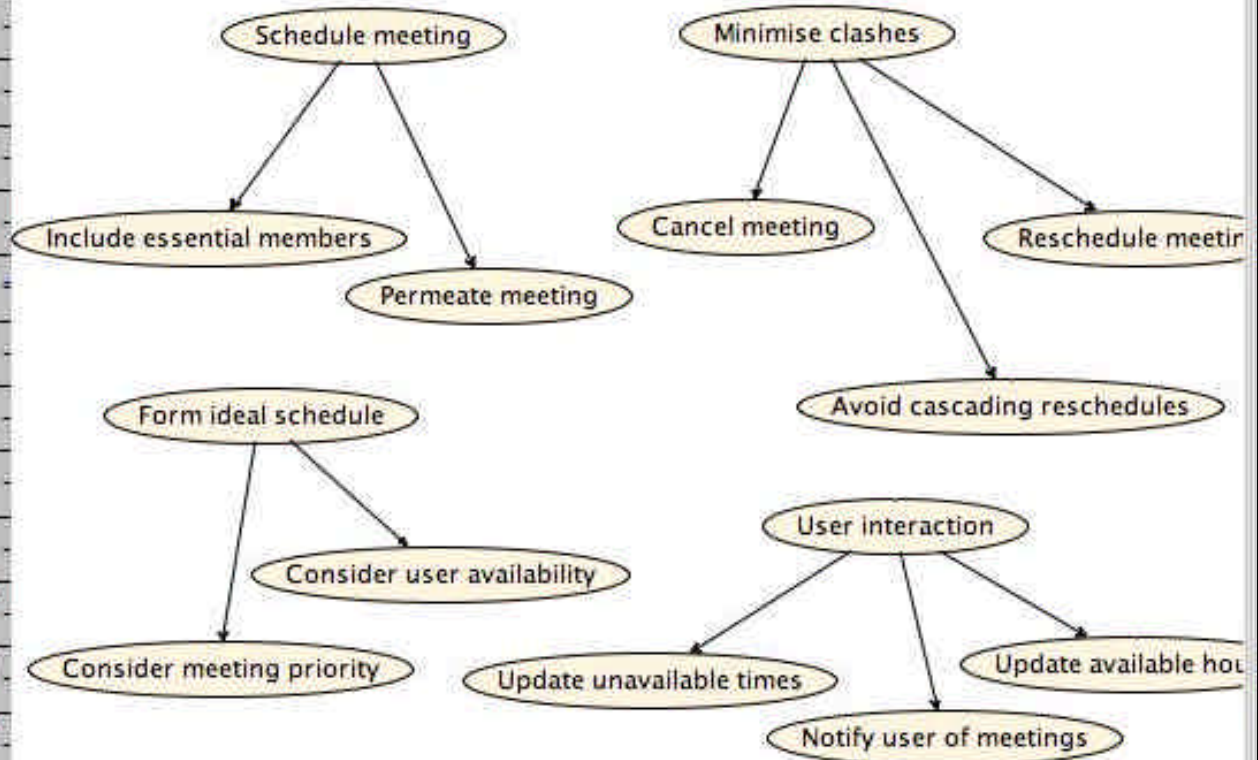
Meeting Scheduler

Entities

System entities Filter

- InformUser
- Meeting accepted
- Meeting cancelled
- Meeting declined
- Request User Intervene
- 👤 GUI
- 👤 Meeting Scheduler

Goal Overview Diagram



Diagrams

Calendar Agent

System Specification

Goal Overview

Functionalities

Scenarios

System Interface

Architectural Design

Data Coupling

Agent Acquaintance

System Overview

Detailed Design

GUI

Messaging

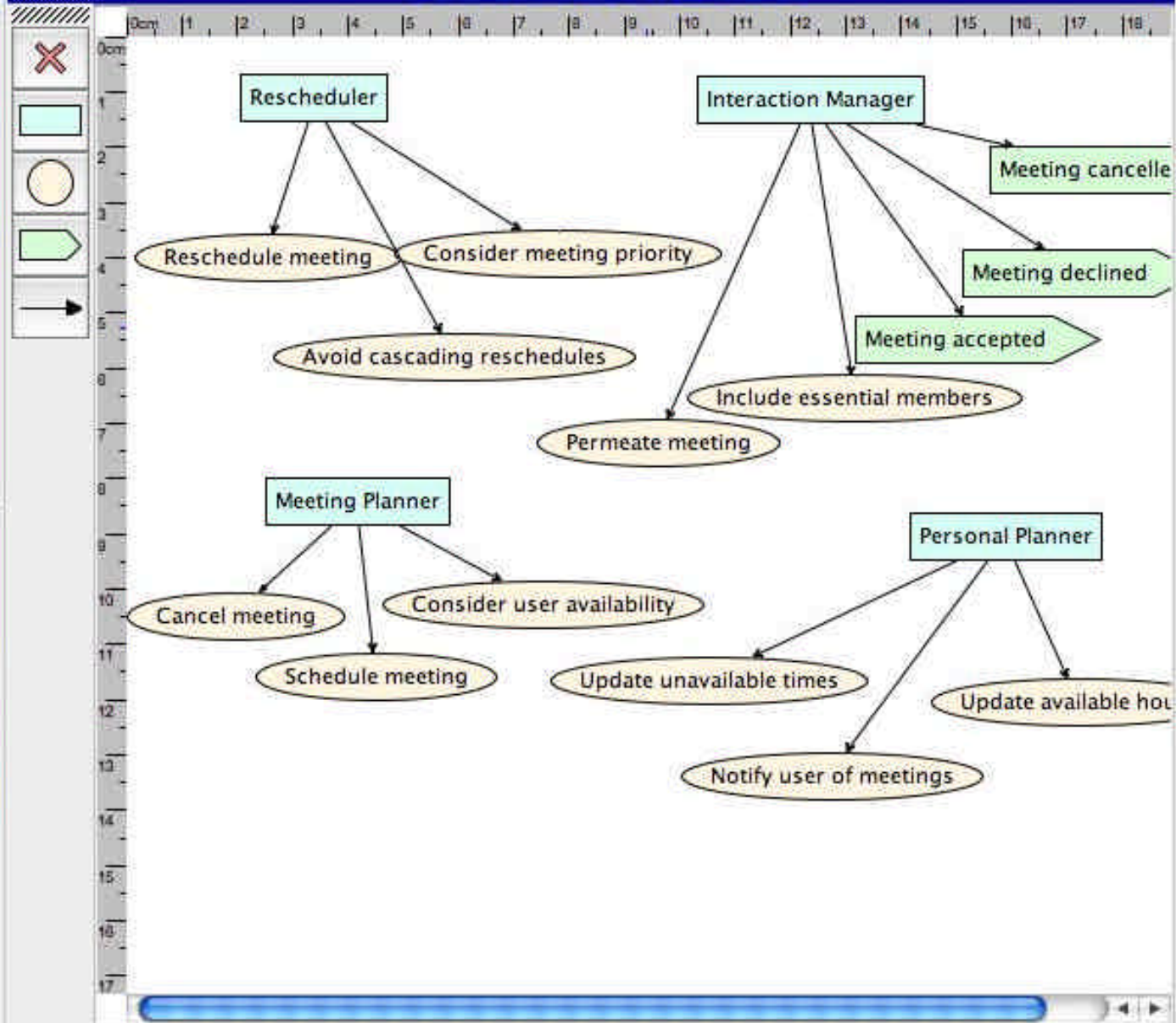
Meeting Scheduler

Entities

System entities Filter

- InformUser
- Meeting accepted
- Meeting cancelled
- Meeting declined
- Request User Interv
- GUI
- Meeting Scheduler

Functionalities Diagram



Diagrams

Calendar Agent

System Specification

- Goal Overview
- Functionalities
- Scenarios
- System Interface

Architectural Design

- Data Coupling**
- Agent Acquaintance
- System Overview

Detailed Design

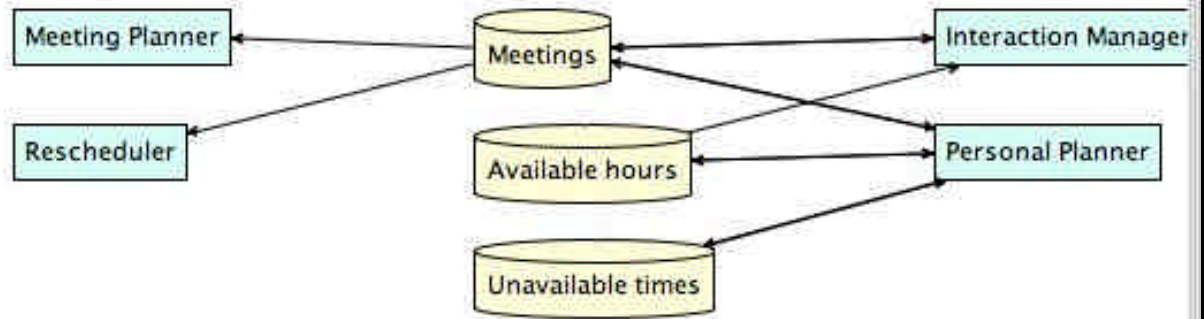
- GUI
- Messaging
- Meeting Scheduler

Entities

System entities Filter

- InformUser
- Meeting accepted
- Meeting cancelled
- Meeting declined
- Request User Interven
- GUI
- Meeting Scheduler

Data Coupling Diagram



Diagrams

Calendar Agent




▼ System Specification

- Goal Overview
- Functionalities
- Scenarios
- System Interface

▼ Architectural Design

- Data Coupling
- Agent Acquaintance
- System Overview









▼ Detailed Design

- ▶  GUI
-  Messaging
-  Meeting Scheduler

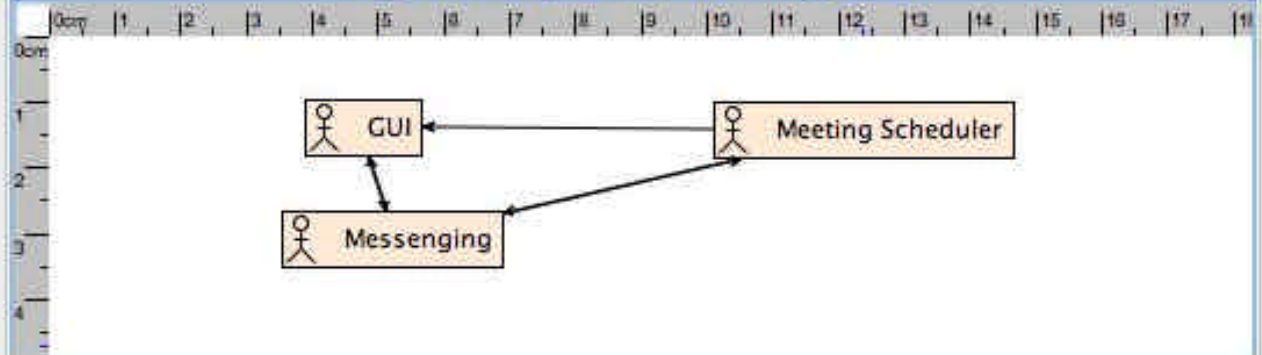
Entities

System entities

Filter

-  Meeting accepted
-  Meeting cancelled
-  Meeting declined
-  Request User Intervention
-  GUI
-  Meeting Scheduler
-  Messaging
-  manageAvailability

Agent Acquaintance Diagram



GUI - Descriptor

Name: GUI

Description:

Interacts with the user and stores user-related data

Cardinality minimum: 1

Cardinality maximum: 1

Lifetime:

System.

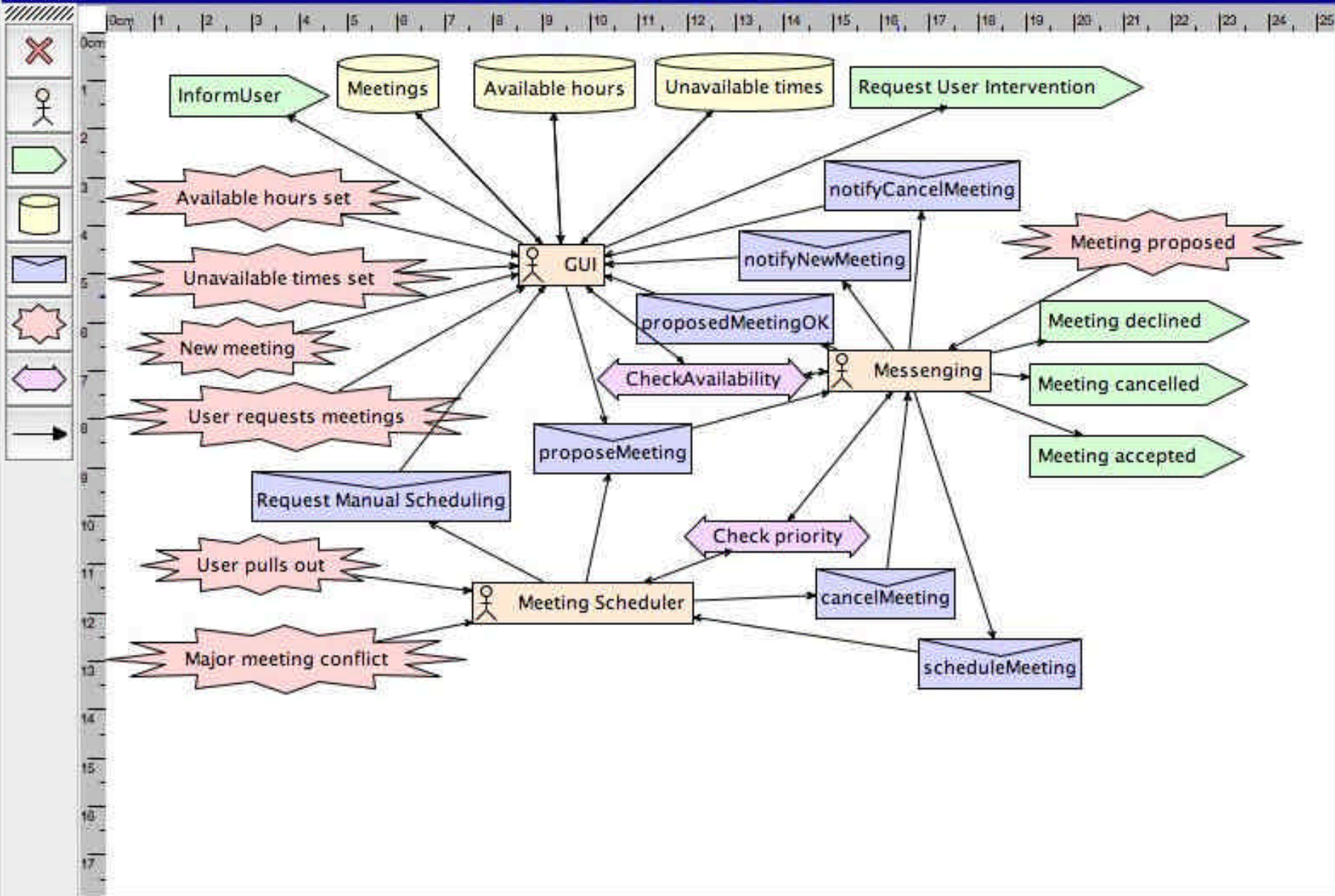
Initialisation:

Demise:

Delete

Set

System Overview Diagram



Diagrams

Calendar Agent

▼ System Specification

- Goal Overview
- Functionalities
- Scenarios
- System Interface

▼ Architectural Design

- Data Coupling
- Agent Acquaintance
- System Overview

▼ Detailed Design

▼ GUI

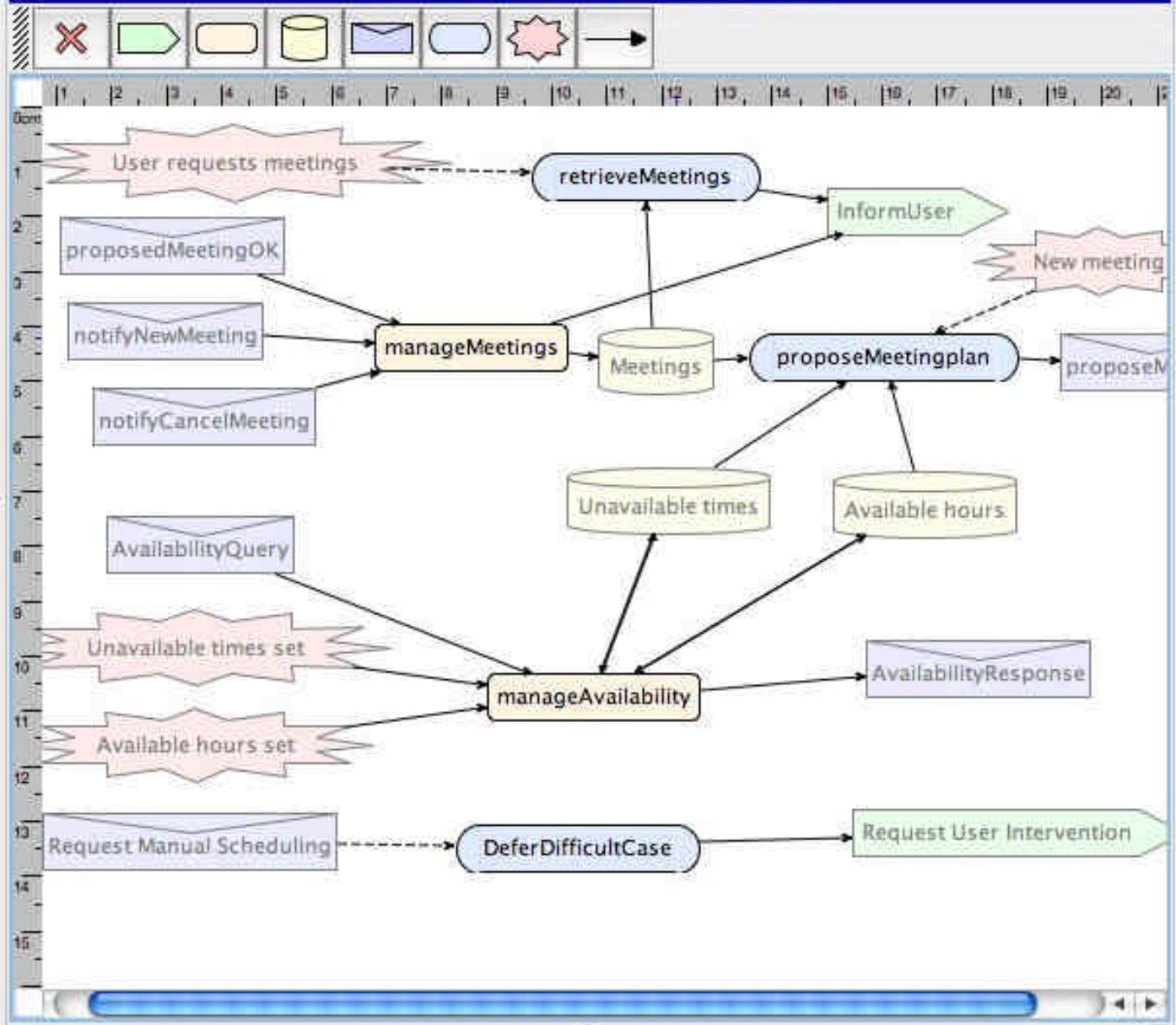
- manageMeetings
- manageAvailability
- Messaging
- Meeting Scheduler

Entities

System entities Filter

- Meeting accepted
- Meeting cancelled
- Meeting declined
- Request User Intervention
- GUI
- Meeting Scheduler
- Messaging

GUI - Agent Overview Diagram



Type	Name
Action	Meeting accepted
Action	Meeting cancelled
Action	Meeting declined
Action	Request User Intervention
Action	InformUser
Agent	GUI
Agent	Messaging
Agent	Meeting Scheduler
Data	Meetings
Data	Available hours
Data	Unavailable times
Capability	manageMeetings
Capability	manageAvailability
Functionality	Meeting Planner
Functionality	Interaction Manager
Functionality	Rescheduler
Functionality	Personal Planner
Message	proposeMeeting
Message	proposedMeetingOK
Message	checkPriority

Close

Type	Name
Plan	addMeeting1
Plan	addMeeting2
Message	AvailabilityQuery
Message	AvailabilityResponse
Data	Available hours
Percept	Available hours set
Goal	Avoid cascading reschedules
Plan	Cancel meet
Goal	Cancel meeting
Message	cancelMeeting
Plan	CancelMeetingplan
Scenario	Check meetings
Plan	Check user availability
Plan	checkAttendance
Message	checkPriority
Plan	checkPriorityplan
Plan	Clash
Scenario	Clash occurs with high priority
Scenario	Clash occurs with low priority
Goal	Consider meeting priority

Close



Consistency checking

Consistency check alternatives

- Check project agents
- Check project capabilities

Find unused:

- Actions
- Capabilities
- Functionalities
- Messages
- Plans
- Scenarios
- Agents
- Data
- Goals
- Percepts
- Protocols

OK: The message "checkPriority" is sent.
OK: The message "priorityResponse" is received.
OK: The message "priorityResponse" is sent.
OK: The message "scheduleMeeting" is received.
OK: The message "scheduleMeeting" is sent.
OK: The message "notifyCancelMeeting" is received.
OK: The message "notifyCancelMeeting" is sent.
OK: The message "notifyNewMeeting" is received.
OK: The message "notifyNewMeeting" is sent.
OK: The message "AvailabilityQuery" is received.
OK: The message "AvailabilityQuery" is sent.
OK: The message "AvailabilityResponse" is received.
OK: The message "AvailabilityResponse" is sent.
OK: The message "Request Manual Scheduling" is received.
OK: The message "Request Manual Scheduling" is sent.
OK: The message "cancelMeeting" is received.

- Ok
- Warnings
- Errors

Close

Clear

Check

index.html

file:///Volumes/winikoff/docs/students/jason/... Google

Address Book gmail AOPD AOPD mark guide 2004 RMITCS RMIT inside JACK Docs

index.html

Goals	Reschedule meeting , Avoid cascading reschedules , Consider meeting priority
-------	--

Functionality Personal Planner

Name	Personal Planner
Description	Concerned with the user interface. Obtains from the user when they are not available, reports to the user their meetings.
Triggers	Unavailable times set , Available hours set , User requests meetings
	Meetings , Unavailable times
	Available times , Available hours
	Available times , Update available hours , Notify user of meetings

File Tools Entities View Help

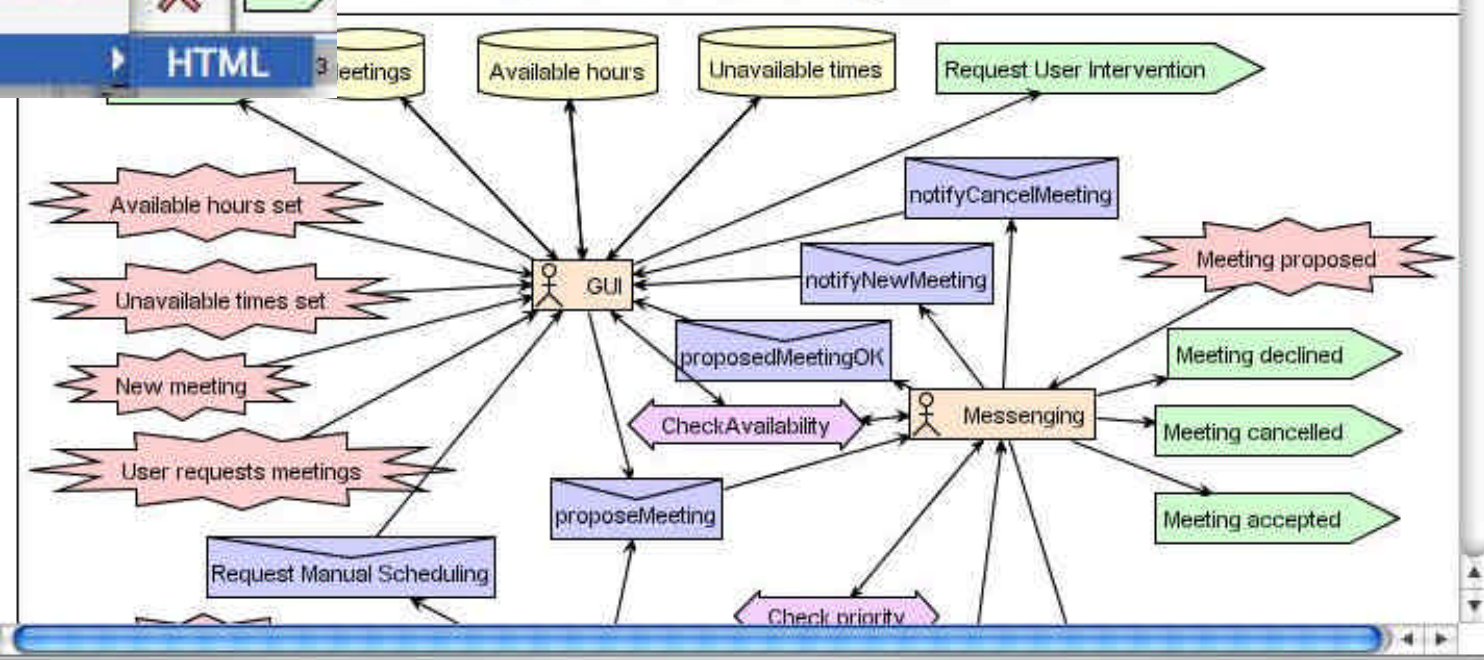
Crosscheck

Save Diagram As PNG

Generate Report

HTML

System Overview Diagram



index.html						
		cancel meeting	reschedule	reschedule meeting from meeting that has been rescheduled counter value for the meeting.		
	4	Scenario Scheduling a meeting	Meeting Planner	Meeting is rescheduled as per a normal meeting, with a higher reschedule counter value.		
Variation:						

Scenario New meeting received

Name	New meeting received.						
Description	The agent has received a meeting from another user's agent.						
Trigger							
Steps	#	Type	Name	Functionality	Description	Data produced	Data used
	1	Percent	Meeting proposed	Interaction Manager	New meeting has been received from another user's agent.		
	2	Goal	Consider user availability	Interaction Manager	Check whether user can make the meeting with their current calendar.	Meetings , Available hours	
	3	Action	Meeting accepted	Interaction Manager	New meeting is accepted, reply is sent back to the agent.		Meetings
	4	Goal	Notify user of meetings	Personal Planner	Notify user of new meeting.	Meetings	
Variation	If the user is unavailable for some reason, this is considered a clash and is handled as per the 'Clash occurs' scenario. If the meeting has been rescheduled too many times, it is flagged for manual rescheduling according to the 'Meeting cannot be rescheduled' scenario.						

Scenario Modify unavailable times

Name	Modify unavailable times						
Description	The user modifies their rules for unavailable times, either modifying or deleting existing rules, or adding a new rule.						
Trigger							
Steps	#	Type	Name	Functionality	Description	Data produced	Data used
	1	Percent	Unavailable times set	Personal Planner	User submits new unavailable time or modifies existing rule.		
	2	Goal	Update unavailable times	Personal Planner	Update Unavailable times KB with new details.		Unavailable times
Variation	If a rule to be modified or deleted does not exist, an appropriate error is returned to the user.						



Consistency checking

Consistency check alternatives

- Check project agents
- Check project capabilities

Find unused:

- Actions
- Capabilities
- Functionalities
- Messages
- Plans
- Scenarios
- Agents
- Data
- Goals
- Percepts
- Protocols

OK: The message "checkPriority" is sent.
OK: The message "priorityResponse" is received.
OK: The message "priorityResponse" is sent.
OK: The message "scheduleMeeting" is received.
OK: The message "scheduleMeeting" is sent.
OK: The message "notifyCancelMeeting" is received.
OK: The message "notifyCancelMeeting" is sent.
OK: The message "notifyNewMeeting" is received.
OK: The message "notifyNewMeeting" is sent.
OK: The message "AvailabilityQuery" is received.
OK: The message "AvailabilityQuery" is sent.
OK: The message "AvailabilityResponse" is received.
OK: The message "AvailabilityResponse" is sent.
OK: The message "Request Manual Scheduling" is received.
OK: The message "Request Manual Scheduling" is sent.
OK: The message "cancelMeeting" is received.

- Ok
- Warnings
- Errors

Close

Clear

Check