System Goal Modelling using the $i^*$ Approach in RESCUE

Centre HCI Design
24th March 2003
Tutorial Timetable

A simple timetable

– Monday 24th March 2003

Am: Develop Strategic Rationale Models
    Practice on case studies

Pm: Continue Strategic Rationale Models
    Use the REDEPEND tool
Part 1:
Modelling a Strategic Rationale Model for each Actor
Strategic Dependency Modelling

Network of dependency relationships among actors

– Depender who is the actor who “wants” something
– Dependee who has the “ability” that something

Explore first of all using dependencies tables

<table>
<thead>
<tr>
<th>Subject</th>
<th>Noun</th>
<th>Dependency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>depends on Agent</td>
<td>for something</td>
</tr>
<tr>
<td>Pilot</td>
<td>depends on Controller</td>
<td>to be safe (SG)</td>
</tr>
<tr>
<td>Pilot</td>
<td>depends on Controller</td>
<td>for instructions (R)</td>
</tr>
<tr>
<td>Controller</td>
<td>depends on Pilot</td>
<td>to pilot aircraft (T)</td>
</tr>
<tr>
<td>Student</td>
<td>depends on Neil</td>
<td>to learn well (SG)</td>
</tr>
<tr>
<td>Neil</td>
<td>depends on Student</td>
<td>to deliver lecture (T)</td>
</tr>
<tr>
<td>Customer</td>
<td>depends on Airline</td>
<td>to have tickets bought (G)</td>
</tr>
</tbody>
</table>
Strategic Dependency Model
For Internet Airline Ticketing System

Think about clusters of dependencies
Strategic Rationale (SR) Modelling

An intentional description of desirable processes
- In terms of goals, tasks, resources and soft goals
- Actors accomplish goals and tasks
  - SR model specifies what actors accomplish themselves
  - Adding SD model specifies what need other actors to accomplish

Three additional semantics added to SD semantics
- Task decomposition links
  - Decompose task into sub-components of all types
- Means-end links
  - A relationship between an end and a means for attaining the end
- Contributes-to soft goal links
  - A means-end link with a softgoal as the end
Task Decomposition Links

Decompose task into sub-components of all types

- All sub-components need to be “completed” for task to be performed, so logical “AND” between them

- Answers “what?” questions

- Any process element

- Task: Purchase ticket

- Goal

- Task

- Resource

- Soft goal

- Be logged on

- Select flight

- Credit card number

- Usable web-site
Task Decomposition Links

Four types of task-decomposition link

- Task goal decomposition

- Task sub-task decomposition

- Task resource decomposition

- Task soft goal decomposition
Means-End Links
Relating requirements and solutions

- Relationship between an end (a goal or resource) and a means (how to do something expressed as a task) for attaining the end

- Models alternative ways (tasks) for accomplishing a goal
- Provides a logical “OR” relationship between sub-components
- Provides answers to “HOW” and “WHY” questions

How to receive a booking confirmation?
- E-mail or letter

Booking is confirmed
- Receive e-mail
- Receive letter

To confirm the booking
- Why receive an e-mail?

End
Goal
Means
Task
Link
Four main types of Means-End Links

Goal-task link
- End is specified as goal and means is specified as task
- Goal might have different means ends links

Resource-task link
- Task indicates how resource can be obtained

Goal-goal link
- Permits the reduction of goals to sub-goals

Task-task link
- Represents one method for decomposing a task into possible alternative subtasks, related with a logical “OR” relationship
Contribute-To Soft Goal Links
A means-end link with a soft goal as the end

- Represents the positive or negative contribution of a process element towards the achievement of a soft goal

- Identifies requirements trade-offs by positive or negative contributions to satisfaction of the soft goal

- Provides an implied logical “OR” relationship between sub-components with +ve and -ve contributions

---

**Task**

<table>
<thead>
<tr>
<th>Contributes to</th>
<th>Soft goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>+/-</td>
<td>Task</td>
</tr>
</tbody>
</table>

---

**Purchase quickly**

- Purchase online
- Purchase via agent
Contribute-To Soft Goal Links

A means-end link with a soft goal as the end

– Note that no means can be specified for the achievement of a goal that cannot be clearly defined
– The only restriction is that the end should always be a soft goal
– Different types of contribute-to soft goal links are permissible

– Purchase online and internet connections both contribute positively, while seeking to minimise own work does not
Heuristics for Rationale Dependencies

*i*+ heuristics to guide rationale modelling

- Task decomposition rules imply no ordering
- Sub-goal of a task to be treated as a pre-condition on the task being completed
- Sub-soft goal of a task must be satisfied by the completion of task, that is post-condition
- When deciding whether to model a concept as a task or goal, always choose a goal unless there are 2 specific ways of doing something
- A task is often decomposed into at least one sub-task, one sub-goal and one sub-resource
- Model the right goals, soft goals, resources and tasks of each actor - mis-allocation is common
- Ask pre-defined questions of each model to check its completeness
Cross-Checking SD and SR Models

The SD model supports the SR model
- Elements of SD model become elements of SR model

Learn the simple heuristic for Actor A
- IF Actor A is a depender is a dependency relationship in the SD model THEN the depended-upon element is modelled in Actor A’s SR model
- A depender wants something - that wanted goal, soft goal, task or resource is included in relevant SR model
- The wanted element is linked using the SD dependency links to other actors in the SD model
How to Produce Strategic Rationale Model

Three key stages

1. Model what each actor can accomplish itself
   • If there are no mandated solutions, start with soft goals
   • If a solution, such as online purchasing, has been mandated, start with high-level tasks
   • Develop contribution hierarchies from soft goals
   • Explore different means of achieving ends (design ideas)
   • Model the requirement (goals) and solution (tasks) space using layered model technique
   • Ensure SR model is complete with respect to SD model

2. Model what each actor depends on other actors for
   • Include dependencies from and to actors in SD model

3. Integrate the single-actor SR models together
   • Use the dependencies from the SD model
Modelling Requirements and Solutions

Consider the cake metaphor to help you

Model requirements as goals
Model solution ideas as tasks
Model requirements as goals
Model solution ideas as tasks
Exercise:

\( i^* \) Strategic Rationale Modelling - Modelling What Each Actor Can Do
Train Signalling Safety Requirement

Learning objective
- To practice SR modelling for a single actor

Problem
- The paramount goal for training signalling near a mainline London railway terminal is that the signal is safe. Analysis is required to decompose this soft goal so that it is better understood, and that some of the goals have measurable fit criteria. If you want to think about a concrete example, recall Paddington’s Signal 109

Task
- Produce an i* strategic rationale model to decompose the high-level safety soft goal
Train Signal $i^*$ SR Model

- Signal is safe
  - Signal works reliably
    - Maintain signal
  - Signal is usable
    - Signal is clearly visible to driver
      - Signal colours are clear to driver
        - Signal visible to driver in good time
        - Effective signal location
  - Send warning of fault to signal box
  - Attend training course
    - Attend classes
    - Sit the exam
    - Training material
  - Safety certificate
    - Driver trained effectively
      - Available instructors
  - Pass reliability test
    - Effective signal location
Automated Bus Indicators

Learning objective
- To practice developing Strategic Rationale models

Problem
- From the information given and your own solutions

Task
- Model SR models for several key actors
- Produce a SR model for the passenger actor consist with the SD model
- Produce a SR model for the route controller actor consist with the SD model
- Produce a SR model for other actors as time permits
Part 2:
Linking the Actor Models Together
Modelling Dependencies on SR Models

Connect actor SR models using dependencies

- Include each SD model dependency between actors or their elements in the SR model
- Mechanical process if SD and SR models are effectively cross-checked
- **IF** actor has no SR model **THEN** link dependency to the actor
Producing a Single SR Model

Integrate partial SR models
- Connect using all dependency links from SD model
- Add additional dependency links that emerge from the richer SR models
- Check the model for correctness, completeness and consistency
Strategic Rationale Model
For Internet Airline Ticketing System

Airline
- Maximise revenues
  - Income
  - Repeat business

Passenger
- Purchase quickly
- Purchase cheaply
- Online purchase
- Go travel agent
- Select flight(s)
- Credit card
- Tickets purchased
- Destination selected

+ -
Strategic Rationale Model: CORA 2

[Diagram showing the strategic rationale model with various processes and interactions such as Resolution generation, Update, sequencing tools, controller, and others.]
Cross Model Checks in RESCUE

Cross-check models at this stage

- Compare \( i^* \) SR model and activity model to check that goals, resources, constraints and context in activity modelling appear, where relevant, in \( i^* \) SR model

- Compare \( i^* \) SR model and use case model to check that all use cases in the use case model are expressed as at least 1 task in the \( i^* \) SR model for the future system, and vice versa

- Compare \( i^* \) SR model and system-level and use case requirements to check that all goals and soft-goals to be achieved by the future system (according to the \( i^* \) SR model) are specified in the system requirements specification and stored in the requirements data base
Exercise:

*i* Strategic Rationale Modelling - Integrating Single-Actor Models
i* Strategic Rationale Modelling

Learning aims

– To develop i* modelling skills, to understand the nature of actor dependencies in a multi-actor strategic rationale model

An airport security system

– Assume that there are 2 actors - the airport security team and the passenger. The paramount soft goals of both are that the security system at the airport shall be secure. The security team undertake different types of searches to ensure security. Passengers want the checks to take place with minimum disruption to them.

Advice

– If you want to think about a concrete example, think about getting on an aircraft in Europe
Airline Security Example

- Passenger
- Check each passenger
- Check securely
- Check is reliable
- Check is robust
- Passenger is checked
- Passenger throughput
- Repair in 20 mins
- Search the person
- Use detector
- Metal detected
- Safety standards met
- Power supply
- Staff are well-trained
- Maintain each week
- Quality equipment
- Be safe
- Pass through securely
- Pass through easily
- Pass through security
- Airport security

Be safe

- Enough staff
- Maintain each week
- Quality equipment
- Be safe
- Pass through securely
- Pass through easily
Airline Check-In System

Learning aims

- To develop i* modelling skills, to understand the nature of goal dependencies and their complexities, and to use these models to make requirements trade-offs

An airline hand-baggage only check-in system

- An airline is reconsidering the design of its rapid check-in system. Passengers have to check-in to flights. Airline management want this check-in process to be more cost-effective. Passengers want a usable system that enables them to check in more quickly. All stakeholders require the system to be reliable. Management also want no additional training for staff or passengers to use the system. Different alternative solutions (tasks) are current manual check-ins, automatic check-in machines, and telephone check-in prior to arrival at the airport.
Airline Check-In Example

Airline staff
- Have job security

Check-in system
- Passenger checked-in to flight

Check-in by telephone
- Check-in manually
- Check-in automatically

Passenger
- Checked-in to flight
- Successfully
- Easily
- Quickly

Airline mgmt
- Check-in effectively
- Check-in is cost-effective
- Reliably

Check-in automatically
- +

Check-in manually
- -

Check-in by telephone
- +

Passenger
- +

Airline staff
- -
Automated Bus Indicators

Learning objective
– To practice developing Strategic Rationale models

Problem
– From the information given and your own solutions

Task
– Integrate SD and SR models for several key actors
– Add dependencies with other actors to the SR model for the passenger actor
– Add dependencies with other actors to the SR model for the route controller actor
– Define dependencies between process elements for the passenger and route controller actors to produce a first-cut integrated SR model
Part 3: REDEPEND $i^*$ Modelling Tool
REDEPEND Software Tool

REDEPEND (RE DEPENDDencies)
- Centre prototype for developing $i^*$ SD and SR models
- MS-Visio plug-ins to draw and analyse models
**Model Checking with REDEPEND**

Model checking
- \( i^* \) models are large, complex and necessitate computational model checking
- Check for unrecognised connections, invalid connections and model **bottlenecks**
REDEPEND Overview

Brief Overview
- Version 1 developed by Alexis Gizikis (March 2001)
- Version 2 developed by Rahoof Nazir (June 2002)
- Version 3.1 circulated to DMAN project recently
- Runs as an add-on stencil for Microsoft Visio

Intended to be simple to use
- Basic drag and drop technique
- Basic copy and paste commands to include models in ORDs.

Simple to install
- Copy and paste files into MS Visio
Exercise:
Using REDEPEND
Develop 2 $i^*$ Models in REDEPEND

Learning objective
– To practice using the REDEPEND modelling tool

Problem
– Any of the previous problems

Task
– To redraw one previous SD model and one previous SR model using the REDEPEND software tool
Part 6: Summary of the Process
System Goal Modelling

First synchronisation stage
- Gather data on human processes
- Determine system boundaries
- Develop use case model
- Define system-level requirements

Second synchronisation stage
- Model human activity
- Determine system dependencies, goals and rationale
- Describe use cases
- Define and document requirements

Third synchronisation stage
- Refine system dependencies, goals and rationale
- Specify use cases
- Define and document requirements

Fourth synchronisation stage
- Use case specifications
- Walkthrough scenarios
- Define and document requirements

- Impact analysis
- Refine and change requirements