Goal and Agent-oriented Modeling with i* Tutorial

Jaelson Castro
Universidade Federal
Pernambuco
Recife, Brazil

Fernanda Alencar
Universidade Federal
Pernambuco
Recife, Brazil

Ricardo Ramos
Universidade Federal
Do Vale do São Francisco
Juazeiro, Brazil
 Agenda

• Requirements Engineering
• Goal-Oriented Requirements Engineering
• i* modeling concepts
• Exercise hand-out
• Guidelines, Q&A
• The i* community, i* wiki
• Conclusions
Goal and Agent-oriented Modeling with i*
Why Manage Requirements?

Even small requirement errors can lead to big problems!
The Cost to Fix Software Defects

- Requirements Errors: 82%
- Design Errors: 13%
- Other Errors: 10%
- Coding Errors: 1%

Source: "An Information Systems Manifesto" by James Martin
Recognizing Where the $$$ Goes

When it comes to software development, a pencil costs just pennies...

...it is the eraser that can cost millions.
Typical Development Process

I'll go up and find out what they need and the rest of you start coding!
It is not easy......

........ To understand the functionality

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
It is not easy......

........ Obtain the right shape

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
It is not easy......

....... to satisfy a cliente

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
It is not easy......

........to understand problems that you are not familiarized
............to understand the details of the solution.

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Nicholas Carr: “IT doesn’t matter”

- From 1992 to 2001, US companies spent over $2.7T on hardware, software, and services – IDC
- Research (2002) shows only a random correlation between IT spending per employee and return on shareholder equity – Strassman
- On average, only 7% of software functionality that was paid for is actually used – Gartner
- IT projects often suffer from a prolonged delay to realizing value, an average of 18 to 24 months from initiation to operations (usually only providing a one-time cost impact) – Standish Group (2003)
- 85% of IT projects fail to meet objectives (with 32% being cancelled outright) – Gartner

But does this mean that IT Doesn’t Matter?
We think not

[Howard Smith, CSC, 2004]

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
80% of a typical product’s development time is spent in correcting errors.

Most rework is related to inadequate, inconsistent, imprecise requirements.

Fixing a defect late in the development life cycle can cost 30-1000 times more.

debbuging … redeveloping…

“Hidden Factory”


Relative Cost… Avoid vs. Fix

Cost of getting it wrong

• Causes of project failure
  – Survey of US software projects by the Standish group:

  **Top 3 success factors:**
  1) User involvement
  2) Executive management support
  3) Clear statement of requirements

  **Top 3 factors leading to failure:**
  1) Lack of user input
  2) Incomplete requirements & specs
  3) Changing requirements & specs
Requirements Engineering

- It is the phase where technology meets the real world, where technical considerations have to be balanced against personal, organizational and social ones;
- This calls for special skills on the part of the requirements engineer, and makes the phase particularly challenging.
- Requirements engineering - not only to elicit and specify what the user wants, but help explore what is possible, desirable, and viable
- The importance of detailed design and implementation will wear off over time, thanks to software reuse, COTS, MDA, SPL and the like; requirements analysis will always be there and will always be important.

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Requirements Analysis

This activity traditionally boils down to three tasks:

Context analysis -- the reasons *why* the system is to be created and *why* certain technical operational and economic feasibilities are the criteria that form boundary conditions for the system.

Functional requirements -- what will the system do.

Non-functional (quality) requirements -- global constraints on how the system is to be constructed and function.

The need for GORE
Goal and Agent-oriented Modeling with i*

Goal-Oriented Requirements Engineering - GORE
Goal-Oriented Requirements Engineering (GORE)

- Appeared ~1993
- Focuses on early requirements phases, when alternatives are being explored and evaluated.
- Starts from initial goals, such as “Higher profits”, “Faster time-to-market”, “Schedule meeting”, “Easily maintainable system”, “Good performance” etc.
- Keeps decomposing the goals until reducing them to alternative collections of design decisions, each of which can satisfy the initial goals.

Initial goals may be organization or system-oriented; they may also be contradictory, so the analysis must facilitate the discovery of tradeoffs and the search of the full space of alternatives, rather than a subset.
Goal-Oriented Analysis is not New!

- Many other researchers worked with goals a decade or more ago including:
  - Specification of composite systems (Feather, 1987)
  - Knowledge representation and reasoning in the design of composite systems – Critter (Fickas, 1992)
  - NATURE (Jarke, 1993)
  - F3 (Bubenko, 1993)
  - Goal-oriented requirements acquisition -- KAOS (Dardenne, 1993)
  - Non-Functional Requirements framework (Chung, 1993)
  - Goal-oriented elaboration of requirements -- ALBERT (Dubois, 1994)
  - i* framework (Yu, 1995), etc.
Goal Analysis leads to Alternatives

- Schedule Meeting
  - AND
  - Collect Timetables
    - OR
    - By Person
    - OR
    - By System
      - OR
      - Manually
      - OR
      - Collect from agents
        - AND
        - Collect from users
          - AND
          - Send request
            - OR
            - Receive request

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Alternatives Lead to Designs/Plans

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Benefits of GORE

- According to Axel van Lamsweerde (ICSE 2000) & John Mylopoulos (RE 2006)
  - Systematic derivation of requirements from goals
  - Goals provide rationales for requirements
  - Goal refinement structure provides a comprehensible structure for the requirements document
  - Alternative goal refinements and agent assignments allow alternative system proposals to be explored
  - Goal formalization allows refinements to be proved correct and complete
  - GORE as a foundation for a Theory of Software Design.
Non-functional goals

- Functional (hard) goals, such as “Schedule Meeting”, are well defined in the sense that they admit a formal definition.

Not all goals are functional

- Non-functional goals, such as “Higher Profits”, “Higher Customer Satisfaction” or “Easily Maintainable System” specify qualities a socio-technical system should adhere to.

Such qualities are represented as softgoals.
Softgoals

- Usually admit no generally agreed upon definition
- Are inter-related and often conflicting.
- Define how well the system accomplishes its functions
- Can be thought as “fuzzy goals” with no clear-cut criteria for satisfaction
- Are satisficed, rather than satisfied

NFR framework (Chung, 1993)
Early Requirements Engineering

- Why Early RE?
- What modelling do we need in order to support Early RE?
Why is Early RE important?

• What do you need to be concerned about?
  – Solving the wrong problem
  – Socio-technical system failure, disuse
  – Changing needs
  – Changing regulations
  – Globalization, internationalization
Why is Early RE important?

- Complex relationships among stakeholders
  - what they want
    - E.g., security, privacy, trust, profitability, market positioning, strategic alliances, intellectual property, …
  - How they can achieve what they want
- Need systematic method, bring into RE process
  - modelling and reasoning support, tools, traceability, …
- Before defining the system to be built
- Consider:
  - E-business; E-learning; E-health; E-government
  - Energy, environment, transportation
What to look for

- Most systems today exist in complex socio-technical settings
- How do we boil down to small number of modeling constructs?
- What do we look for in
  - Expressiveness
  - Reasoning support
So what are the important concepts for Agent Orientation as a Modelling Paradigm?

- Intentionality
- Autonomy
- Sociality
- Identity & Boundaries
- Strategic Reflectivity
- Rational Self-Interest

Goal and Agent-oriented Modeling with i*  

i* modeling concepts
i* Framework

• i-Star (intentional)
• Proposed by Eric Yu, University of Toronto (MIP RE’07)
• Approach centered in system stakeholders and their relationships
  – Actors depend on each other to fulfill their goals
• Helps to answer questions, such as:
  – Why is a requirement of one type and not of another type?
  – Why is a specific requirement needed?
• Helps to understand the system requirements, as well as to prepare them to future changes
Approach

• Strategic actors modelling
  – To model and analyze complex relationships among actors with strategic intent
    • includes humans and machines

• What $i^*$ does not aim to do
  – Execution level analysis
  – Temporal dimension
Approach: model social relationships for analysis and design

• Strategic actors
  – What do I want?
  – How can I achieve what I want?
  – Who do I depend on to achieve what I want?
i* Framework

i*(iStar)

SD Model

Actor

Intentional Element

SR Model

Dependency / Relationship

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Strategic Actors

- Have goals, beliefs, abilities, commitments
- Are semi-autonomous
  - freedom of action, constrained by relationships with others
  - not fully knowable or controllable
  - has knowledge to guide action, but only partially explicit
- Depend on each other
  - for goals to be achieved, tasks to be performed, resources to be furnished
Strategic Actors

- It is an active entity which performs action to achieve goals
- Actors depend on each other to fulfill goals, perform tasks and provide resources
- Thus, they can accomplish goals hard to be achieved by themselves
- They are represented by a circle in the i* diagrams
Strategic Actors

- They can be
  - agents (humans or not),
  - roles (functions) or
  - positions (job locations/job positions)
i* Models

• Includes two basic models:
  – Strategic Dependency (SD) : describes dependency relationships between actors
  
  ![Strategic Dependency Diagram]

  – Strategic Rationale (SR) : explains how actors achieve their goals
  
  ![Strategic Rationale Diagram]
A dependency is a type of relationship between two actors. It describes an intentional connection where one actor (the dependum) relies on another actor (the dependum) for some service or information. This relationship is typically represented in a diagram as follows:

\[
\text{Actor 1} \rightarrow \text{Dependum} \rightarrow \text{Actor 2}
\]

- The dependum defines a kind of dependency and describes the nature of the agreement between two entities.
Types of Dependencies

- Dependum types:
  - goals,
  - tasks,
  - resources, or
  - softgoals
Dependencies

Diagram showing relationships between actors and dependencies.
i* Framework – an example

- Media Shop is a store selling and shipping different kinds of media items such as books, newspapers, magazines, audio CDs, videotapes, and the like.
- To increase market share, Media Shop has decided to open up a B2C retail sales front on the internet.
- The system has been named Medi@ and its basic objective is to allow an on-line customer to examine the items in the Medi@ internet catalogue, and place orders.

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Resource

- It is an entity, physical or not, whose main concern is its availability
  - Dependencies involving a resource imply that the dependee actor provide the resource to the depender actor
Task

• Specify a specific way to do something. It can be seen as a solution in the system-to-be
  – These solutions are functions, operations, procedures, processes, etc.
  – Dependencies involving tasks happen when an actor depend on another actor to perform an activity.
Goal

• It is a condition, or state of affairs that the stakeholders wish to achieve
  – It is not specified how the goal will be achieved, what allows considering many alternatives;
  – It is allowed goal dependencies between actors to represent responsibility delegation to satisfy a specific goal
Softgoal

• It is a condition, or state of affairs that the stakeholders wish to experiment
  – In oposition to the goal concept, there is no clear-cut criteria to state that the softgoal have been achieved
  – Hence, its degree of satisfaction is subjective. Different stakeholders may have distinct evaluation!
Criticality of the Dependencies

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
SD Model

Customer

Availability

Browse Catalogue

Place Order

Security

Buy Media Items

Increase Market Share

Media Shop

Process Internet Orders

Process Online Money Transactions

Accounting

Media Items

Continuing Business

Internet Services

Telecom Cpy

Bank Cpy

Media Supplier

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
SD and SR models

• When the relevant actors have been identified, the SD model can be seen as concluded and we can move to the SR model

• SR models the intentional relationships inside an actor
  – Intentional Elements (goals, tasks, resources, softgoals)
  – Means-end, task-decomposition and contribution links

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
SR: New Relationships

Relacionamento Melo-fim

Contribuição

 Decomposição de tarefas

i*Wiki:
http://istar.rwth-aachen.de/tiki-index.php

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Means-end link

- Relationship meaning an intention to achieve an end
- This end can be a goal, a resource, a softgoal or a task
- The means to achieve this end are defined as tasks to be performed

There is a notion of alternative
Contribution link

• A special kind of means-end link
  – Indicate goals, resources or tasks which contribute positively or negatively to fulfill a particular goal

• How operationalizations will contribute to achieve the “Securança” softgoal?
Contribution Relationships

- Make
- Some+
- Help
- And
- Unknown

- Break
- Some-
- Hurt
- Or
Task-decomposition link

- A task is related to its sub-components through a decomposition relationship
  - The four kinds of intentional elements can be a sub-component of a task
Roles, Agents, Positions

- Role as abstract actor
- Agent as concrete actor
- Position as a set of roles typically assigned to one agent
An Example

Meeting Scheduler

From: E. Yu. Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering
Strategic Dependency (SD) model

[Yu RE97]

Meeting Initiator → Preferred Dates (p) → Proposed Date (m) → Assured (Attends Meeting (ip,m))

Meeting Participant → Exclusion Dates (p) → Agreement (m,p) → Attends Meeting (p,m)

Meeting Scheduling Example
Strategic Rationale (SR) model

Strategic Rationale Model with Meeting Scheduler

- Meeting Initiator
- Organize Meeting
- Quick
- Let Scheduler Schedule Meeting
- Schedule Meeting
- Low Effort

- Attend Meeting
- Meeting Participant
- Participate
  - Attend Meeting
  - Convenient
    - (Meeting, Date)
  - Arrange Meeting
  - Low Effort
  - User Friendly
  - Richer (Meeting, Date)

- Find Agreeable Date
  - Find Agreeable Date Using Scheduler
  - Agreeable (Meeting, Date)
  - Proposed Date
  - AgreeToD ate
  - Merge Available Dates
  - Obtain Agreement
  - Enter Date Range

- User Friendly
- Richer (Meeting, Date)
- Low Effort
- Agreeable (Meeting, Date)
- Find Agreeable Date
  - User Friendly
  - Richer (Meeting, Date)
  - Low Effort
Strategic Rationale Model

Development-World model
refers to and reasons about…

As-is

Alt-1

Alt-2

To-be

Operational-World models

Strategic Dependency Models
Analysis and Design Support

• opportunities and vulnerabilities
  – ability, workability, viability, believability
  – insurance, assurance, enforceability
  – node and loop analysis

• design support
  – raising issues
  – exploring alternatives
  – evaluating, making tradeoffs
  – justifying, settling
  – based on qualitative reasoning
Softgoal Operationalizations: Contribution Relationship

Side-effects to softgoals: Correlation Relationship

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Analysis/Evaluation of $i^*$ Models  [Jennifer Horkoff]

• The creation of $i^*$ models is a useful activity

• However…

  – We can make further use of models by evaluating them.

  – Purpose is to determine to what degrees stakeholder goals will be satisfied or denied, given a particular situation or design option.
Qualitative, Interactive, Forward Analysis of i* Models

- Evaluation values propagated through the model using a combination of propagation rules and human judgment.

- Our approach to analysis is interactive.
  - The procedure prompts the user for input at various points in the procedure.

- Our approach to analysis is qualitative.
  - We use six course-grained qualitative analysis values, based on evaluation in the NFR Framework:
    - Satisfied, Partially Satisfied, Conflict, (Unknown), Partially Denied, and Denied
Example:

- Evaluation based on an analysis question:
  - If the Application implements **Restrict Structure of Password**, but not **Ask for Secret Question**, what effect will this have on **Attract Users**?
- Place Initial Labels reflecting Analysis Question
Example:

- Propagate labels
- Resolve labels
- Iterate on the above steps until all labels have been propagated

Human Intervention

Usability Receives the following Labels:
- Partially denied from Restrict Structure of Password
- Partially denied from Ask for Secret Question

Select Label…
Select denied

© Eric Yu 2010
Interactive Evaluation of i* Models

• Analyze result
  – If the Application implements **Restrict Structure of Password**, but not **Ask for Secret Question**, **Attract Users** is partially denied, as **Usability**, considered important by the evaluator, is denied.
  – This is not a viable design alternative.
• Next Steps:
  – Repeat with new analysis question…
Another Example:

*Car insurance*

From: E. Yu. WITS94.
The Strategic Dependency Model

*automobile insurance – example 1*
The Strategic Dependency Model

auto insurance – example 2
“Let the Insurance Agent handle it.”

The Strategic Dependency Model

auto insurance – example 3
“Let the Body Shop handle it.”
The Strategic Rationale Model

means–ends links and
task decomposition links
The Strategic Rationale Model

“Functional” Alternatives

The Strategic Rationale Model
“Non-Functional” Rationales
Goal and Agent-oriented Modeling with i*

Exercise
Exercício – parte 1

• Fazer o modelo de **requisitos iniciais** para uma empresa que realiza **cursos**
Exercício 1: Construir Modelos i* nos Requisitos Iniciais
Descrição

• Objetivo
  – Exercitar uso da Notação i*

• Atividade
  – Construir um modelo SD de acordo com o cenário apresentado
  – Usar pelo menos uma vez as construções apresentadas (Decomposição, Meio-Fim e Contribuição)
  – Não incluir soluções de software
Cenário: Organização de Curso

• Em uma instituição de ensino serão organizados cursos. Durante a preparação de um curso, alunos, instrutores e os organizadores necessitam interagir para que o curso seja ministrado
Cenário: Organização de Curso

- Os alunos interessados pelo curso procuram aprender o novo assunto
- Para isso além de se inscrever e pagar pela matrícula, precisam fazer os exercícios e comparecer às aulas

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Cenário: Organização de Curso

- Além de ministrar o curso o instrutor também é responsável por preparar o material de aula.
- Em troca ele espera ser pago pelo trabalho.
- Para que o curso seja bem sucedido o instrutor espera que os alunos compareçam as aulas e façam os exercícios.
Cenário: Organização de Curso

- A organização do curso por sua vez é responsável por realizar todas as atividades de inscrição incluindo a matrícula, a coleta dos dados do aluno e o pagamento pelo curso.
- Além disso a organização deve garantir que o instrutor prepare as aulas e o material de aula. Para isso deve fornecer todos os recursos necessários incluindo o pagamento.
Hints - Framework i*

i*Wiki: http://istar.rwth-aachen.de

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Exercício 1 - SD

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Exercício – parte 2

• Fazer o modelo de **requisitos finais** para uma empresa que realiza **cursos**
Descrição

• Objetivo
  – Exercitar uso Notação i*

• Atividade
  – Construir um modelo SD e um modelo SR de acordo com o cenário apresentado
  – Usar pelo menos uma vez as construções apresentadas (Decomposição, Meio-Fim e Contribuição)
  – Refazer modelagem e modificar modelo para incluir soluções de software

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Cenário: Organização de Curso

- Reconhecendo o excesso de trabalho para os organizadores do curso durante a matrícula a instituição de ensino resolveu implantar uma solução de software para auxiliar esse processo.
- Qual seria a solução e qual seu impacto para a organização?

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Hints - Framework i*

i*Wiki: http://istar.rwth-aachen.de

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Exercício 2 – Exemplo de SD

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Exercício 2 – Exemplo de SR

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Exercício 2 – Exemplo de SR

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Guidelines

- **Goal**: To avoid some of the common misuses of i*
- **Audience**: It is intended to be both an introduction to *new* users and a resource for *experienced* users
- **Examples**:
  - **Actor**: Dangling Actor, Internal Actor, Actor Boundary, Actor Dependency
  - **Goal**: Softgoal, Goal refinement
  - **Dependency**: Internal and External dependency links
  - **Contribution**: Internal and External contribution links
  - **Means-End**: Internal and External means-end links
  - **Decomposition**: External decomposition link
- **Sources**: [Webster et al. 2005], [IStarGuide 2008] & [Santos_2008]
Dangling Actor

<table>
<thead>
<tr>
<th>1 - Dangling Actor</th>
<th>Actor without link to another Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>An Actor without link to another Actor <strong>does not</strong> contribute with information to model</td>
</tr>
<tr>
<td>Question</td>
<td>Are there Actors without dependencies or structural actor’s links (i.e., Is-A, Is-part-of, Covers, Occupies, Plays)?</td>
</tr>
</tbody>
</table>
Dangling Actor

Correct

Wrong
## Internal Actor

<table>
<thead>
<tr>
<th>2- Internal Actor</th>
<th>Do not include an Actor within another Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Actors are active and autonomous entities that should be modeled separately. Only intentional elements can be inside an actor (eg. Goal, Softgoal, Task or Resource)</td>
</tr>
<tr>
<td>Question</td>
<td>Is there more than one actor inside the same boundary?</td>
</tr>
</tbody>
</table>
Internal Actor

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
## Actor Boundary

<table>
<thead>
<tr>
<th>3- Actor Boundary</th>
<th>Internal element connection to Actor boundary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Do not connect internal elements (goals, tasks, etc.) to the actor boundary</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any internal element connected to the actor boundary?</td>
</tr>
</tbody>
</table>
Actor Boundary

Correct

Wrong
## Actor Dependency

<table>
<thead>
<tr>
<th>4 – Actor Dependency</th>
<th>Use only the dependency symbol (“D”) to denote a dependency link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Do not use other types of links (Means-end, Contribution, Decomposition) to denote dependency links.</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any Actor Dependency which does not use the Dependency symbol?</td>
</tr>
</tbody>
</table>
Actor Dependency

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
## Softgoal

<table>
<thead>
<tr>
<th>5 - Softgoal</th>
<th>Softgoals are distinct from Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Softgoals are special Goals that have no clear cut achievement state. There are different levels of satisfaction (e.g., Satisfied, Partially Satisfied, Denied, etc.) They are used to capture quality issues or constraints such as: security, performance, etc.</td>
</tr>
<tr>
<td>Questions</td>
<td>Is there any Goal with adverbs or adjectives in their label? The condition expressed in label could be partially satisfied or not completely achieved?</td>
</tr>
</tbody>
</table>

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
# Goal Refinement

<table>
<thead>
<tr>
<th>6 - Goal Refinement</th>
<th>Linking of Goals in SR models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>In SR models</td>
</tr>
<tr>
<td></td>
<td>- Goals cannot be directly linked to other Goals</td>
</tr>
<tr>
<td></td>
<td>- Goals cannot be present at both ends of Decomposition, Contribution, or Means-Ends links</td>
</tr>
<tr>
<td>Question</td>
<td>Are there Goals linked to other Goals?</td>
</tr>
</tbody>
</table>

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Goal refinement

Correct

Wrong
## Internal Dependency Link

<table>
<thead>
<tr>
<th>7 – Internal Dependency Link</th>
<th>Do not use dependency links inside an actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Dependency links cannot be used to connect two intentional elements inside the same actor. They can be used to connect an internal element to a dependum, but not to connect two internal elements.</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any internal element that is connected to another internal element using a dependency link?</td>
</tr>
</tbody>
</table>
Internal Dependency Link

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
## External Dependency Link

<table>
<thead>
<tr>
<th>8 – External Dependency Link</th>
<th>Do not use a dependency link between two actors without showing the dependum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>The Dependency link should contain a Dependum. Extending the Dependency Link from the Depender to the Dependee without showing the Dependum does not convey what the dependency is about.</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any actor directly connected to another actor?</td>
</tr>
</tbody>
</table>
External Dependency Link

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
## Internal Contribution Link

<table>
<thead>
<tr>
<th>9 – Internal Contribution Link</th>
<th>A contribution link should only be used to connect an internal intentional element (of any type) to a softgoal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Contribution links are not allowed from any element to a goal, only to softgoals</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any Goal receiving Contribution links?</td>
</tr>
</tbody>
</table>

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Internal Contribution Link

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
# External Contribution Link

<table>
<thead>
<tr>
<th>10 – External Contribution Link</th>
<th>Do not use Contribution Links between actors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Contribution links are drawn within an actor and not between actors</td>
</tr>
<tr>
<td>Questions</td>
<td>Is there any contribution link between elements outside the actor’s boundary? Is there any contribution link between an internal element and an element outside the actor’s boundary?</td>
</tr>
</tbody>
</table>
External Contribution Link

Correct

Wrong

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
### Internal Means-End Link

<table>
<thead>
<tr>
<th>11 – Internal Means-End Link</th>
<th>Means-Ends are only used to link a Task to a Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>The only place where a Means-Ends link can be used is from a Task to a Goal</td>
</tr>
<tr>
<td>Question</td>
<td>Is there any Goal as a Means in a Means-ends link?</td>
</tr>
</tbody>
</table>
Internal Means-End Link

Correct

Wrong
# External Means-End Link

<table>
<thead>
<tr>
<th>12 – External Means-End Link</th>
<th>Do not extend means-Ends link beyond the boundaries of actors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Means-Ends Links are drawn within an actor and not between actors</td>
</tr>
<tr>
<td>Questions</td>
<td>Is there any Means-Ends link outside actor’s boundary? Is there any Means-Ends link between an internal element and an element outside actor’s boundary?</td>
</tr>
</tbody>
</table>
External Means-Ends Link

Correct

Wrong
## External Decomposition Link

<table>
<thead>
<tr>
<th>13 – External Decomposition Link</th>
<th>Do not extend Decomposition links beyond the boundaries of actors.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Decomposition Links are drawn within an actor</td>
</tr>
<tr>
<td>Questions</td>
<td>Is there any Decomposition link outside actor’s boundary? Is there any Decomposition link between a Task inside actor’s boundary and an external element?</td>
</tr>
</tbody>
</table>
External Decomposition Link

Correct

Wrong
Questions?
Goal and Agent-oriented Modeling with i*

The i* community, i* wiki
The i* Wiki
Fostering Investigation, Collaboration, and Evaluation

http://istar.rwth-aachen.de/

Current offerings
– Case Studies
– Events
– i* Quick Guides
– Publications
– i* Tools
– Who is Who
– Organizational Stuff

i* Wiki Team
Gemma Grau,
Jennifer Horkoff,
Dominik Schmitz,
Samer Abdulhadi,
Eric Yu
Welcome to the i* Wiki

This is the welcome page of the i* Wiki - intended to enable collaborative work on i* and related topics. Please visit Registration and User Rights Policy for the i* Wiki for questions regarding how to register for this site.

The general structure of the Wiki pages:

- i* Wiki Home
  - Case Studies
  - Events
  - i* Guides
  - Publications
  - i* Tools
  - Who is Who
  - Organizational stuff

For a view of the full structure of the Wiki please visit i* Wiki Home.

What's New

RE'10 Tutorial: Goal- and Agent-oriented Modeling with i*
By: Joshuec on: Thu 5 of Aug, 2010 [12:36 UTC] (1 reads)

A half day tutorial at RE'10

18th IEEE International Requirements Engineering Conference

Overview and a Comparison of \( i^* \) Tools

- We have created a questionnaire to evaluate and compare existing \( i^* \) modeling tools.
- The following categories are included in the questionnaire:
  - General information about the tool;
  - \( i^* \) modeling suitability provided by the tool;
  - Usability facilities provided;
  - maturity of the tool; and,
  - extensibility and operability with other tools, which includes the facilities for importing/exporting files and the development of new functionalities.

- Currently, there are 12 tool evaluations in the \( i^* \) wiki, submitted by developers of each tool.
  - Please update your tool description if needed.
- There is also a comparison table which summarizes the main features of the tools.

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Available i* Tools

See a table summary of the features exhibit by this tools in the section Comparing the i* Tools.
See the published metamodels in the section i* Metamodels.

- **OpenOM**
  - As a standalone application and as a plug-in for other popular tools, such as Eclipse and Protegé, OpenOM is designed to be a goal-oriented and/or agent-oriented modeling and analysis tool.

- **OEM**
  - A graph editor to support goal-oriented and/or agent-oriented modeling.

- **REDEPEND-REACT BCN**
  - REDEPEND-REACT is a tool that supports i* modelling and the analysis of the resulting models. This version extends the REDEPEND i* modelling tool. The extension focus on the representation of the information system using the i* framework and provides specific functionalities for the generation and evaluation of alternative architectures for the modelled information system.

- **TAOMIE**
  - TAOMIE supports a model-driven, agent oriented software development and, in particular, the Tropos methodology. It has been designed taking into account Model Driven Architecture (MDA) recommendations.

- **GR-Tool**
  - Forward and backward reasoning is supported in Tropos by a Goal Reasoning Tool (GR-Tool). Basically, the GR-Tool is graphical tool in which it is possible to draw the goal models and run the algorithms and tools for forward and backward reasoning. The algorithms for the forward reasoning have been fully developed in Java and are embedded in the GR-Tool.

- **T-Tool**
  - T-Tool provides a framework for the effective use of formal methods in the early requirements phase. The framework allows for the formal and mechanized analysis of early requirements specifications expressed in a formal modelling language.

- **ST-Tool**
  - ST-Tool, the Secure Tropos tool, is a graphical tool where it is possible to draw Secure Tropos models and to perform the effective formal analysis of Secure Tropos specifications. The tool is written in Java with the swing components, and uses XML as its document format. Formal analysis is based on logic programming. ST-Tool allows to different systems based on Datalog to analyze Secure Tropos specification.

- **J-PRIM**
  - J-PRIM is a tool in Java that supports PRIM, a methodology that addresses i* modelling and analysis from a Process Reengineering point of view. J-PRIM allows to analyse an existing information system and to represent it as a hierarchy of i* elements. Once modelled, several alternatives for the system as can be explored, each of one modelled as a different i* model. All the generated alternatives can be evaluated by defining and applying metrics over the i* models in order to establish which is the most appropriate for the system to be.

- **JUCMNav**
  - JUCMNav is a graphical tool in Java that supports UCM and UCM supplements. JUCMNav is based on OCL-based UCM analysis tools. JUCMNav allows to edit several levels of the UCM metamodel and provides editors for both notations, links between both views, analysis capabilities (including GRL model evaluations), and various import and export formats.
## Detailed Information

<table>
<thead>
<tr>
<th><strong>Main Purpose of the tool:</strong></th>
<th>OpenOME</th>
<th>OME</th>
<th>REDEPEND</th>
<th>REACT-BCN</th>
<th>TAOM4E</th>
<th>GR-Tool</th>
<th>T-Tool</th>
<th>ST-Tool</th>
<th>J-PRIM</th>
<th>jUCHNav</th>
<th>Snet Tool</th>
<th>DesCARTES</th>
<th>VISIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>I* modelling</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Analysis or Evaluation of models</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support the TROPOS Methodology</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Early requirements analysis using KAOS and Formal Tropos</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Formal analysis of functional and security requirements</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>User Requirements Notation (URN) = UCM + GRL</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Simulation of networks scenarios</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support the PRIM methodology</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>NFR, UML and AUML modelling</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Support the I-Tropos methodology</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Provide forward engineering capabilities</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Provide an integrated software project module</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><em><em>I</em> framework supported:</em>*</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Yu95</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>TROPOS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Secure Tropos</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Iterative Tropos</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>GRL</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Customizable</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Availability of the tool:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The tool is available for use</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>The tool is available for development</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Programming Language</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>VBA (Visual Basic for Applications)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Datalog</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Microsoft proprietary</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Platform requirements:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>SunOS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>MacOS</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Other technology needed</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java Runtime Environment</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>VISIO 2003</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Datalog Solver</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>GraphViz® for automatic layout</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>OME</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Eclipse Prolog</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Eclipse with GEF and EMF</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>MyGML compiler</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
4.1.1.5 Do not include an Actor within another Actor

This Guideline Wiki Page displays the guideline as per the i* Style of the University of Toronto. Use Comment tab above to read or write comments about this guideline. Scroll down to see variations of this guideline for other i* modeling styles.

4.1.1.5 Guideline (Beginner,Concept) Do not include an Actor within another Actor.

Discussion: Actors are active and autonomous entities that should be modeled separately. "Sub-system" in the illustration can be modeled as actors that have Dependency Links with the main system and/or other actors. They can also be modeled with Association Links such as “is-part-of” and “ISA” to the higher-level system.

Any Actor type can not be included within another Actor’s boundaries using any type of Link. Internal Actors need to be drawn outside another Actors. Agents, Roles, or Positions with the proper Actor Association Link between them such as ISA, is Part Of, INS, Plays, Covers, or Occupies.
Usage Guidelines

• The guidelines are intended to be flexible recommendations, serving as a catalyst for feedback and future development.

• Beyond the main list of guidelines, there are individual wiki pages for all the guidelines. These pages are intended to allow:
  – Comments and suggestions on individual guidelines.
  – Research groups to post their own variations to the individual guidelines.

• The intention is to compile a list of i* syntax variations. This will help to:
  – Make it easier to understand each other’s work
  – Understand the motivations behind differing i* syntax

• Please contact us if you would like to post your syntax variations to the individual pages of the i* Guide, we will provide you with the correct page access.
How to Contribute

• The success of the i* wiki depends on having active members!
• Although many of the pages can be viewed without registration, any researcher, practitioner or student can become a contributing member of the i* wiki.
• Simply send an email request to istarwiki@i5.informatik.rwth-aachen.de with your name, affiliation, email address and how you have learned about the site.
• If you want to add your publications, your tools, or participate in any of the sections, just register.
• We would like everyone to contribute.

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Goal and Agent-oriented Modeling with i*

Conclusions
Some Ongoing work

4th i* Workshop - Hammamet, Tunisia (June, 2010)

- **Using i* to Support a Summative Evaluation** - James Lockerbie, Neil Maiden, Amir Dotan, Valentina Lichtner
- **Using intentional actor modeling to support the evolution of enterprise software architectures in organizations** - Daniel Gross, Eric Yu
- **Improving the Modularity of i* Models** - Fernanda Alencar, Márcia Lucena, Carla Silva, Emanuel Santos Jaelson Castro
- **Itemized Strategic Dependency: a Variant of the i* SD Model to Facilitate Knowledge Elicitation** - Hesam Esfahani, Eric Yu, Maria Carmela Annosi
- **Deriving Adaptive Behaviour from i* models** - Kristopher Welsh, Pete Sawyer
- **From Adaptive Systems Design to Autonomous Agent Design** - Alexei Lapouchnian, Yves Lespérance
- **Exploring risk-awareness in i* models** - Constantinos Giannoulis, Jelena Zdravkovic
- **A Framework for Iterative, Interactive Analysis of Agent-Goal Models in Early Requirements Engineering** – Jennifer Horkoff, Eric Yu

Jaelson Castro, Fernanda Alencar, Ricardo Ramos
Some Ongoing work

4th i* Workshop - Hammamet, Tunisia (June, 2010)

- **Requirements Engineering for Control Systems** - Dominik Schmitz, Hans W. Nissen, Matthias Jarke, Thomas Rose
- **A bit of "persona", a bit of "goal", a bit of "process" ... a recipe for analyzing user intensive software systems** - Chiara Di Francescomarino, Chiara Leonardi, Alessandro Marchetto, Cu D. Nguyen, Nauman A. Qureshi, Luca Sabatucci, Anna Perini, Angelo Susi, Paolo Tonella, Massimo Zancanaro
- **i* on ADOxx®: A Case Study** - Margit Schwab - Dimitris Karagiannis, Alexander Bergmayr
- **Using i* and Tropos in a Software Engineering Contest: Lessons Learnt and Some Key Challenges** - João Pimentel, Emanuel Santos, Bárbara Santos, Clarissa Borba, Josias Paes, Carlos Lima, Andr Bezerra, Jaelson Castro, Fernanda Alencar, Carla Silva, Ricardo Ramos, Marcia Lucena
- **From Business Services to Web Services: an MDA Approach** - Hugo Estrada, Itzel Morales-Ramírez, Alicia Martínez, Oscar Pastor
- **Bridging the Gap between Goals, Agents and Business Processes** - Renata Guizzardi, Giancarlo Guizzardi, João Almeida, Evellin C. Cardoso
- **A goal-oriented approach for workflow monitoring** - Alicia Martínez, Nimrod Gonzalez, Hugo Estrada
- **Strategy Representation Using an i*-like Notation** - Lam-Son Lè, Bingyu Zhang, Aditya Ghose
- **Using i* Meta Modeling for Verifying i* Models** - Antonio Oliveira, Julio Leite, Luiz Cysneiros
- **From i* to OO-Method: Problems and Solutions** - Fernanda Alencar, Beatriz Marín, Giovanni Giachetti, Emanuel Santos, Oscar Pastor, Jaelson Castro, Xavier Franch
- **Definition and Uses of the i* Metamodel** - Carlos Cares, Xavier Franch, Lidia Lopez, Jordi Marco
- **On temporally annotating goal models** - Sotirios Liaskos, John Mylopoulos
Tools

- Canada (UoT)
  - OME, OpenOME
- Italy
  - TAOM4E, GR Tool, T Tool, ST Tool
- Spain & England
  - REDEPEND, REACT
- Spain
  - GR-Tool, J-PRiM
- Germany
  - Snet Tool
- Brazil
  - Istar Tool, xGOOD, GOOSE
- Belgium
  - DesCARTES
Extensions & variations

• Goal Requirements Language  GRL (Yu et al.)  
  – ITU-T  Z.151 URN- User Requirements Notation
• Tropos  [Mylopoulos et al.]
• Normative i*  
  – Combining law compliancy with intentional Modelling  
    [A. Siena et al. FBK-IRST]
• Tool-supported development of self-adaptive sw agents  
  – From design-time goal models to agents goal directed behaviour  
    [M. Morandini, N. Qureshi et al. FBK-IRST]
• TAOM4E with extensions available at  
  http://se.fbk.eu/tools
Extensions & variations

- Aspectual i-star [Alencar et al.]
- A service-oriented approach for the i* framework [Estrada et al.]
- Defining Inheritance in i* [Lopez et al.]
- ERi*c - Intentional Requirements [Oliveira et al.]
- Conceptual Schemas Generation from Organizational Models [Martinez et al.]
- Requirements Elicitation using an approach that integrates social cultural analysis and organization modeling [Cruz Neto et al.]
- etc
Research issues
Research Issues

- Requirements Engineering
- Software Development
- Social Modeling
- Services
- Security & Privacy

- Tools & Environments
  - Visualization
  - Scalability
- Empirical Studies
- Process & Methods
- Concepts
  - Timing
  - Ontology, etc
- Education
- Standards
- Integration & Interoperability
References

• **i*Wiki**: http://istar.rwth-aachen.de/tiki-index.php


References


Obrigado
Thanks
Grazie
Mile
Gracias!

CONTACT AND INFORMATION:

Jaelson Castro  jbc@cin.ufpe.br
Fernanda Alencar  fernandaalenc@gmail.com
Ricardo Argenton  ricargentonramos@gmail.com