

Course Topics

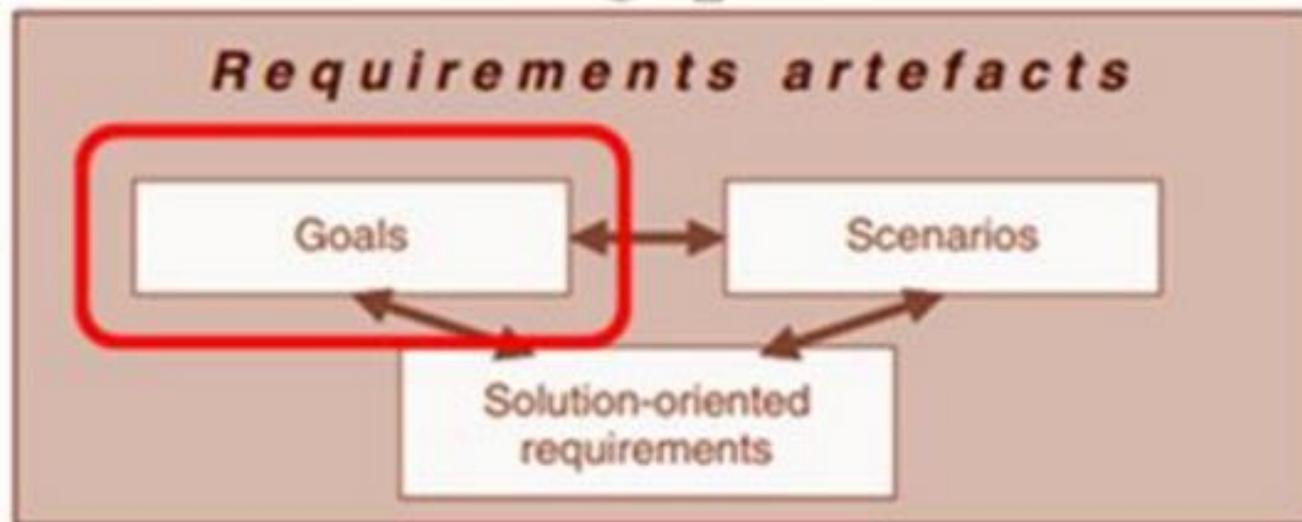
- ~~• Why Requirements Engineering?~~
- ~~• Introduction to Requirements~~
- ~~• RE in Software Development Life Cycles~~
- ~~• System Vision, Context, and RE Framework~~
- Fundamentals of Goal Orientation
- Fundamentals of Scenarios
- Requirements Discovery
- User Stories and Agile Estimation
- Features Prioritization
- Requirements Negotiation
- Requirements Validation
- Fundamentals of Requirements Management

Objectives



- Fundamentals of goal modeling in requirements engineering
 - Basic concepts of documenting goals
 - The Goal-oriented Requirements language (GRL)
 - i* Language
- 

The three kinds of requirements artifacts



Goals



- Goals are **high-level objectives of the business, organization, or system.**

[Anton 1996]

- A goal is an **objective** the system under consideration should achieve.

[Van Lamsweerde 2001]

- A goal is an intention with regard to the objectives, properties, or use of the system.

[Klaus Phol 2010]

- Goals have a prescriptive nature, i.e. a goal states **what is expected or required from the system.**
- 

Example of Goals



Goals for a **car navigation system** :

G1: The system shall guide the driver to a desired destination automatically.

G2: The response times of the system shall be 20% lower compared with the predecessor system.



Motivation (1)

- *Better understanding of the system:*

Goals refine the overall system vision

- *Requirements elicitation:*

Goals drive and guide the elicitation of requirements.

For instance, for each goal, **a set of requirements** can be defined which must be fulfilled to satisfy the goal.

- For each goal, **scenarios** can be defined to define typical interaction sequences which lead to **goal satisfaction**.
- Defining scenarios in which a goal is not satisfied also contributes to a better understanding of the goal and supports requirements elicitation.

Motivation (2)

■ *Identification and evaluation of alternative realizations:*

Typically, several possibilities exist to satisfy a goal.

- By decomposing goals into sub-goals, alternative realizations can be identified systematically.

■ **Detection of irrelevant requirements:** The explicit consideration of goals supports the identification of irrelevant requirements.

- The stakeholders check for each requirement **whether the requirement contributes to the satisfaction of a goal or not.**
- If a requirement **does not support the satisfaction of any defined goal**, either the requirement is **irrelevant** for the system or the defined **goals are incomplete.**

Motivation (3)

- *Justification of requirements:* If a requirement contributes to the satisfaction of a goal, the goal documents a rationale for defining the requirement.
- *Completeness of requirements specifications:* With respect to the defined goals, a requirements specification is **complete** if, **by implementing the defined requirements, all goals can be satisfied.**
- *Identification and resolution of conflicts:* Quite often, the origins of conflicting requirements are different stakeholder intentions. Hence, conflict resolution should, at first, **focus on resolving conflicting goals.**
- *Stability of goals:* Goals often remains **unchanged.** Therefore, in comparison with functional or quality requirements, goal models are more stable.

AND/OR Goal Decomposition

- Goals can form a **decomposition graph** in which **child nodes refine parent node**.
- Root node of that graph is actually **system vision**, that can be considered as **top-level goal**.
- **Two kinds** of goal decomposition:
 - **AND-decomposition** – The decomposition of a super-goal G into a set of sub-goals G_1, \dots, G_n with $n \geq 2$ is an AND-decomposition **if and only if all sub-goals G_1, \dots, G_n must be satisfied in order to satisfy the super-goal G** .
 - **OR-decomposition** – The decomposition of a super-goal G into a set of sub-goals G_1, \dots, G_n with $n \geq 2$ is an OR-decomposition **if and only if satisfying one of the sub-goals G_1, \dots, G_n is sufficient for satisfying the super-goal G** .

Goal Decomposition Example

- AND-decomposition of the goal “**Navigation system must provide comfortable and fast navigation to the destination**”:
 - **G1:** Easy entry of the destination.
 - **G2:** Automatic routing according to user-specific parameters.
 - **G3:** Displaying of traffic jams and automatic re-routing to avoid traffic jams.

- OR-decomposition of the goal “**Navigation system must have the ability to localize the position of the car**”:
 - **G1:** Localization of the car via cell phone.
 - **G2:** Localization of the car via GPS.

Goal Dependencies



- Goals can have the following types of **dependencies** between each other:
 - **Requires**
 - **Support**
 - **Obstruction**
 - **Conflict**
 - **Equivalence**

Goal Dependencies: Requires

- G1 **requires** G2 if **the satisfaction of G2 is a prerequisite for satisfying G1**
- However, the “requires” dependency **does not imply that G2 is a sub-goal of G1.**
 - “Requires” dependency can exist between goals that are not in a decomposition relationship with each other.

G1: The system shall navigate the driver around traffic congestion.

G2: The system shall be able to receive traffic messages.

G1 requires G2

Goal Dependencies: Support

- G1 **supports** G2 if **the satisfaction of G1 contributes positively to satisfying G2**

G1: The navigation system shall be able to download electronic maps on demand.

G2: The system shall allow simple entry to the destination for navigation.

G1 supports G2

Explanation: If a destination is outside the maps that are available to the navigation system, **the goal “simple entry of destination” cannot be satisfied**. However, as expressed by the goal G2, the system has the facility to download the needed electronic maps and then allow the driver to select the destination in the navigation system. Thus, the goal “download map” **supports** the goal “simple entry of destination”.

Note on Support Dependency



- AND- or OR-decomposition implicitly represents a special type of "support" dependency.
 - If, for example, G2 is a sub-goal of G1 and G2 is related to G1 by an **AND-decomposition**, the satisfaction of G2 **partially supports** satisfying G1.
 - If G2 is related to G1 by means of an **OR-decomposition**, G1 is satisfied whenever G2 is satisfied. Hence G2 **strongly supports** G1.
- 

Goal Dependencies: Obstruction

- G1 **obstructs** G2 if satisfying G1 hinders the satisfaction of G2
- An “obstruction” dependency can be understood as the **opposite of a goal support dependency**.
- An “obstruction” dependency cannot exist between goals that are part of an AND-decomposition

G1: the navigation system shall be able to download electronic maps via the GSM network on demand.

G2: The data traffic over the GSM network caused by the navigation system shall be as low as possible.

“Obstruction” Dependency: G1 interferes with G2

Satisfying the goal G 1 causes high data traffic and thus hinders the satisfaction of the goal G2 “The data traffic shall be as low as possible”.

Goal Dependencies: Conflict

- A **conflict** between G1 and G2 exists if:
 - **Satisfying G1 excludes satisfying of G2 and**
 - **Satisfying G2 excludes satisfying of G1**
- A “conflict” dependency documents a very strong obstruction and is, in addition, symmetric.

G1: It shall be possible to localize the car via GPS.

G2: The country-specific privacy laws shall be observed.

G1 and G2 are conflicting

If a stakeholder requires that a car can be localized via GPS, yet the privacy laws of a country forbid the localization of vehicles. In this case, the goal of some stakeholder and the law of the country are clearly in conflict. Satisfying one of the two goals makes the satisfaction of the other goal impossible.

Goal Dependencies: Equivalence

- Two goals G1 and G2 are **equivalent** (with respect to the goal satisfaction) if:
 - Satisfying G1 leads to the satisfaction of the G2**and**
 - Satisfying G2 leads to the satisfaction of the G1

G1: The system shall comply with the car safety regulations of country A.

G2: The system shall comply with the car safety regulation of country B.

- If the car safety regulations in country A are identical to the regulations in country B, the two goals are equivalent (with respect to goal satisfaction). Satisfying the goal G1 implies the satisfaction of the goal G2 and vice versa.
- The example illustrates that a **goal equivalence relationship** does not require that the **two goal definitions be identical**, i.e., **goal equivalence should not be confused with the equality of goal definitions.**

Identifying Goal Dependencies

- **Context changes** affect goal dependencies
- **Example:**
Change of a data protection law in a country may prohibit the electronic localization of a car
- Stakeholders must be aware of such changes and constantly analyze their influences!

Document Goals

- It is **very important** to document goals properly.
- The effort required to document goals in requirements engineering is, compared with the advantages gained, rather low.
- Goals can be documented:
 - **Using unstructured natural language.**
 - **Using templates (structured)**
 - **Using dedicated goal modeling languages.**
- Each approach has its positive and negative sides.

Documenting Goals using unstructured natural Language

- **Unstructured approach** implies **specifying goals one after the other in free text, without any specific rules.**

- **Example:**

G: Comfortable and fast navigation to the destination.

The goal G is refined into the following three sub-goals (AND-decomposition):

GI: Easy entry of the destination

G2: Automatic routing according to user-specific parameters

G3: Displaying of traffic jams and automatic re-routing to avoid traffic jams

Documenting Goals using templates

- Template-based documentation of goals **offers significant advantages**. It comprises the following types of attributes:
 - Attributes for uniquely **identifying** goals.
 - **Management** attributes.
 - Attributes for documenting references to the **context**.
 - Specific goal attributes, i.e. the goal level, the description of the goal, dependencies to other goals, as well as relationships to scenarios.
 - An attribute for documenting any type of additional information

Template for Documenting Goals

	No.	Section	Content/Explanation
Goal Identification	1	Identifier	Unique identifier of the goal
	2	Name	Unique name for the goal
Managing Attributes	3	Authors	Names of the authors who have documented the goal
	4	Version	Current version number of the documentation of the goal
	5	Change history	List of the changes applied to the documentation of the goal
	6	Priority	Importance of the documented goal
	7	Criticality	Criticality of the goal, e.g. for the overall success of the system
documenting references to the context	8	Source	Name of the source from which the goal originates
	9	Responsible stakeholder	Name of the stakeholder who is responsible for the goal
	10	Using stakeholders	Stakeholders who benefit from the satisfaction of the goal

Template for Documenting Goals (Cont.)

	No.	Section	Content/Explanation
Specific goal attributes	11	Goal level	Identifier for the abstraction level at which the goal is defined
	12	Goal description	Description of the goal
	13	Super-goal	Reference to the super-goal including the type of decomposition
	14	Sub-goals	References to the sub-goals including the type of decomposition
	15	Other goal dependencies	Further dependencies with other goals such as <i>requires</i> , <i>conflict</i> , etc.
	16	Associated scenarios	References to scenarios that describe the (dis)satisfaction of the goal
additional information	17	Supplementary information	Additional information about this goal

Example of a Template

No.	Section	Content/Explanation
1	Identifier	G-2-17
2	Name	Automatic navigation
3	Authors	Peter Miller, Dan Smith
4	Version	V1.2
5	Change history	V1.0 12.01.2009 Dan Smith V1.1 14.02.2009 Peter Miller
6	Priority	High
7	Criticality	Medium
8	Source	William Garland (product manager)
9	Responsible stakeholder	Peter Miller
10	Using stakeholders	Driver of the car

Example of a Template

No.	Section	Content/Explanation
11	Goal level	System level
12	Goal description	The system shall automatically direct the driver to the desired destination.
13	Super-goal	G-2-2: Comfortable and fast navigation to the destination
14	Sub-goals	G-2-25: Localization of the car via GPS G-2-26: Download of electronic maps on demand
15	Other goal dependencies	Conflict with G-1-45: Reduce costs for cars Support of G-1-37: Technological leadership in the automotive segment of medium-sized vehicles
16	Associated scenarios	S-2-34: Navigate to destination
17	Supplementary information	The competing system SX-23-44 realizes this goal.

Systematic Elicitation of Goals and Goal attributes

- Try to elicit all **relevant goals first**
- Avoid capturing all goal attributes right at the beginning
- When defining attributes for a goal, define the **basic attributes** (identifier, name, source, responsible stakeholder, goal description) **first**. Subsequently, define the attributes **super-goal** and **sub-goals** for each goal
- Validate whether the elicited goals are **complete** and the documented goal relationships are **correct**
- **Complement missing goals and missing goal relationships** and, if required, **revise** the defined goals and goal relationships
- Define **scenarios** in order to **support the elicitation and validation of goals**
- Add missing information in all slots of the goal template

Seven Rules for Documenting Goals



- **Rule 1:** Document goals concisely.
 - **Rule 2:** Use the active voice.
 - **Rule 3:** Document the stakeholder's intention precisely.
 - **Rule 4:** Decompose high-level goals into more concrete sub-goals.
 - **Rule 5:** State the additional value of the goal.
 - **Rule 6:** Document the reasons for introducing a goal.
 - **Rule 7:** Avoid defining unnecessary restrictions.
- 

Dealing with stakeholders demanding a particular solution

- If a stakeholder (such as the client) demands a specific solution or expresses a specific constraint for the realization of the system, apply the following steps **to weaken the restrictions**:
 1. Elicit the actual, solution/constraint-free super-goal that is behind the required solution by asking “**why**” questions.
 2. Try to identify **viable solution alternatives for the super-goal**.
 3. Document the identified, alternative solutions as sub-goals of the solution-free super-goal using an OR-decomposition.

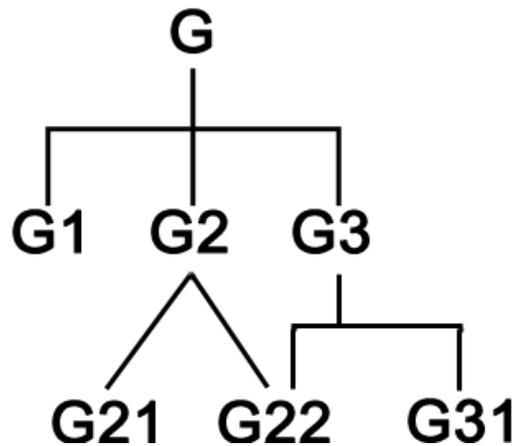
Goal Modeling Languages

Goal model Definition:

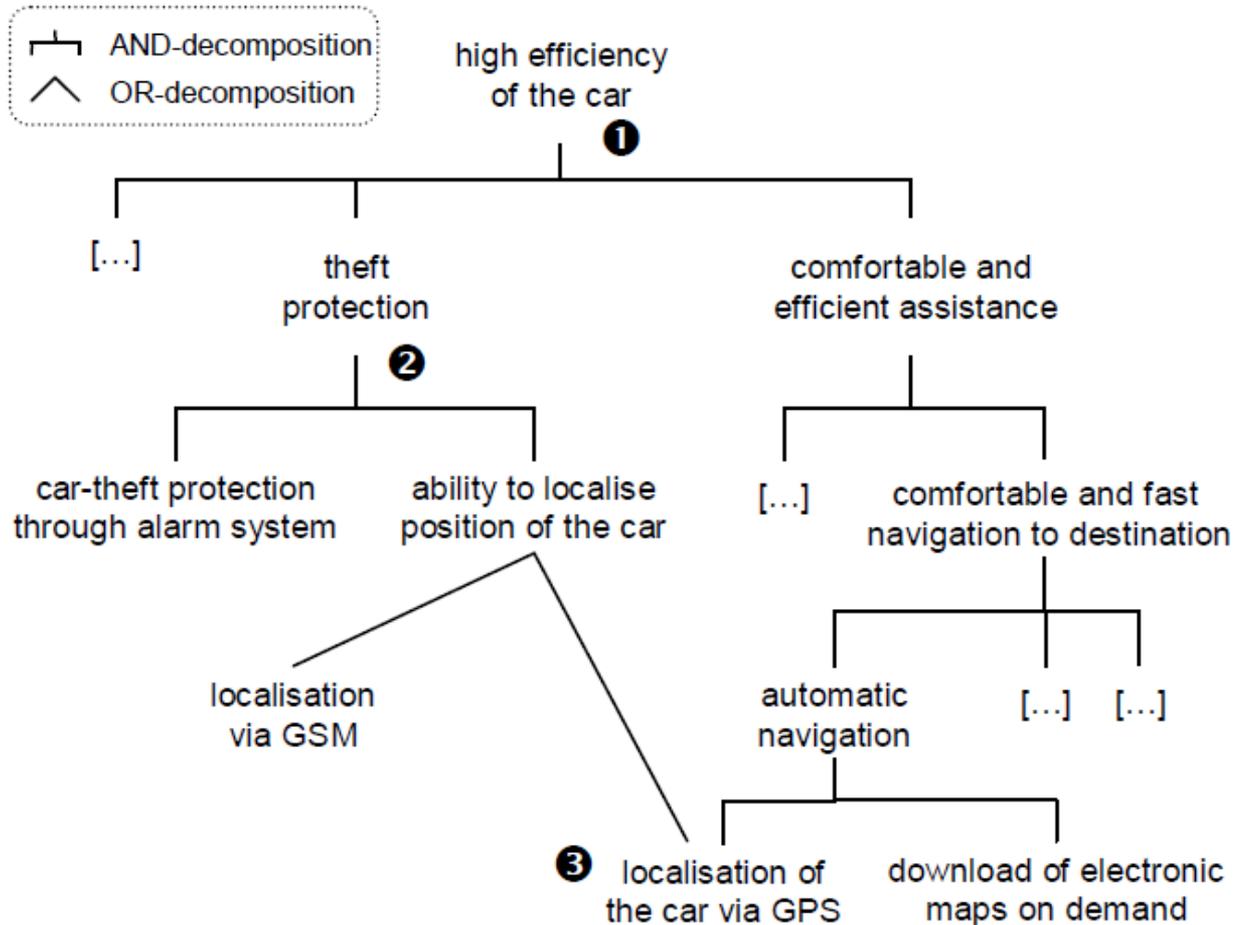
- **A goal model is a conceptual model that documents goals, their decomposition into sub-goals, and existing goal dependencies.**
- **Model-based goal documentation**
 - helps understanding and communicating goals
 - complements template-based documentation
- **Goal modeling method consists of language, rules, guidelines and management practices**
- Common goal modelling languages include different dialects of **AND/OR graphs**, the Goal-oriented Requirements Language (GRL), i* (iStar), TROPOS, and KAOS.

Documenting Goals Using AND/OR Graphs

- **Definition:** An AND/OR goal graph is a **directed, acyclic graph** with **nodes** that represent **goals** and **edges** that represent AND/OR-decomposition relationships between the goals.
- Some sub-goals **contribute to the satisfaction of more than one super goal**

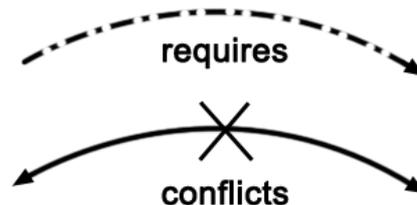


Example of goal modeling using AND/OR Graphs

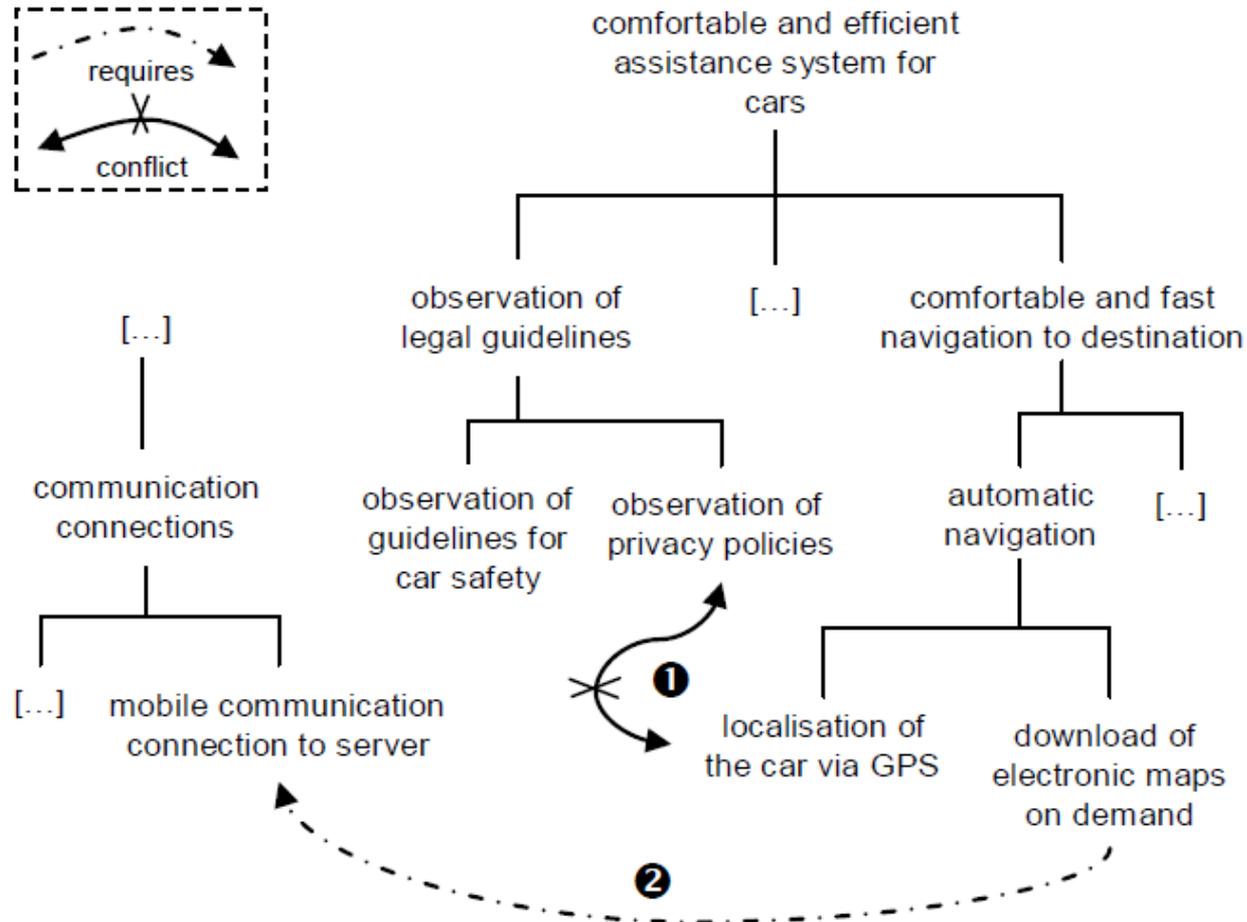


Requires and Conflict dependencies in AND/OR Graphs

- AND/OR graphs can be extended by defining two additional types of edges representing the **requires** and the **conflict** dependencies.
- **Requires** edge directed from goal G1 to goal G2 implies that **to satisfy the goal G1, the goal G2 must be satisfied.**
- **Conflict** edge is **an edge** between two goals G1 and G2 that documents a conflict dependency.



Example of goal modeling using AND/OR Graphs (requires and conflict)





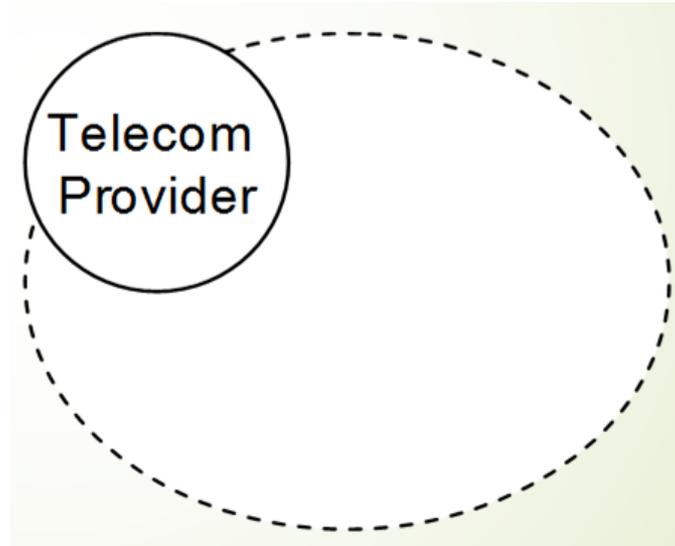
Goal-oriented Requirements Language (GRL)

Goal-oriented Requirements Language (GRL)

- Targets systems/software/requirements engineers
- **Part of URN (User Requirements Notation) language, an ITU-T standard.**
- URN Formalizes and integrates two notations:
 - **Goal-oriented Requirement Language (GRL)**
 - **Use Case Maps (UCMs) for expressing scenarios**
- URN models can be used to specify and analyze various types of (proposed or evolving) reactive systems, business processes, and telecommunications standards

GRL Actors

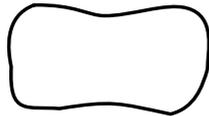
- Holder of intentions (stakeholders)



GRL Intentional Elements



Goal



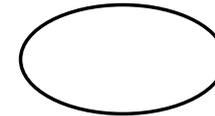
Softgoal



Task



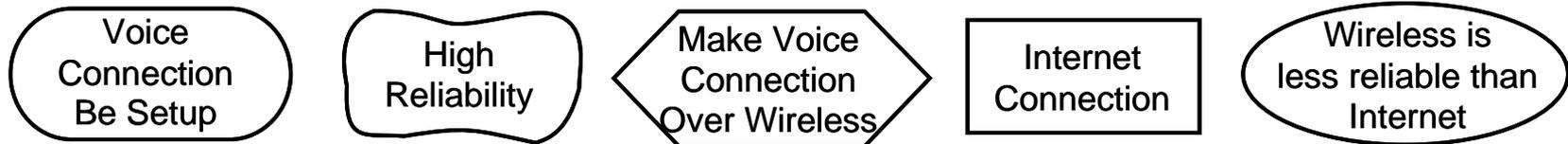
Resource



Belief

- A (hard) **Goal** is a condition or state of affairs in the world that the stakeholders would like to achieve. A goal can be either a business goal or a system goal.
- A **Softgoal** is a condition or state of affairs in the world that the actor would like to achieve, but unlike in the concept of (hard) goal, there are **no clear-cut criteria** for whether the condition is achieved. Softgoals are often **used to describe qualities and non-functional aspects such as security, robustness, performance, usability, etc.**
- A **Task** specifies a particular way of doing something.
- A **Resource** is a physical or informational entity, for which the main concern is **whether it is available.**
- A **Belief** is used to represent design rationale.

Example of GRL Intentional Elements



- “Voice Connection Be Setup” is defined as a **(hard) goal** because this is something that can be achieved entirely.
- “High Reliability” is defined as a **softgoal** because this is something that **can never be entirely achieved (but that can be sufficiently achieved)**.
- “Make Voice Connection Over Wireless” is defined as a **task** because this is a particular way of setting up a connection.
- “Internet Connection” is defined as a **resource** because this is a **physical entity that can be available or not**.
- “Wireless is less reliable than Internet” is defined as a **belief** because this provides a **rationale for some of the design decisions**.

GRL Links



- **Contribution**
Link input to goals/softgoals (in general)
- **Dependency**
Defined between actors (or their intentional elements), with a dependum
- **Decomposition**
Defines what an intentional element needs to be satisfied; e.g., OR, AND.
- **Correlation**
Same as contribution but indicates a side-effect, often across actors

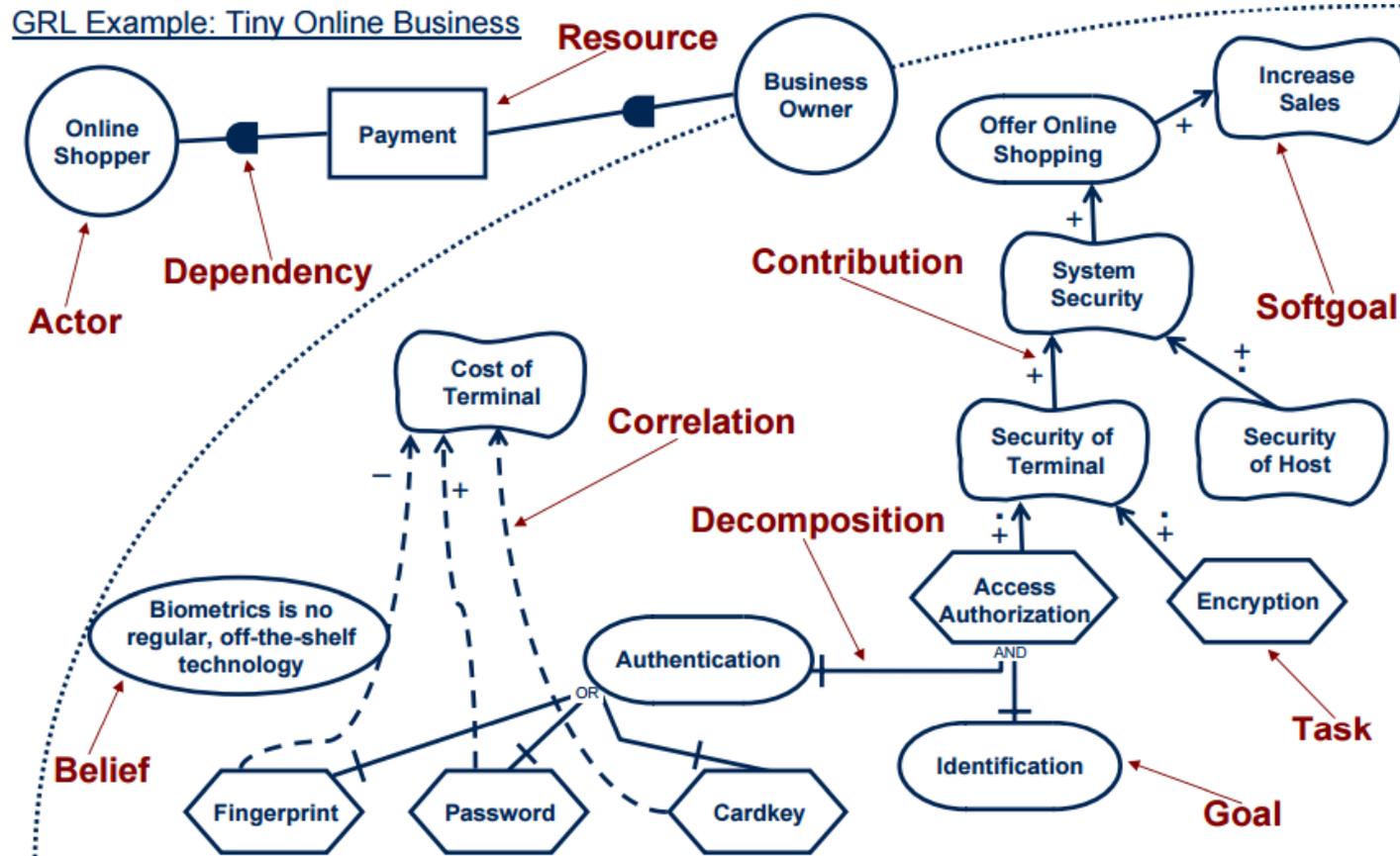
GRL Contribution Types (Qualitative and Quantitative)



- **Make:** The contribution is positive and sufficient.
- **Help:** The contribution is positive but not sufficient.
- **SomePositive:** The contribution is positive, but the extent of the contribution is unknown.
- **Unknown:** There is some contribution, but the extent and the degree (positive or negative) of the contribution is unknown.
- **SomeNegative:** The contribution is negative, but the extent of the contribution is unknown.
- **Hurt:** The contribution is negative but not sufficient.
- **Break:** The contribution of the contributing element is negative and sufficient.

Qualitative Contribution	Quantitative Contribution
Make	100
SomePositive	75
Help	25
Unknown	0
Hurt	-25
SomeNegative	-75
Break	-100

GRL Notation: An Example



GRL Strategies

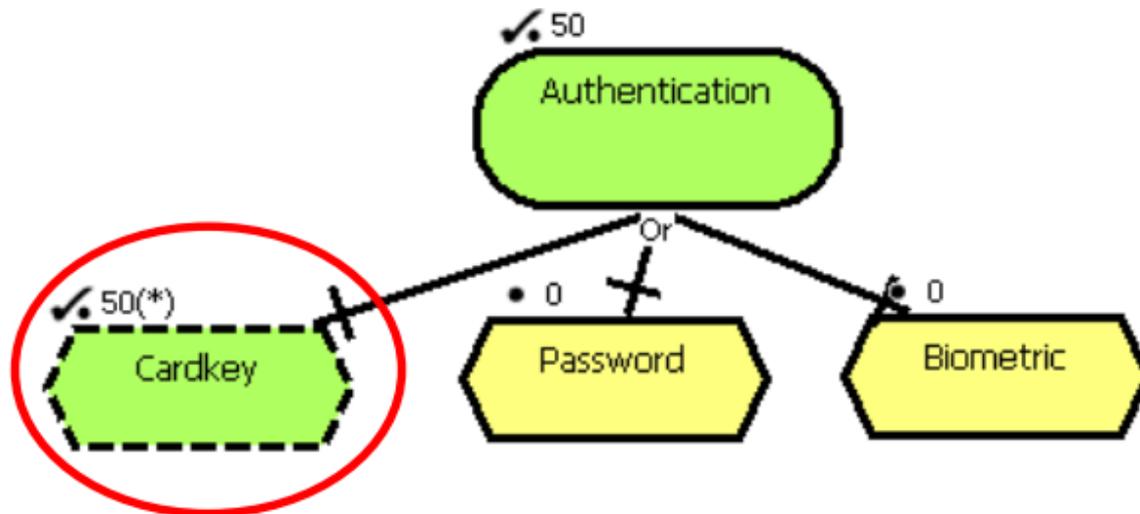
- GRL allows a particular configuration of intentional elements to be defined in a **strategy** (i.e., one possible solution)
 - Captures **the initial, user-defined satisfaction levels** for these elements separately from the GRL graphs
 - Strategies can be compared with each other for trade-off analyses
- **Evaluation mechanism** executes the strategies:
 - Propagating **satisfaction levels** to the other elements and to actors shows **impact** of proposed solution on high level goals for each stakeholder
 - Propagation starts at user-defined satisfaction levels of intentional elements (usually bottom-up)

GRL Satisfaction Qualitative Symbols



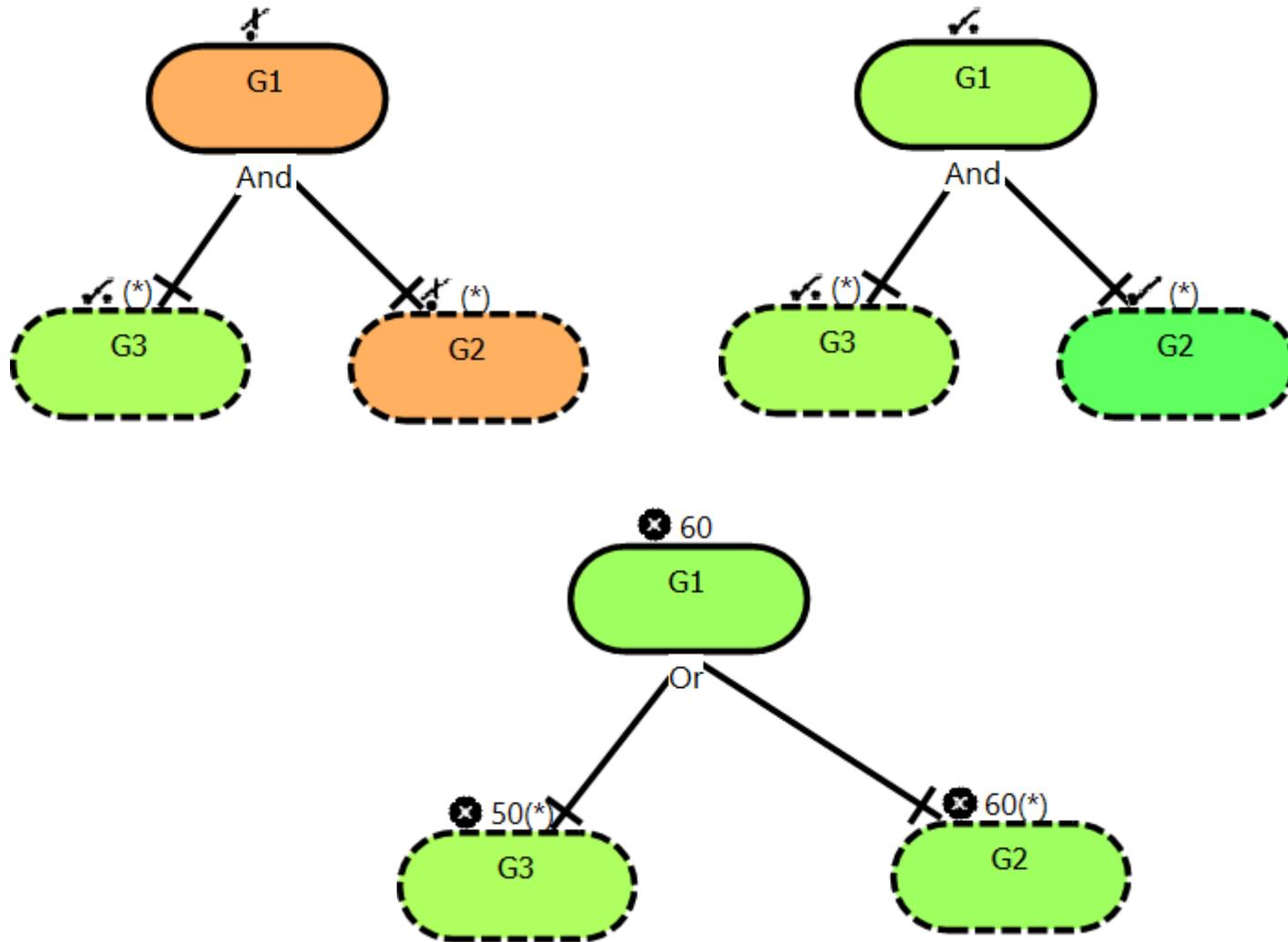
- **Denied:** The intentional element is sufficiently dissatisfied.
- **Weakly Denied:** The intentional element is partially dissatisfied.
- **Weakly Satisfied:** The intentional element is partially satisfied.
- **Satisfied:** The intentional element is sufficiently satisfied.
- **Conflict:** There are arguments strongly in favour and strongly against the satisfaction of the intentional element.
- **Unknown:** The satisfaction level of the intentional element is unknown.
- **None:** The intentional element is neither satisfied nor dissatisfied.

GRL Intentional Elements/links Satisfaction Values

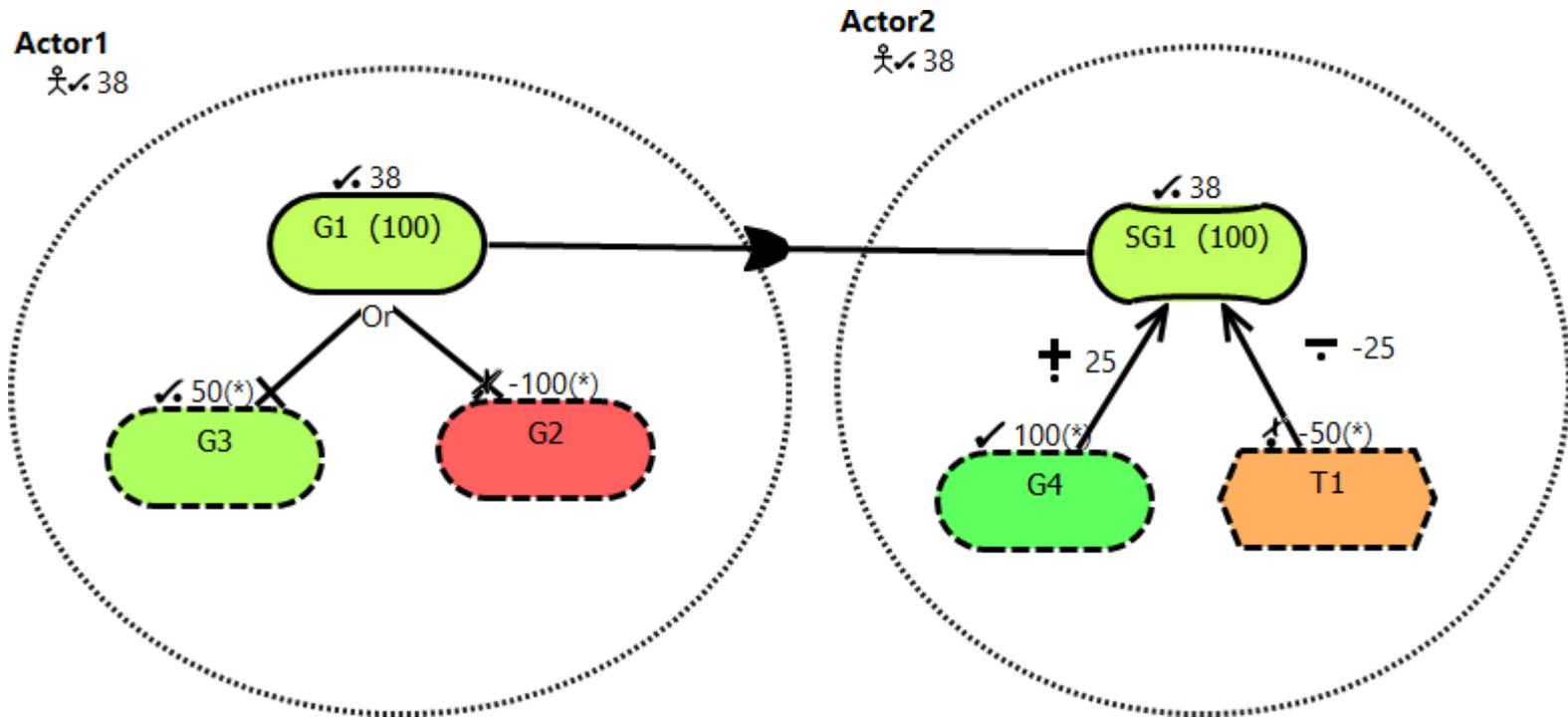


A star (*) indicates an initial value part of a given strategy (element also shown in dashed lines).

GRL Propagation through AND/OR

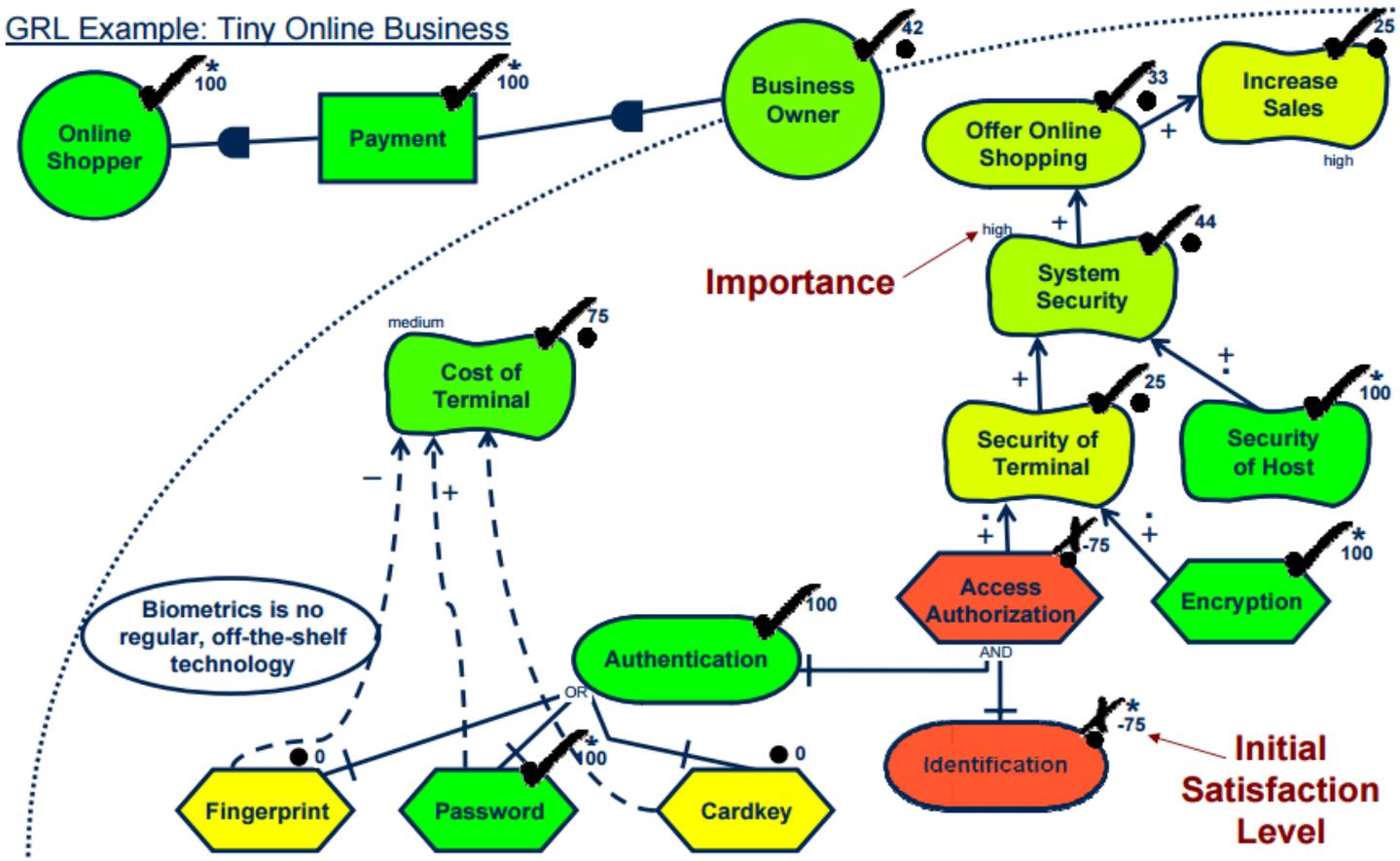


GRL Propagation through Dependencies and Contributions



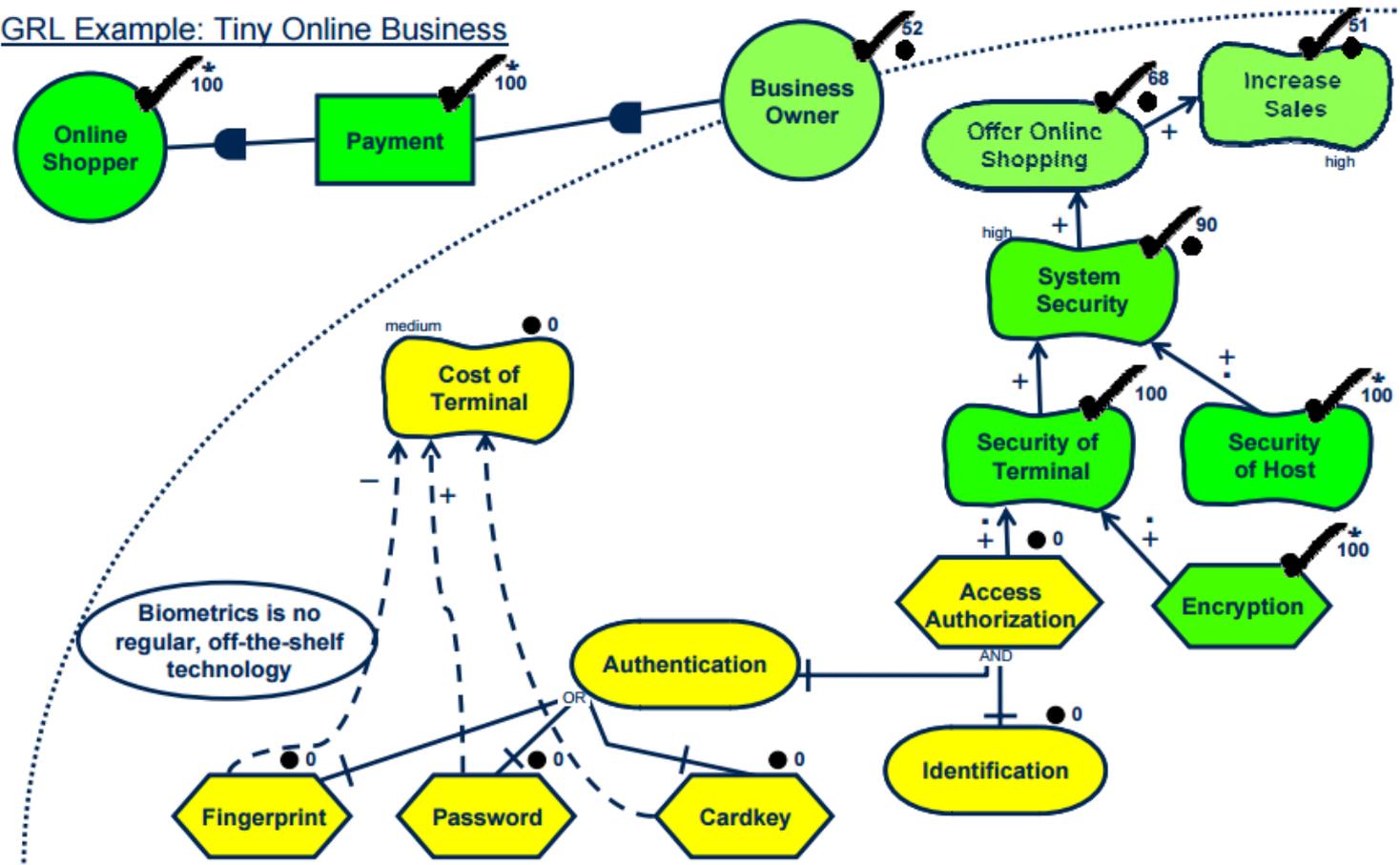
GRL Strategy Execution (Strategy 1)

GRL Example: Tiny Online Business



GRL Strategy Execution (Strategy 3)

GRL Example: Tiny Online Business



jUCMNav tool (URN tool)



- Web site:

<http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/WebHome>

- Installation of the jUCMNav tool within eclipse:

- Select Help -> Install New Software... → Add
- In the field Name write: jUCMNav
- In the field Location write:

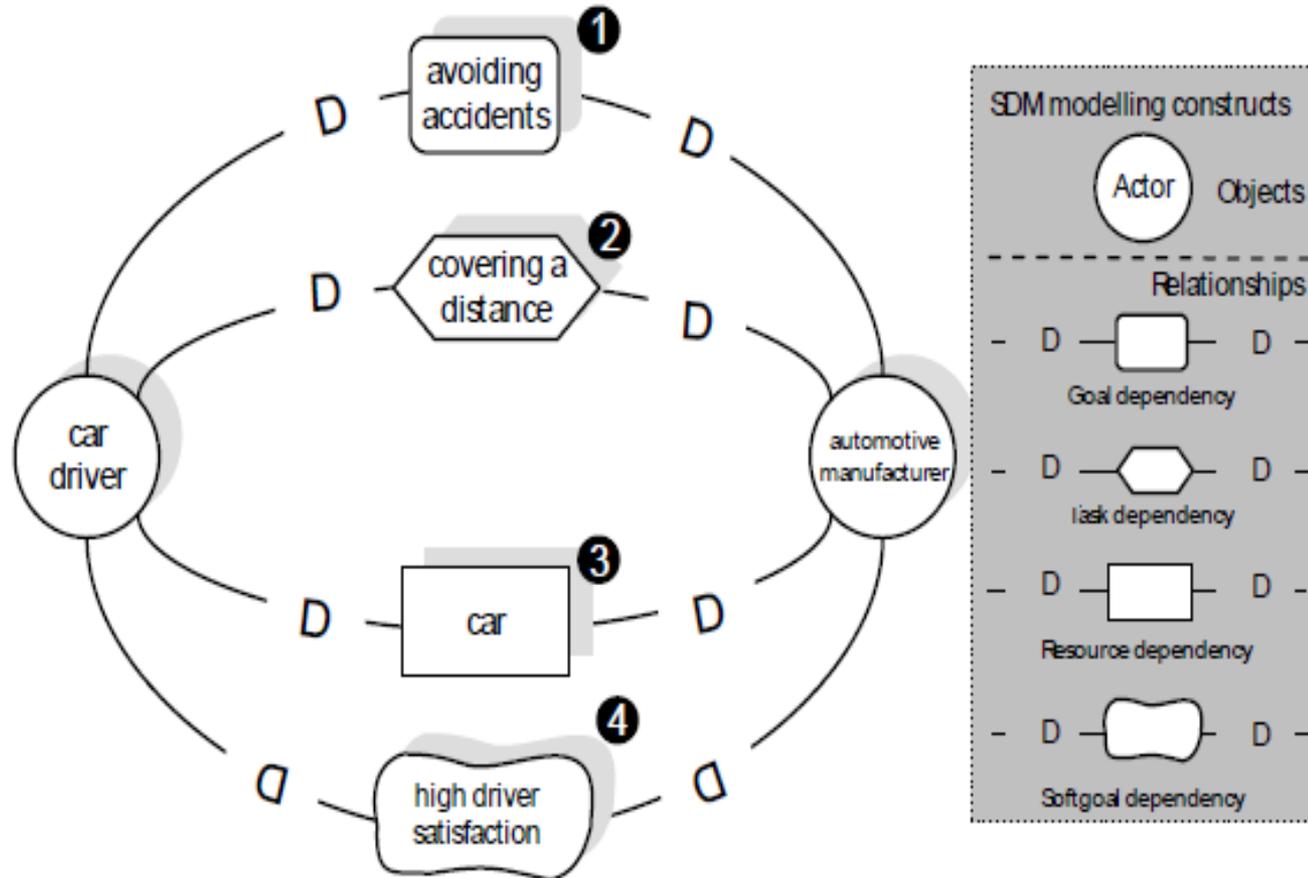
<http://jucmnav.softwareengineering.ca/jucmnav/updatesite/>



iStar (i*) Language

- Latest version: iStar 2.0
- Two kinds of goal models:
 - **Strategic Dependency (SD) Model**
 - Documents dependencies between actors.
 - Documents on which tasks, goals, softgoals, and resources they depend.
 - **Strategic Rationale (SR) Model**
 - Details each actor by defining the actor's internal structure.
 - Provides rationales for the external dependencies.

Example of a strategic dependency model in i*



Example of a SR Model in iStar

