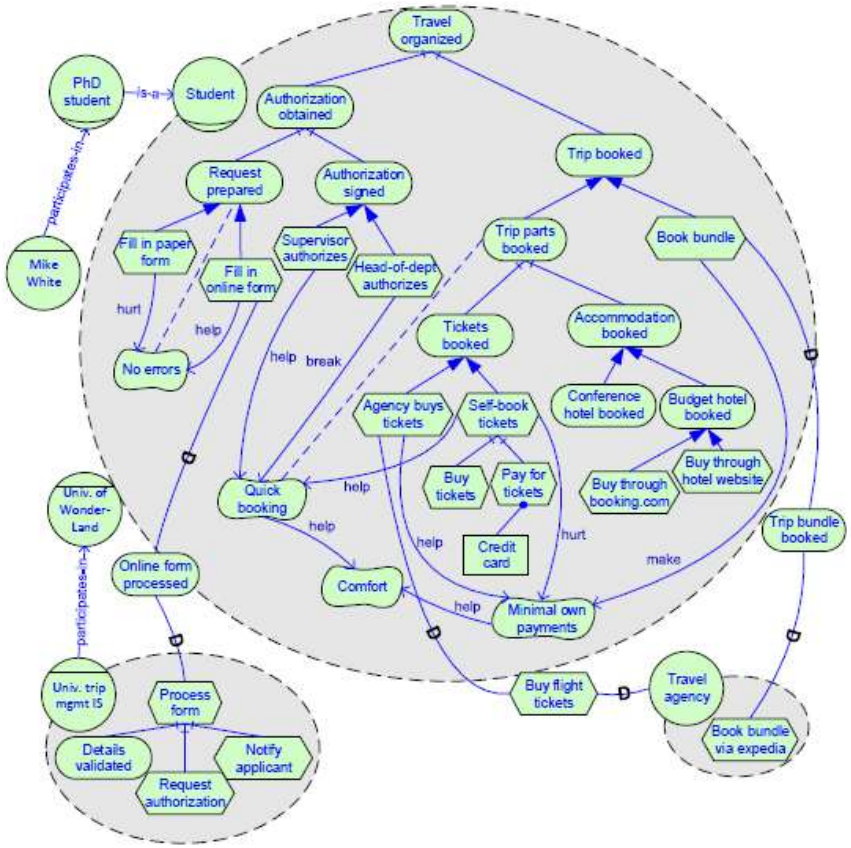
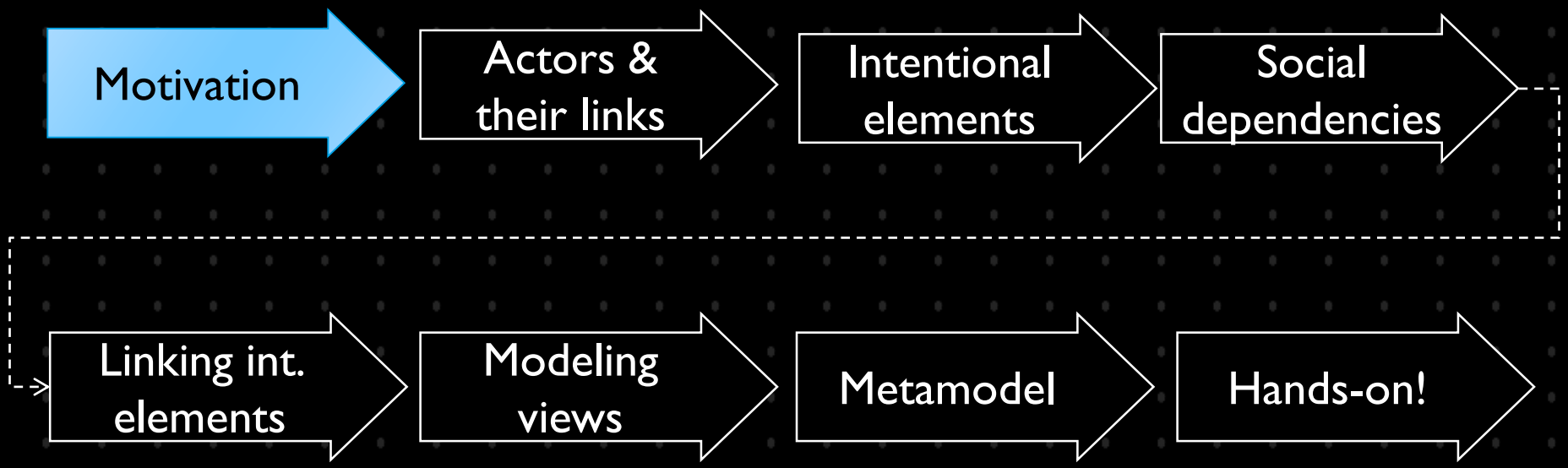


Social modeling of organizations with iStar 2.0

Based on material by Fabiano Dalpiaz

Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

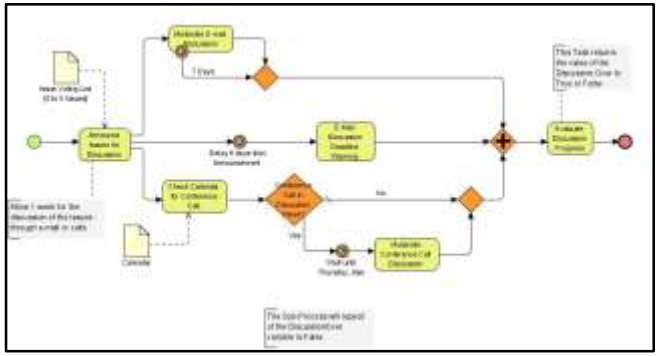
Organizations

- A (business) organization is a social structure with a purpose, e.g., providing services or producing products
- Organizations can be understood as composite systems intended to achieve organizational goals and objectives
- Two basic types of organizations:
  - **Production organizations**: manufacturing, farming, construction and agriculture, software, games
  - **Service organizations**: transportation, communication, banking and finance, medicine, education and retailing, distribution

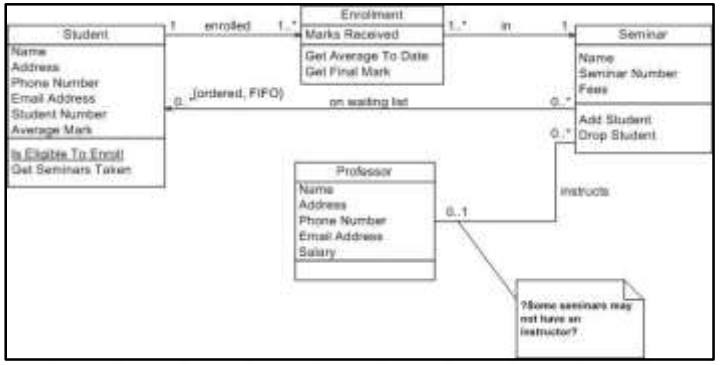
Modeling organizations: why?

- Many reasons exist that justify creating conceptual models of an organization
  - Training of (new) employees
  - Knowledge management
  - Certification and accreditation (e.g., ISO)
  - Re-engineering of / improving the organization
  - Requirements engineering
- The purpose affects the suitability of modeling languages

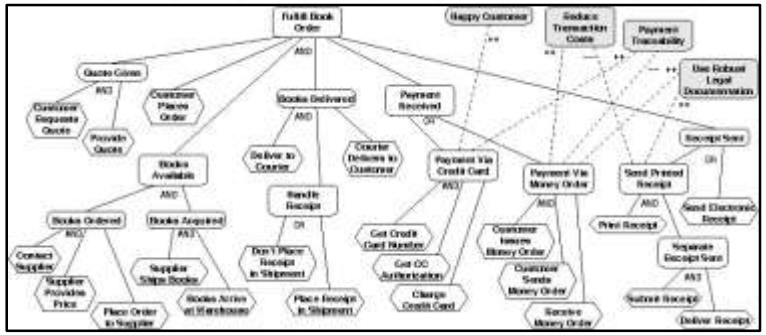
Which modeling language?



How does the org operate?
Business processes



What elements and assets?
Class/ER diagrams

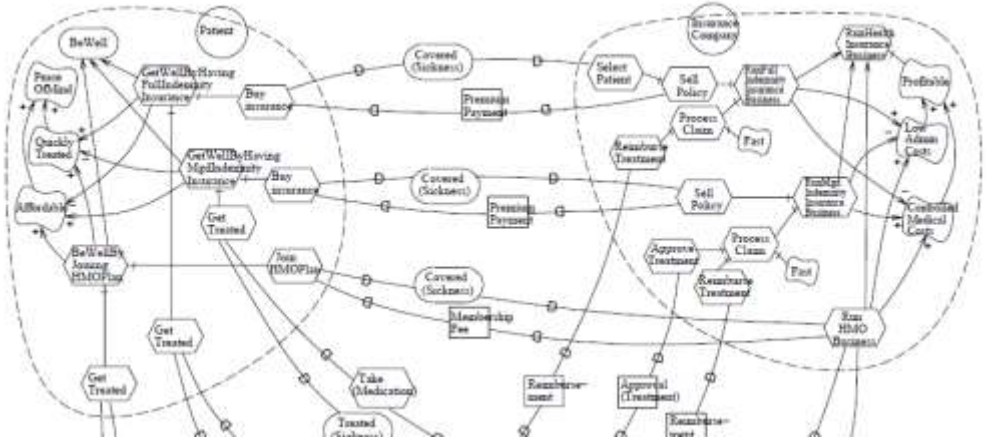


Why do actors act in certain ways?
Goal models

...

The *i** language

- Developed in the mid Nineties [Yu 1995]
- Provides a framework for asking 'why' questions
- Based on the notion of an '**intentional actor**'
- Models are created using two diagrams
  - Strategic Diagrams: social relationships between actors
  - Rationale Diagrams: goals and sub-goals of actors



*i** in 2016: the good, the bad, the ugly

- Quickly adopted by the research community
- Multiple extensions were proposed, e.g., for specific domains such as security, risk, law

But

- Many extensions make it hard for newcomers to learn it
- Practitioners won't adopt it
- Educators will teach their own variant

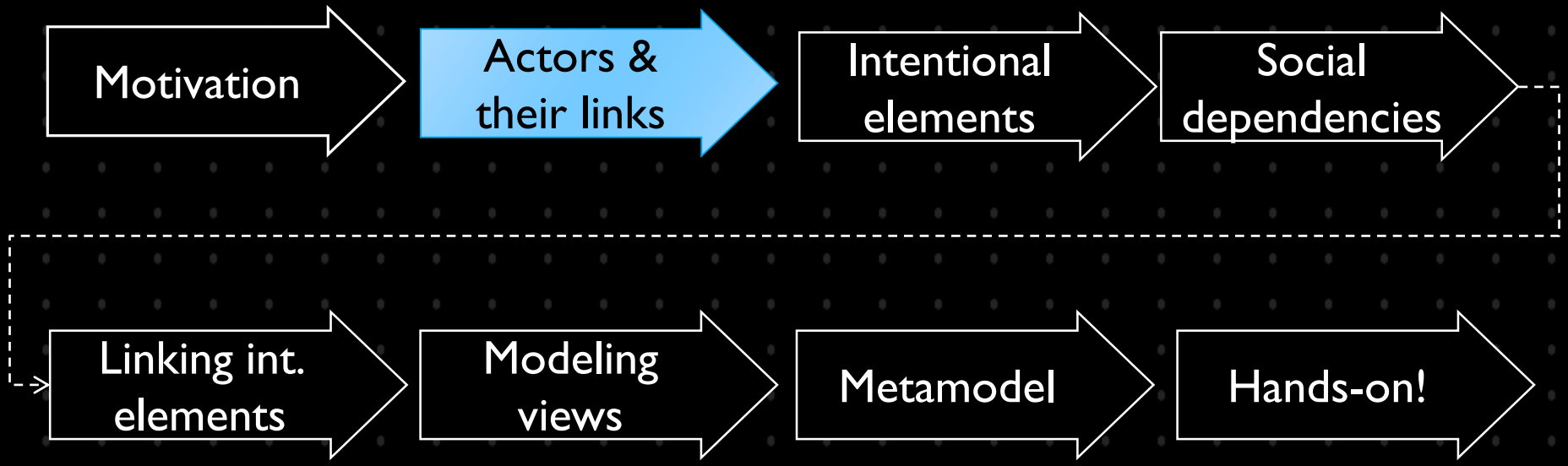
The road to iStar 2.0

- A **community effort** to solve the above-mentioned issues
  - 10/2014: One-day meeting the day before the ER'14 conference in Atlanta
  - 01/2015: Draft of discussions and open questions
  - 06/2015: Community meeting at CAiSE'15 in Stockholm (iStar teaching workshop)
  - 08/2015: First draft (v 0.1)
  - 09/2015: Discussion at the iStar Workshop at with RE'15
  - 10/2015: Dedicated one-day meeting before ER'15 in Stockholm
  - 12/2015: Draft distributed among the community
  - 01/2016: Draft updated (v 0.2)
  - 03/2016: Three authors meet at REFSQ'16 in Gothenburg
  - 03/2016: iStar 2.0 Language Guide, first draft
  - 05/2016: iStar 2.0 Langauge Guide released on arXiv.org (**3 authors, 22 endorsers**)

Running example

- University travel reimbursement
  - **Students** organize trips to conferences
  - They rely on **travel agencies** and the university's **trip management information system**
  - Multiple **alternatives** exist to arrange a trip

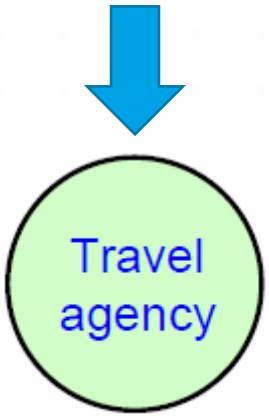
Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Actors

- Organizations are social entities
- Their operation relies on the effective interaction among a number of actors

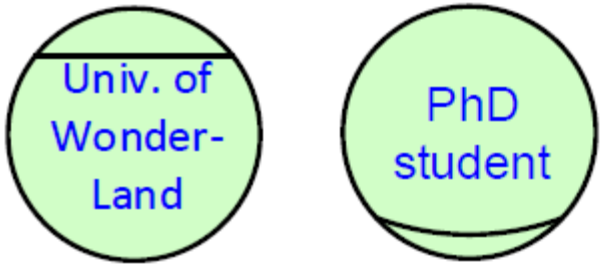
- **Actor:** an active, autonomous entity that aims at achieving its goals by exercising its know-how, in collaboration with other actors



Agents and Roles

- Two types of actors exist in iStar 2.0: agent and role

- **Agent**: an actor with concrete, physical manifestations, such as a human individual, an organization, or a department
- **Role**: an abstract characterization of the behavior of a social actor within some specialized context or domain of endeavor



An agent and a role

Which one should I use?

- Can I identify a concrete individual or (sub)organization?

  → **Agent**

- Do I want to characterize an abstract class?

  → **Role**

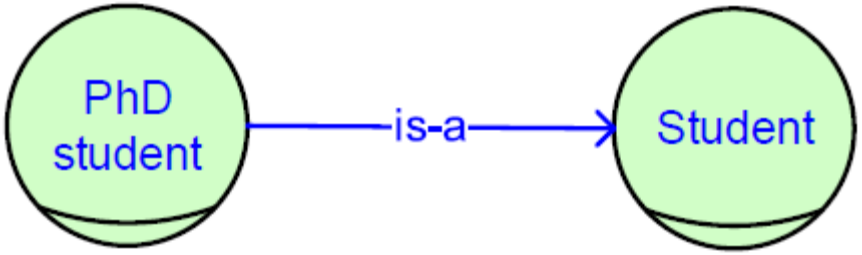
- I don't know at this time, or I do not care

  → **Actor**

Actor association links

- Often one wants to relate multiple actors (incl. agents & roles)
- iStar 2.0 offers binary, directed actor links

- **is-a**: represents the concept of generalization / specialization, and can be applied to (role to role) or (actor to actor)
  - Does not apply to agents. Why?



Actor association links

- **participates-in**: represents any kind of association, other than is-a, between two actors
- Depending on the linked elements, takes different meanings
  - (agent to role) typically represents the **plays** relationship

  - (linking elements of the same type) typically represents the **part-of** relationship

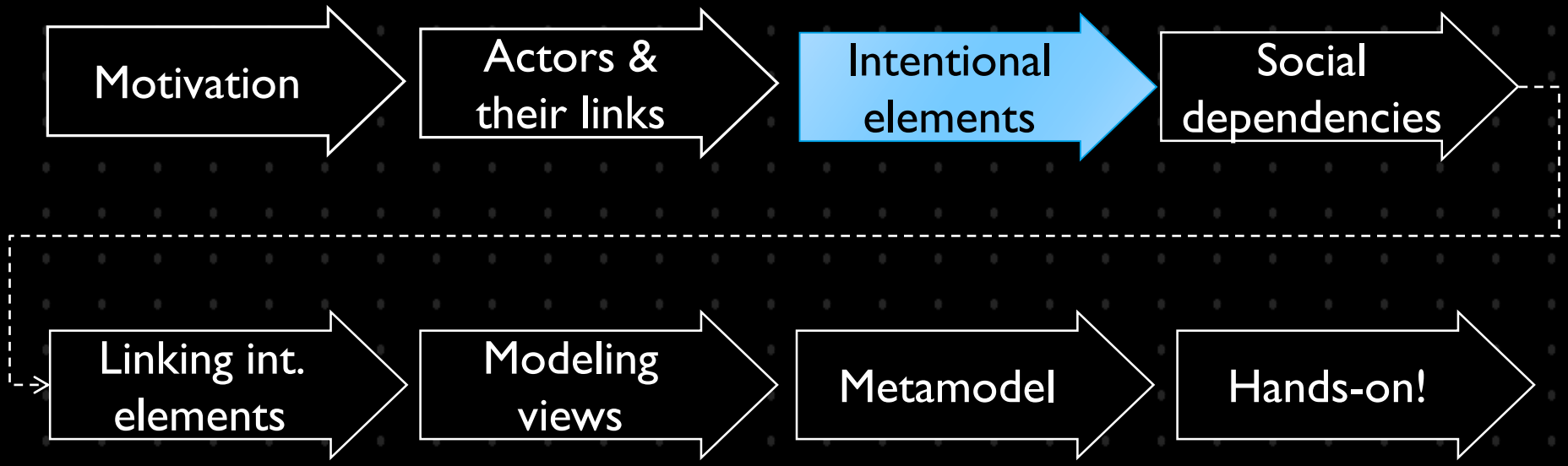
Actor association links

- **participates-in**: represents any kind of association, other than is-a, between two actors
- Depending on the linked elements, takes different meanings
  - (agent to role) represents the **plays** relationship

  - (linking elements of the same type) represents the **part-of** relationships

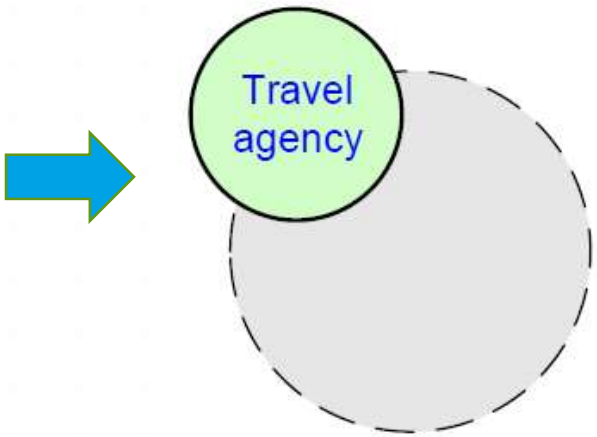
Examples for role-role?

Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Intentional elements

- iStar 2.0 focuses on **intentions: things that actors want**
- Intentional elements appear inside a so-called actor boundary, representing that actor's perspective in the model
- Four types of intentional elements
  - Goal
  - Quality
  - Task
  - Resource
- An actor with an empty actor boundary



Goals

- A **goal** is a state of affairs that the actor wants to achieve and that has clear-cut criteria of achievement
  - "Travel from Amsterdam to Osaka"
  - "Paper published"
  - "Tickets booked"

There is a clear criterion to determine if these are achieved. E.g., did I reach Osaka?

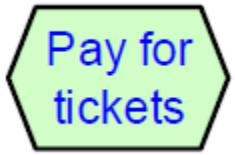
- Goals are represented as ovals

Qualities

- A **quality** is an attribute for which an actor desires some level of achievement
- Being attributes, they always relate to an entity
  - "Performance (of a system)"
  - "Yearly profit (of an organization)"
  - "Quick booking (of a trip)"
- Qualities guide the search for ways of achieving goals
- Represented as curved, cloud-like shapes

Tasks

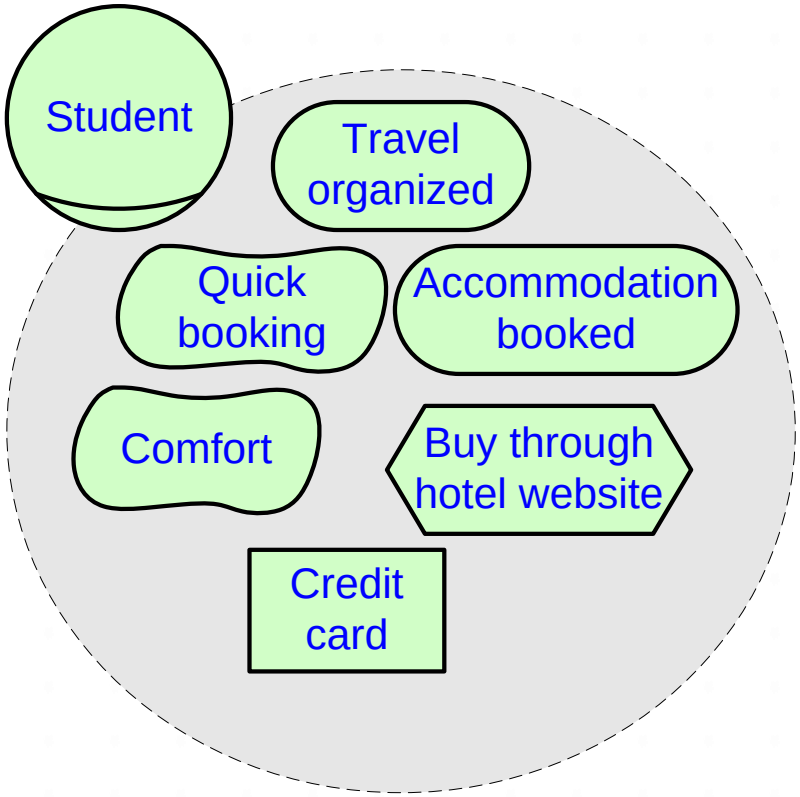
- A **task** represents actions that an actor wants to be executed
  - Usually within the purpose of achieving a goal
- Examples
  - "Pay for tickets"
  - "Take the train"
  - "Scan the receipt"
- Represented as diamonds



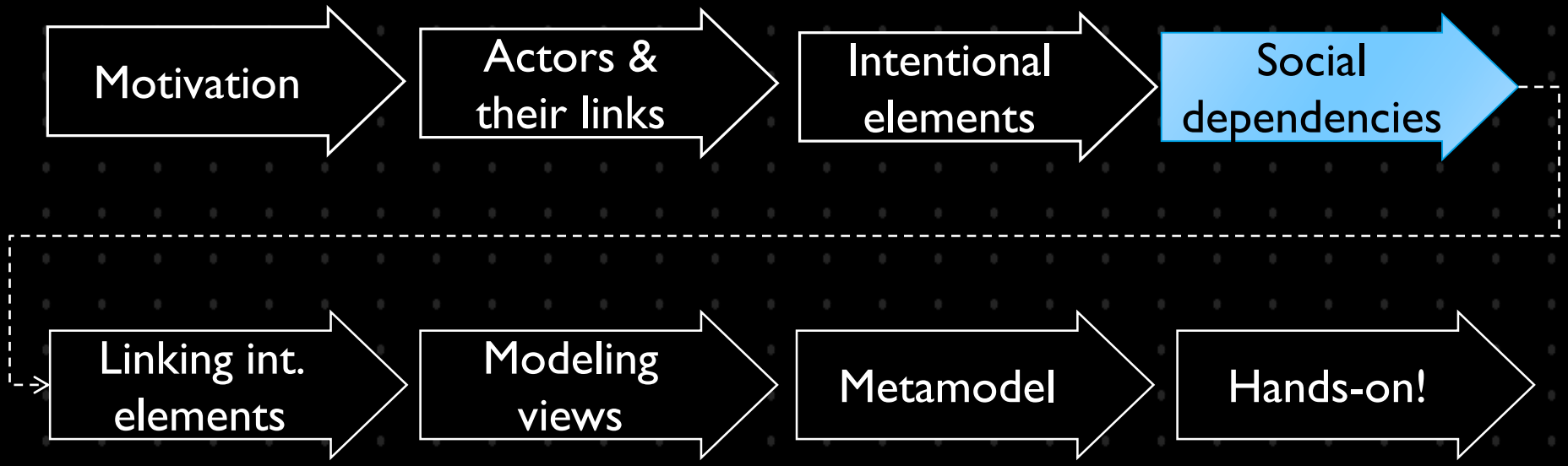
Resources

- A **resource** is a physical or informational entity that an actor requires in order to perform a task
- Examples
  - Credit card
  - Server
  - Personal details
- Represented as rectangles

Example of intentional elements



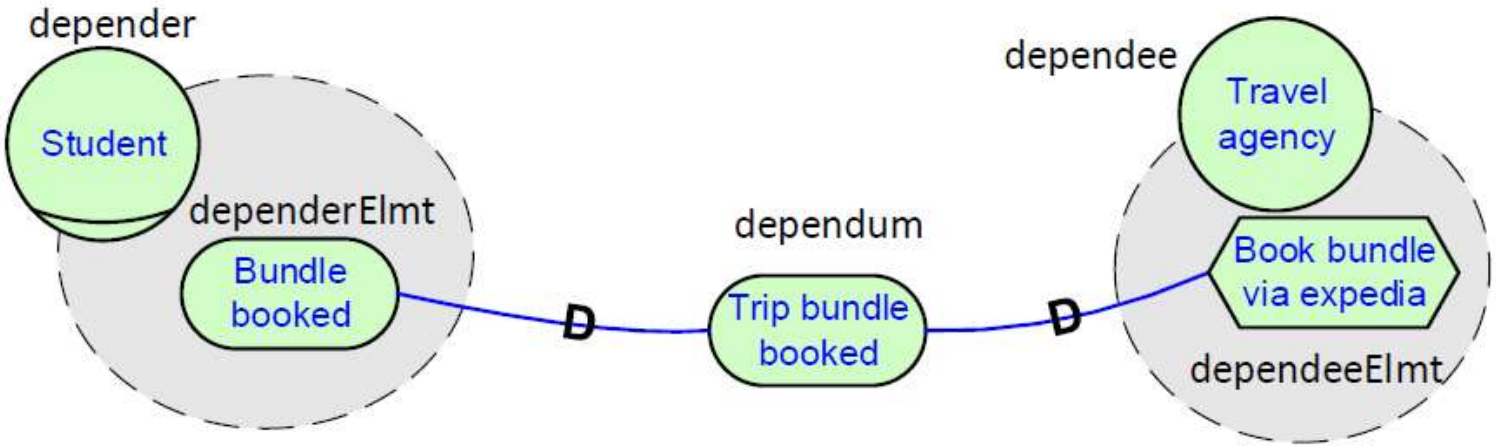
Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

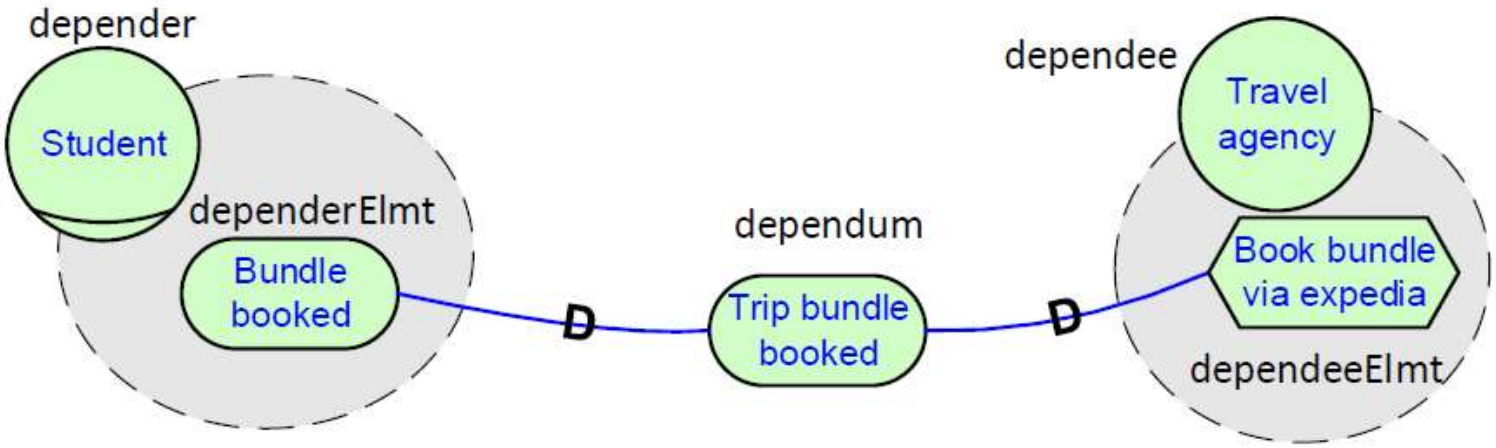
Dependencies

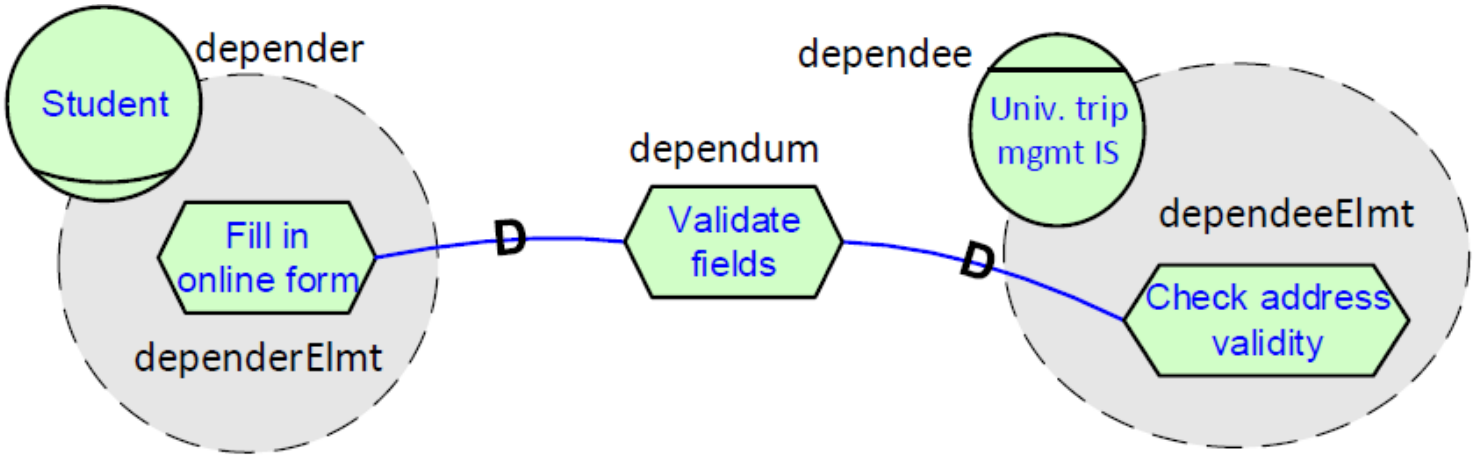
- Social relationships are represented as **dependencies**
- A dependency is a relationship with five arguments:
  - *Depender:* an actor that depends for something (the dependum) to be provided
  - *DependerElmt*: an intentional element within the depender's actor boundary where the dependency starts from, which explains **why** the dependency exists
  - *Dependum*: an intentional element that is the object of the dependency
  - *Dependee*: the actor that should provide the dependum
  - *DependeeElmt*: the intentional element that explains how the dependee intends to provide the dependum.

Dependencies, an example



Dependencies, an example





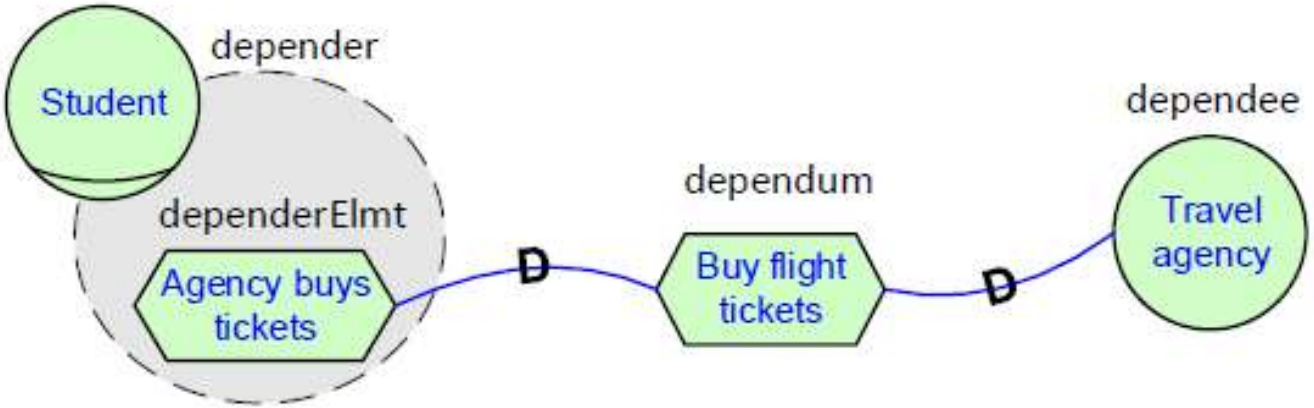
Any differences?

Dependum types

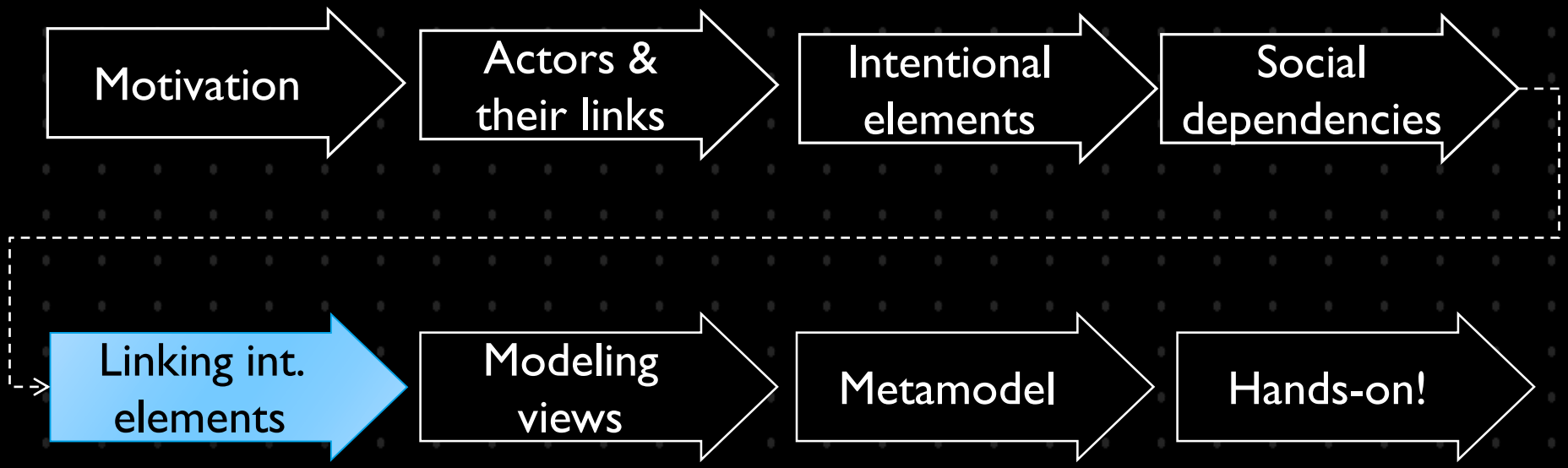
- The type of the dependum specializes the semantics of the dependency relationship
  - **Goal**: the dependee is free to choose how to achieve the goal
  - **Quality**: the dependee is free to choose how to sufficiently satisfy the quality
  - **Task**: the dependee is expected to execute the task in a prescribed way
  - **Resource**: the dependee is expected to make the resource available to the depender
- Different dependum types give the dependee different degrees of freedom

Omitting dependency parts

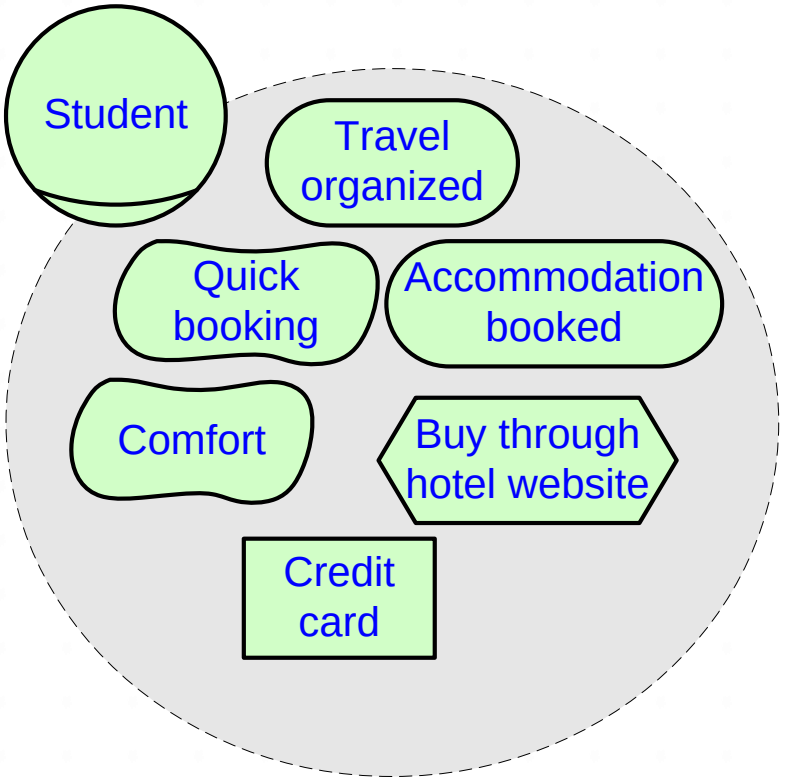
- **Omitting the dependerElmt** implies not specifying why the dependency exists
- Omitting the **dependeeElmt** implies not specifying how the dependency will be fulfilled



Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Intentional element links



The elements within an actor boundary are interrelated.

But we have seen no ways to relate them so far.

Any idea?

Intentional element links: overview

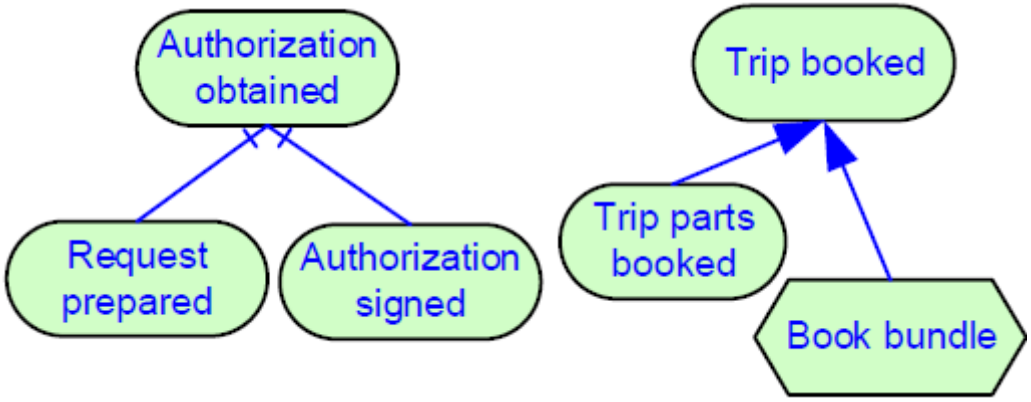
- Four link types:
  - Refinement
  - NeededBy
  - Contribution
  - Qualification

		Arrowhead pointing to			
		Goal	*Quality*	*Task*	*Resource*
Link starts from	*Goal*	Refinement	Contribution	Refinement	n/a
	Quality	Qualification	Contribution	Qualification	Qualification
	Task	Refinement	Contribution	Refinement	n/a
	Resource	n/a	Contribution	NeededBy	n/a

Refinement

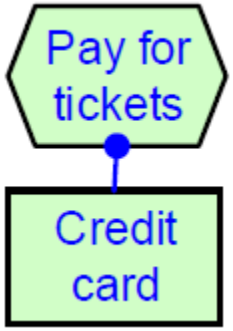
- **Refinement** is a generic relationship that links goals and tasks hierarchically
  - n-ary relationship linking one parent to one or more children
  - An intentional element can be the parent in at most one refinement link
- Two types of refinement
  - **AND:** the fulfillment of all *n* children (n≥2) makes the parent fulfilled
  - **Inclusive OR:** the fulfillment of at least one child makes the parent fulfilled

AND & OR refinements



NeededBy

- The **NeededBy** relationship links a task with a resource and it indicates that the actor needs the resource in order to execute the task
  - No details on the reason for this need: consumption, reading, …

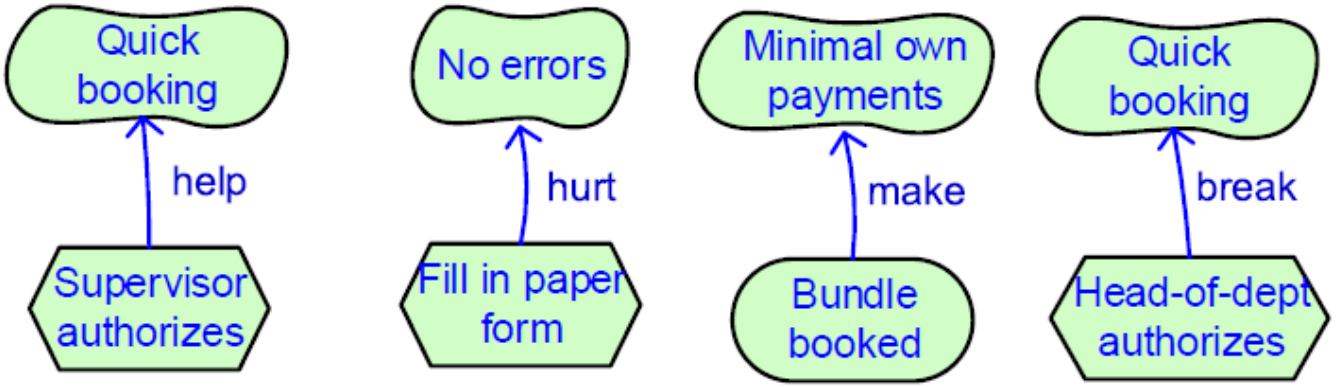


Contribution

- **Contribution links** represent the effects of intentional elements on qualities
  - These are qualitative links
  - Assist analysts in the decision-making process among alternative goals / tasks
- Qualities can be
  - **Fulfilled** (or satisfied), having sufficient positive evidence
  - **Denied**, having strong negative evidence
- No details here on how fulfillment / denial are calculated

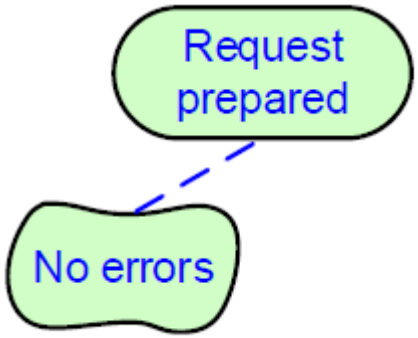
Contribution types

- Four types, expressing that "the source provides…"
  - **Make**: sufficient positive evidence for the satisfaction of the target
  - **Help**: weak positive evidence for the satisfaction of the target
  - **Hurt**: weak evidence against the satisfaction (or for the denial) of the target
  - **Break**: sufficient evidence against the satisfaction (or for the denial) of the target

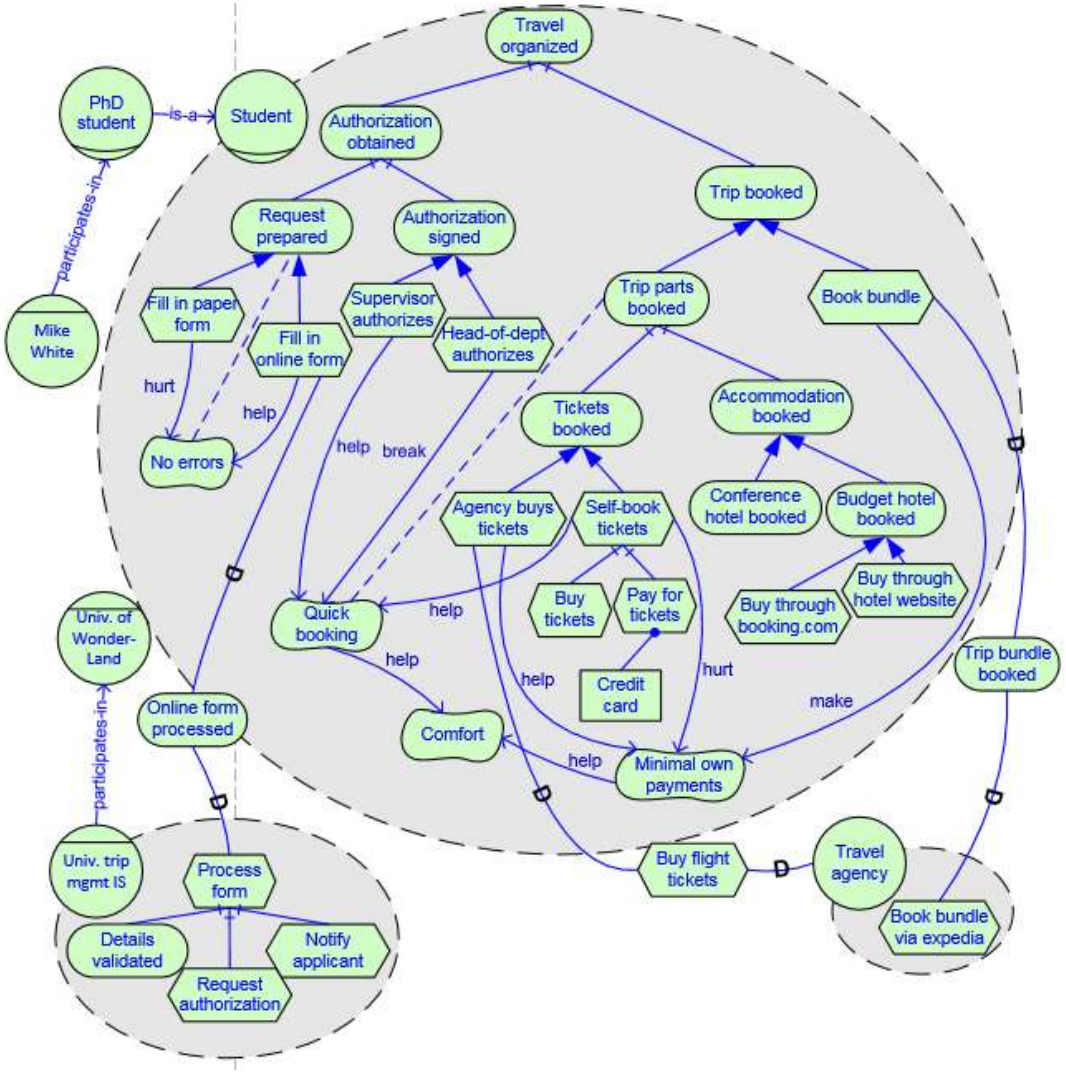


Qualification

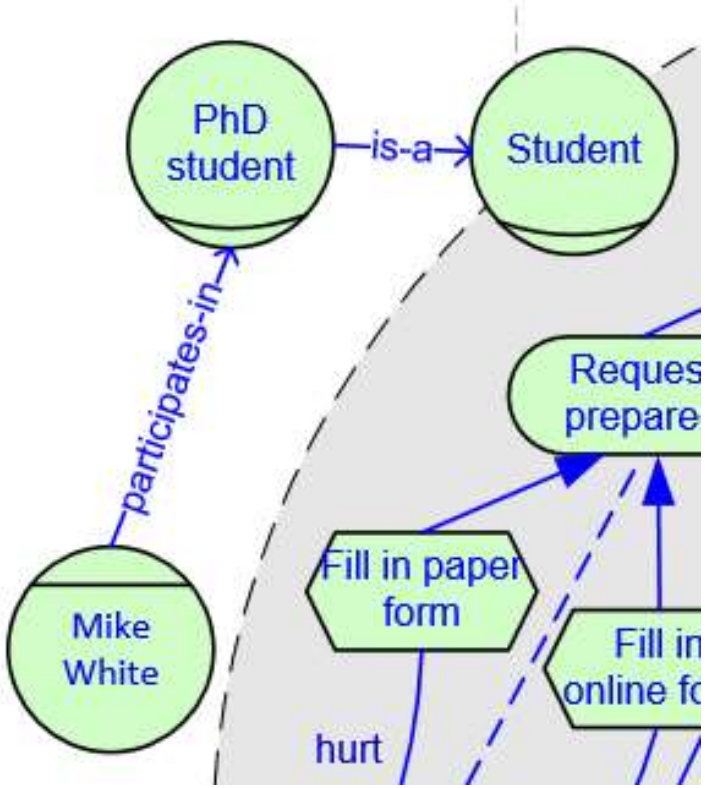
- The **qualification** relationship relates a quality to its subject: a task, goal, or resource
- Examples:
  - the quality "Quick booking" refers to the goal "Trip parts booked", elaborating on how this goal might be achieved
  - the quality "No errors" refers to errors possibly created while fulfilling the goal "Request prepared"



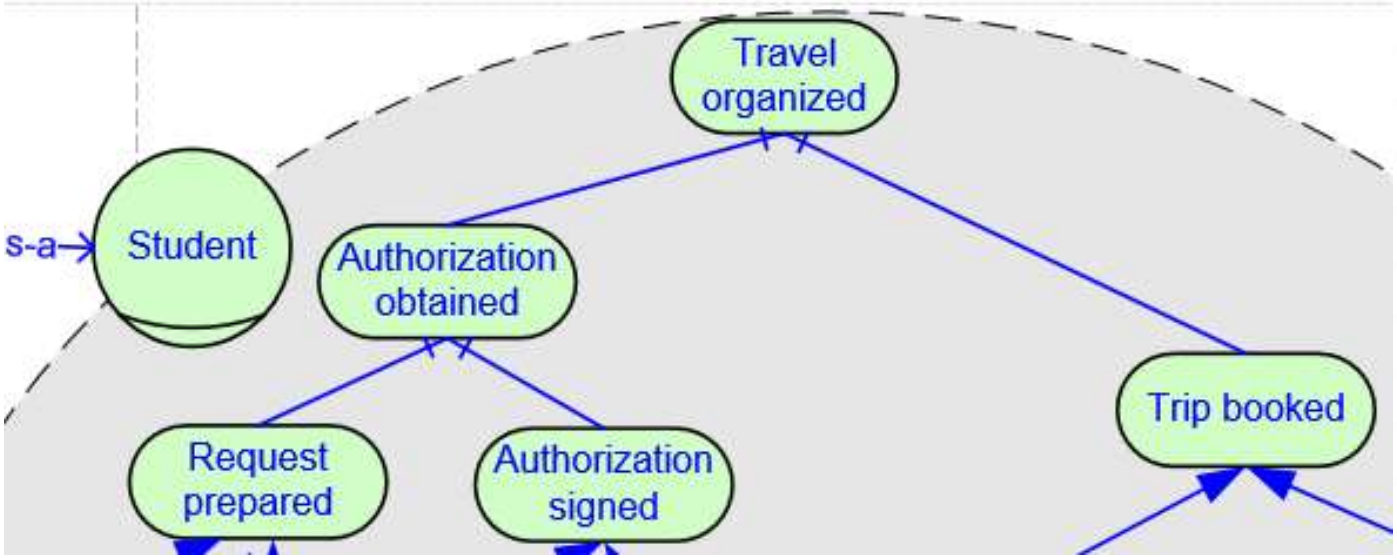
The resulting model: full!



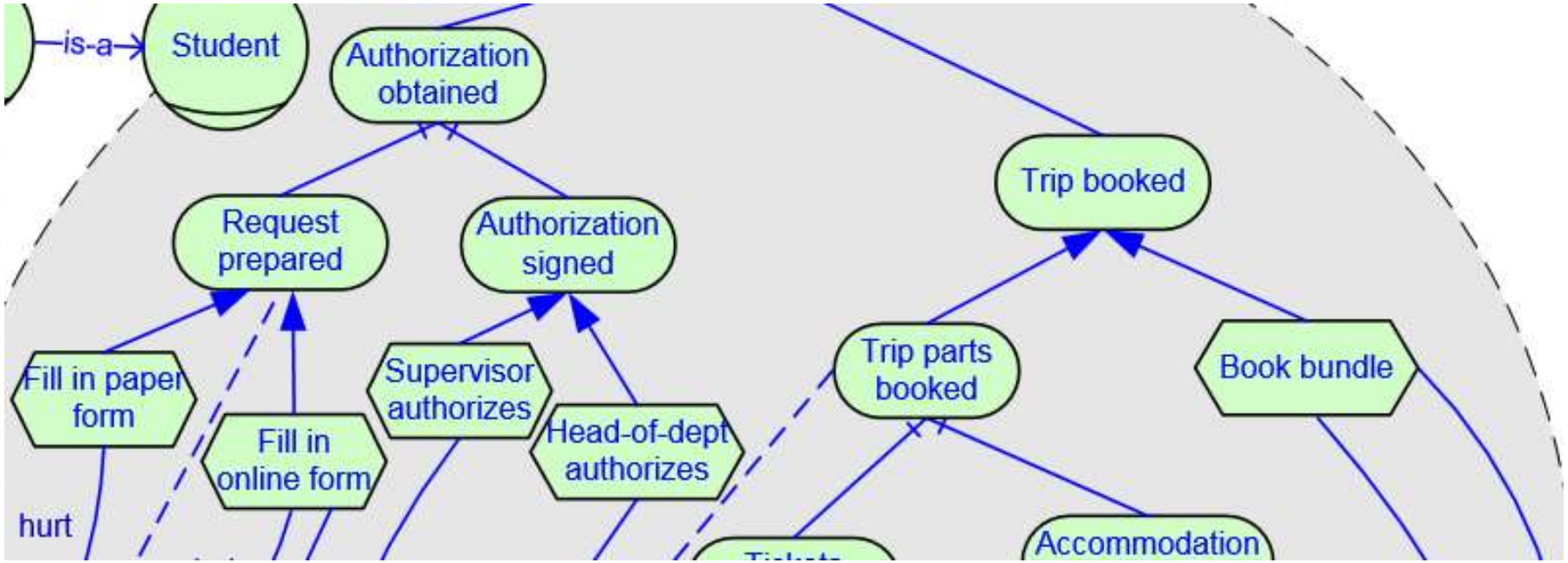
Zoom-in: actors and their links



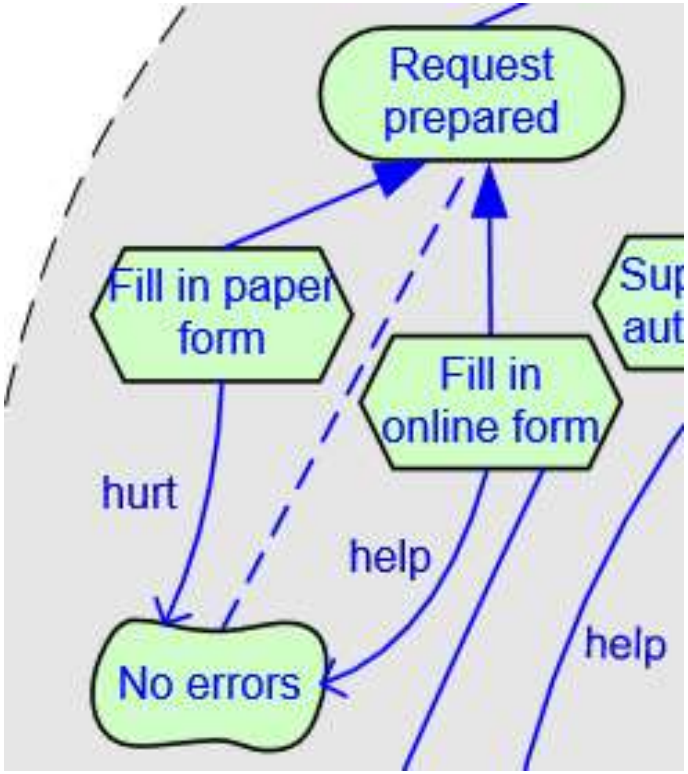
Zoom-in: goals AND-refinement



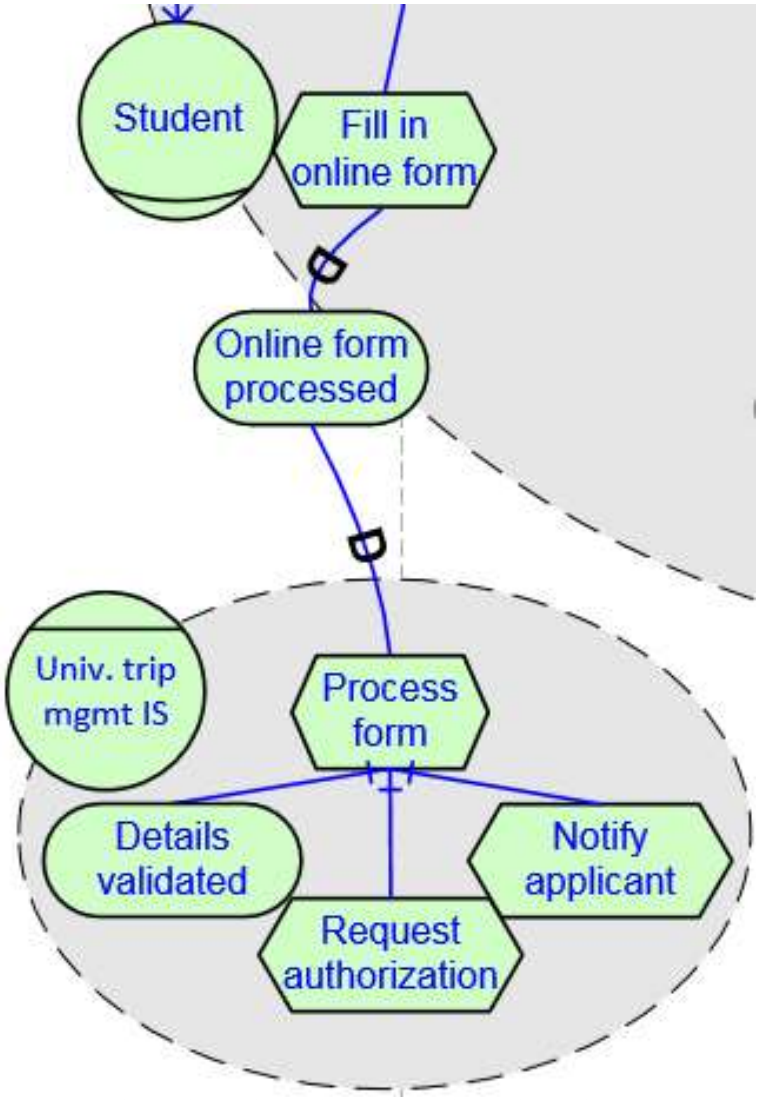
Zoom-in: goals OR-refinement



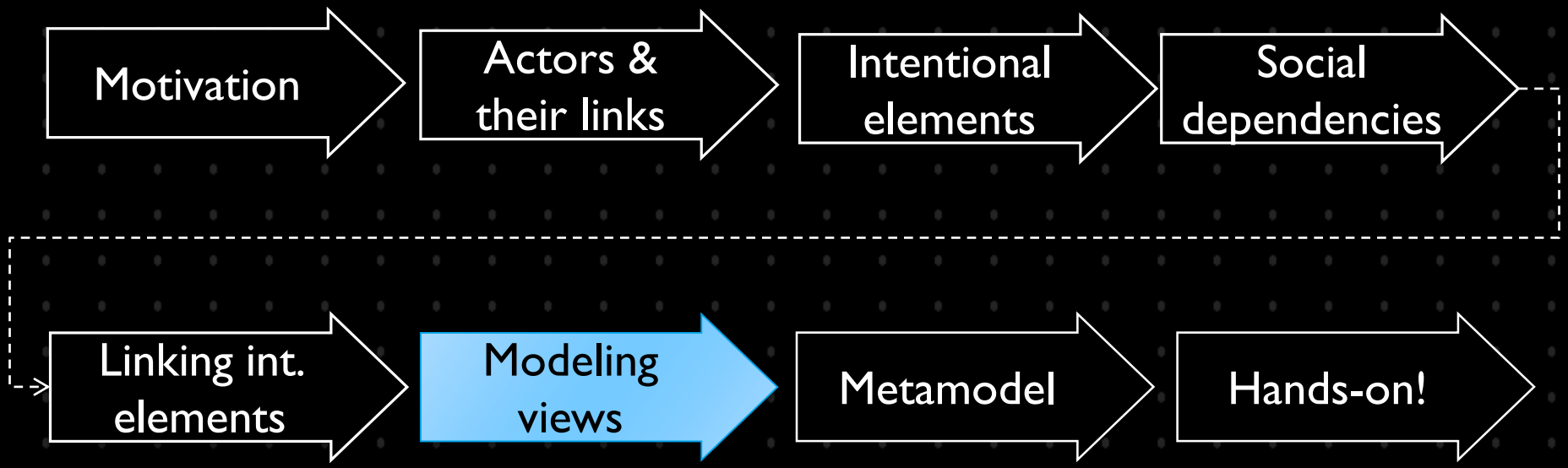
Zoom-in: qualities to compare alternatives



Zoom-in: dependencies



Outline

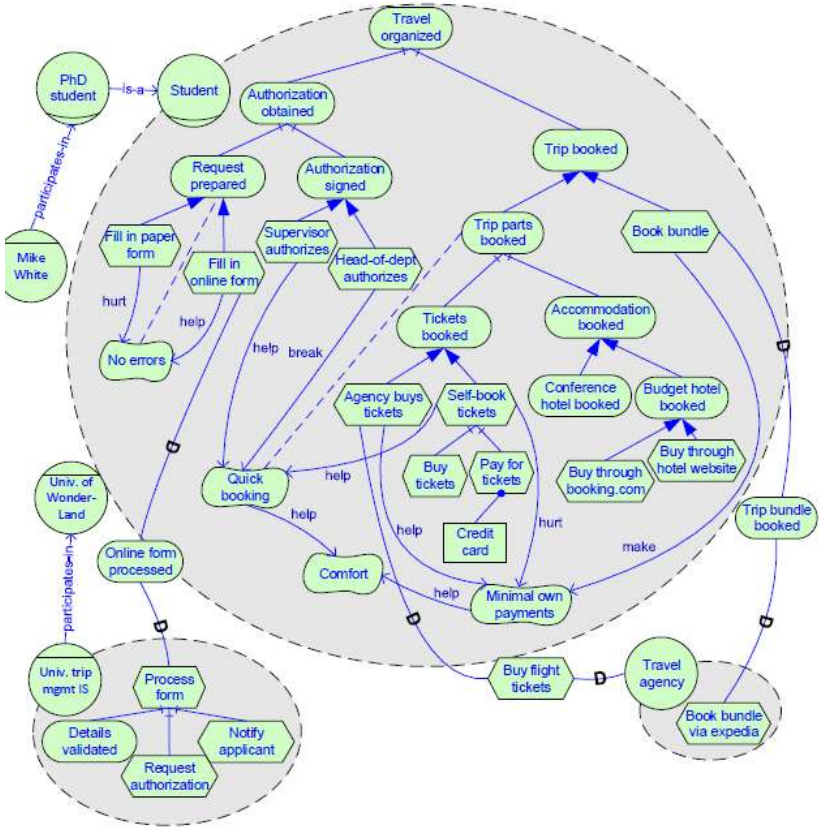

Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Model views

- When using iStar 2.0, the analyst creates a **model**
- Such model can be visualized via multiple perspectives or **model views**
- Standard views exist, including two from *i**:
  - Strategic rationale (SR)
  - Strategic dependency (SD)
- Hybrid views can be defined

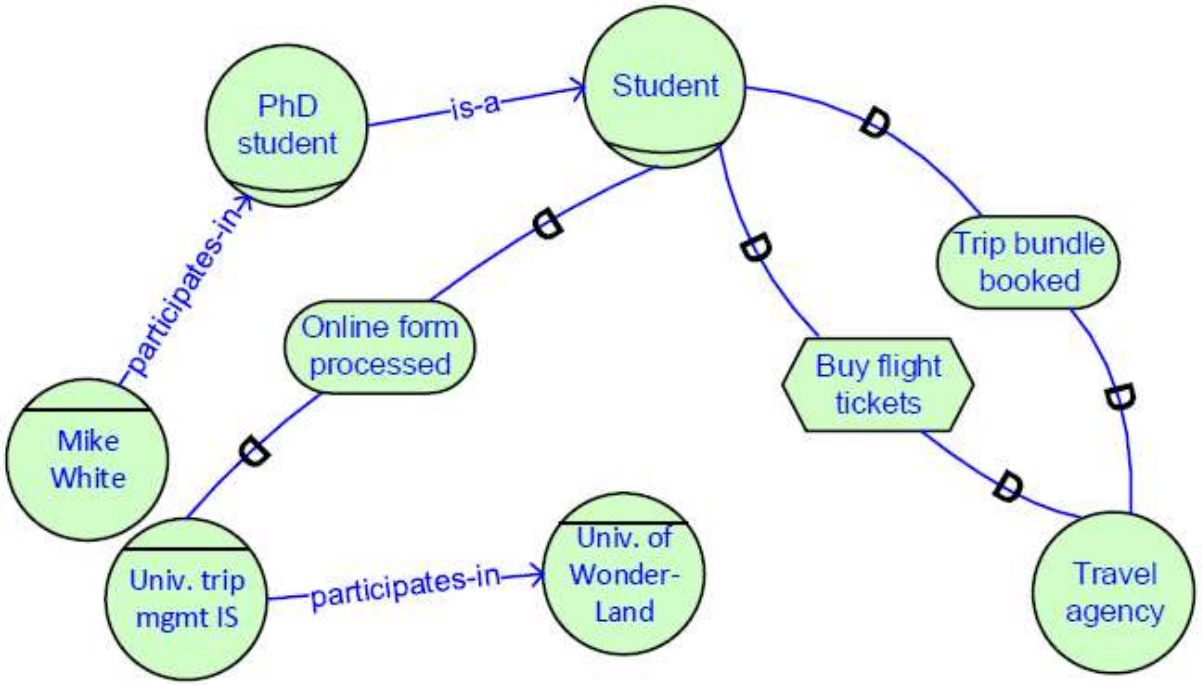
Strategic rationale in iStar 2.0

- Shows **all** details captured in the model!
  - Actors
  - Actor links
  - Intentional elements
  - Dependencies
  - Intentional element links



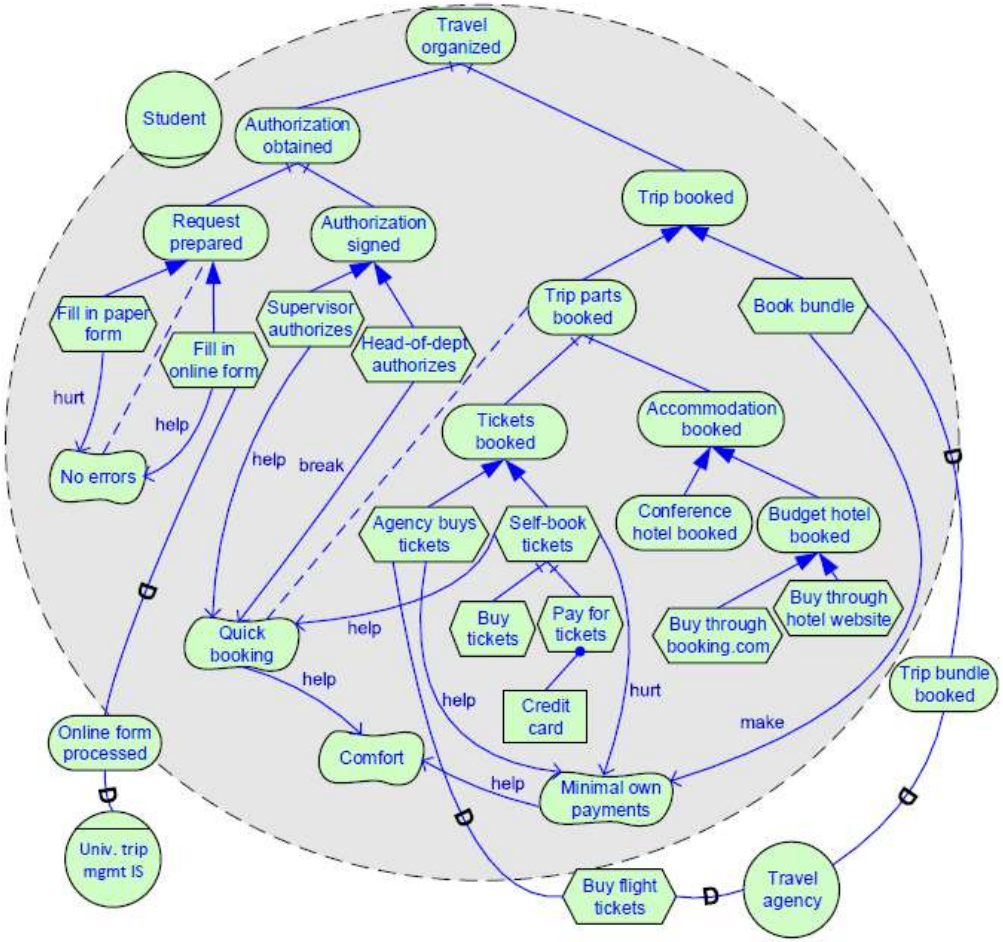
Strategic dependency in iStar 2.0

- Shows **only** the social part of the model
  - Actors
  - Actor links
  - Dependencies
- … but **not** intentional ele-ents and their links

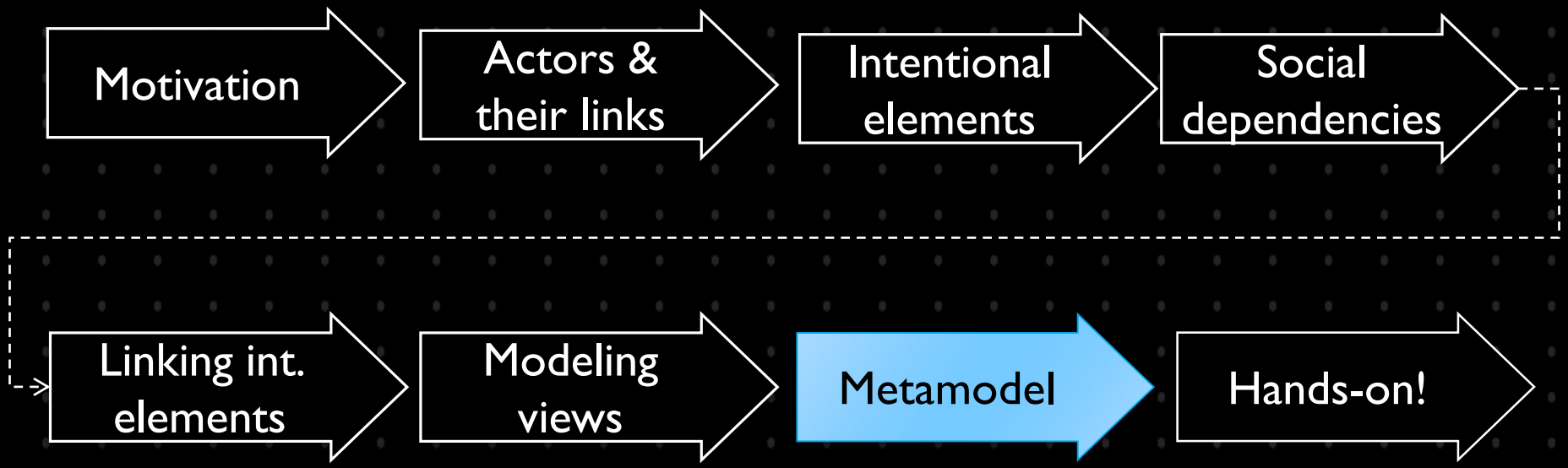


A hybrid view

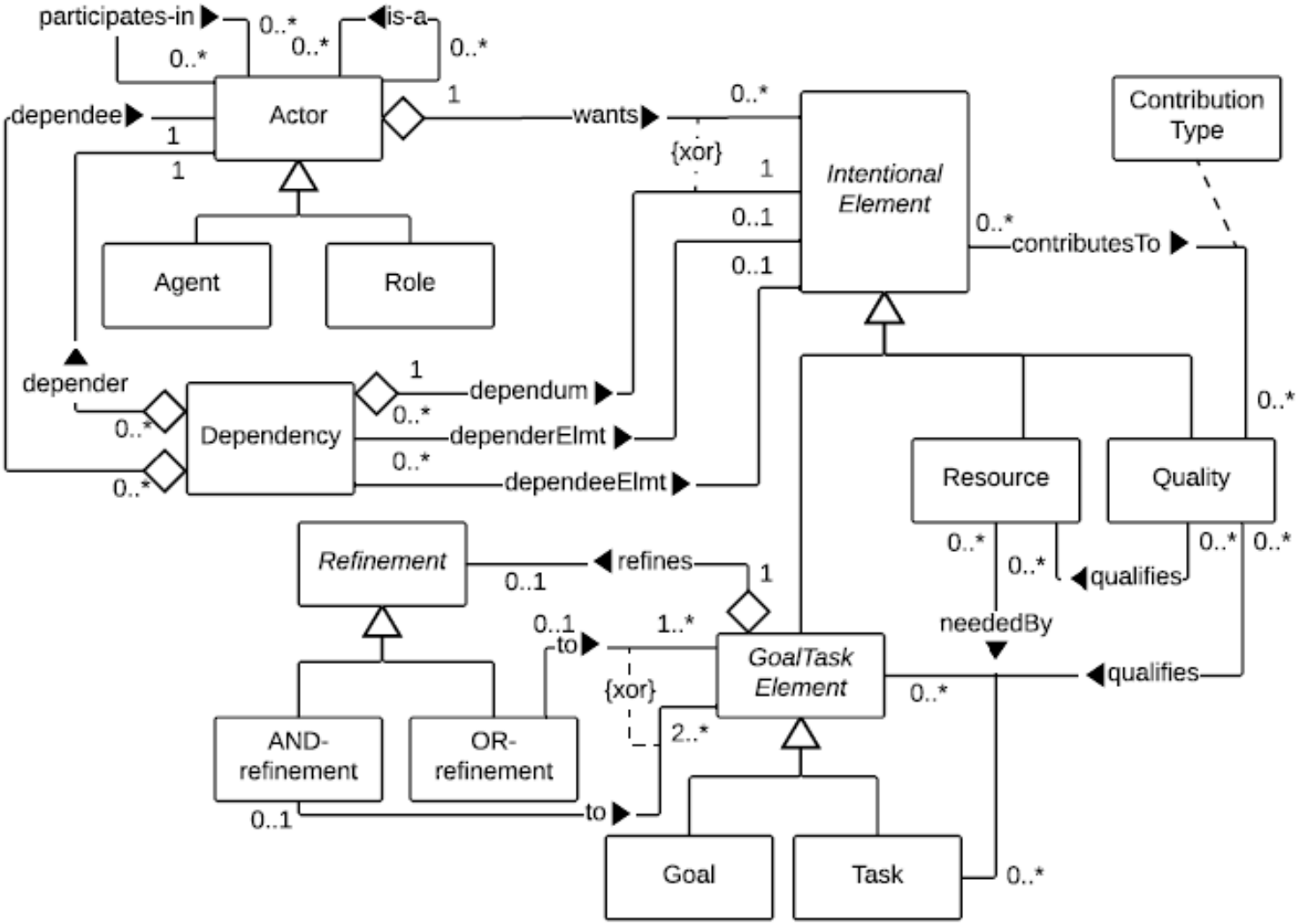
- For example
  - Some actor boundaries are open, but not all
  - Actor links are hidden
- Other hybrid views
  - *Functional* (no qualities)
  - *Actor* view (only actors and their links)



Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Syntax of iStar 2.0



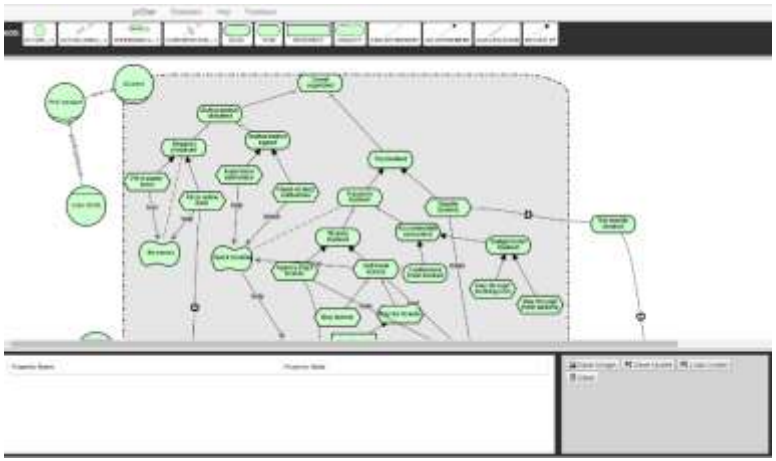
Precise syntax of iStar 2.0

- Some details cannot be captured via a metamodel
- A few of them here (more in the paper)
  - No *is-a* cycles
  - No *participates-in* cycles
  - Two actors can be linked by at most one actor link
  - The depender and dependee of a dependency should be different actors
  - Refinement should not lead to refinement cycles
  - It is not possible for a quality to contribute to itself

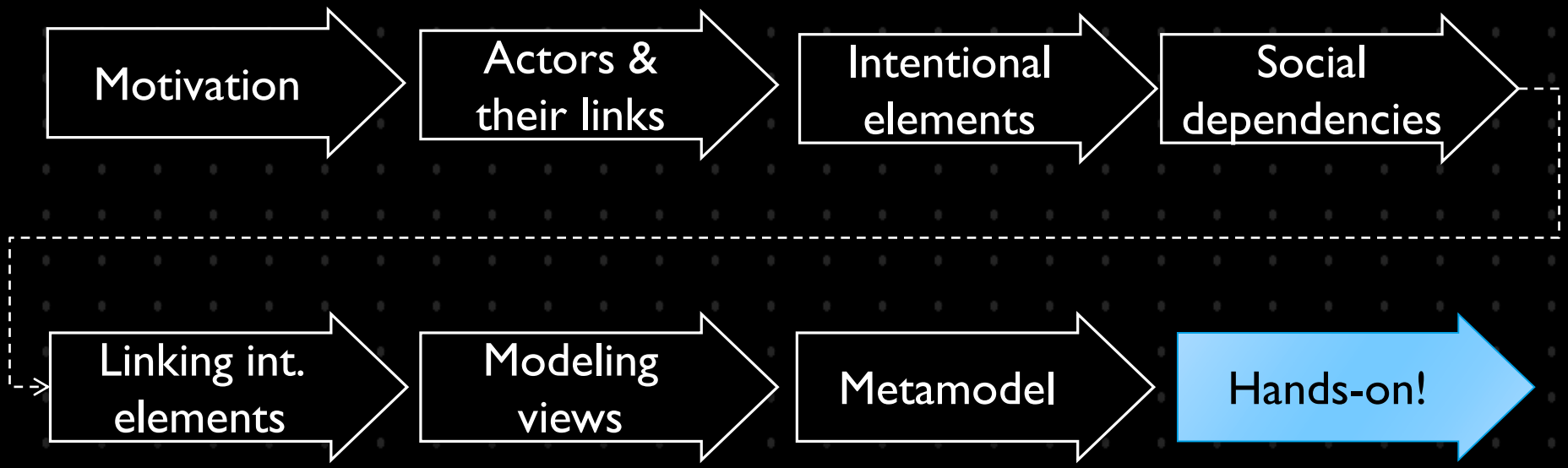
Why is precise syntax that important?

- Minimize ambiguity to facilitate homogeneous learning
- Guide tool developers

- Two weeks after the release of the standard, the first iStar 2.0 compliant tool was released by researchers in Brazil



Outline


Motivation
Actors & their links
Intentional elements
Social dependencies
Linking int. elements
Modeling views
Metamodel
Hands-on!

Practice!

- Take a scenario as instructed by the student assistant
- Tasks
  - Identify the main actors
  - Define their goals
  - Find their dependencies
  - Use intentional element links
  - Analyze and evaluate alternative ways of fulfilling goals!
- Use the cheat sheet
- Create the models pen-on-paper
  - Scan and send us the models by the end of the day!

Literature

- Fabiano Dalpiaz, Xavier Franch, Jennifer Horkoff. *iStar 2.0 Language Guide*. arXiv:1605.07767, 2016 https://arxiv.org/abs/1605.07767