

Diamond Path Framework Practically

© Leonard Wallezký

Why we need diamonds?

- ▶ We need to describe things
- ▶ And their relations
- ▶ In some given context
- ▶ Than we need to organize/plan operations
- ▶ And execute them in some time perspective
- ▶ Our natural language is
 - Redundant
 - Ambiguous

4 diamonds

- ▶ See
 - Describing things (objects) and basic relations
- ▶ Recognize
 - Adding context to relations
- ▶ Organize
 - How agents behave to recognized objects, what kind of operations we can do
- ▶ Do
 - Executing planned operations and getting results

See

- ▶ We are describing the seen object
- ▶ It has
 - Particular shape or form
 - There can be some different varieties of this object
 - It can be used for some purposes
 - Using this object is under some rules
- ▶ There can be connections to other objects

Which objects do we find interesting for modelling?

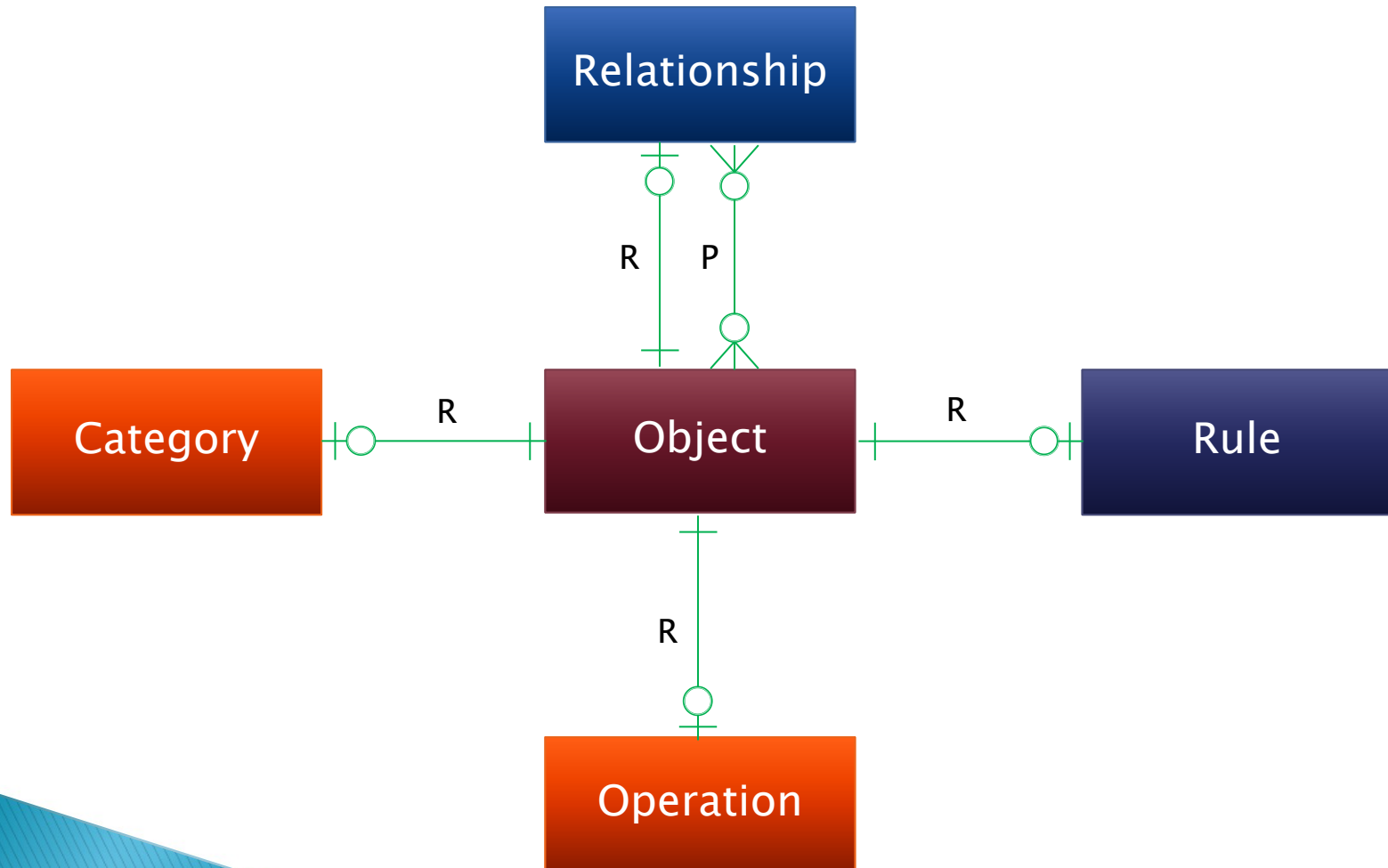
Relationship

Category

Rule

Operation

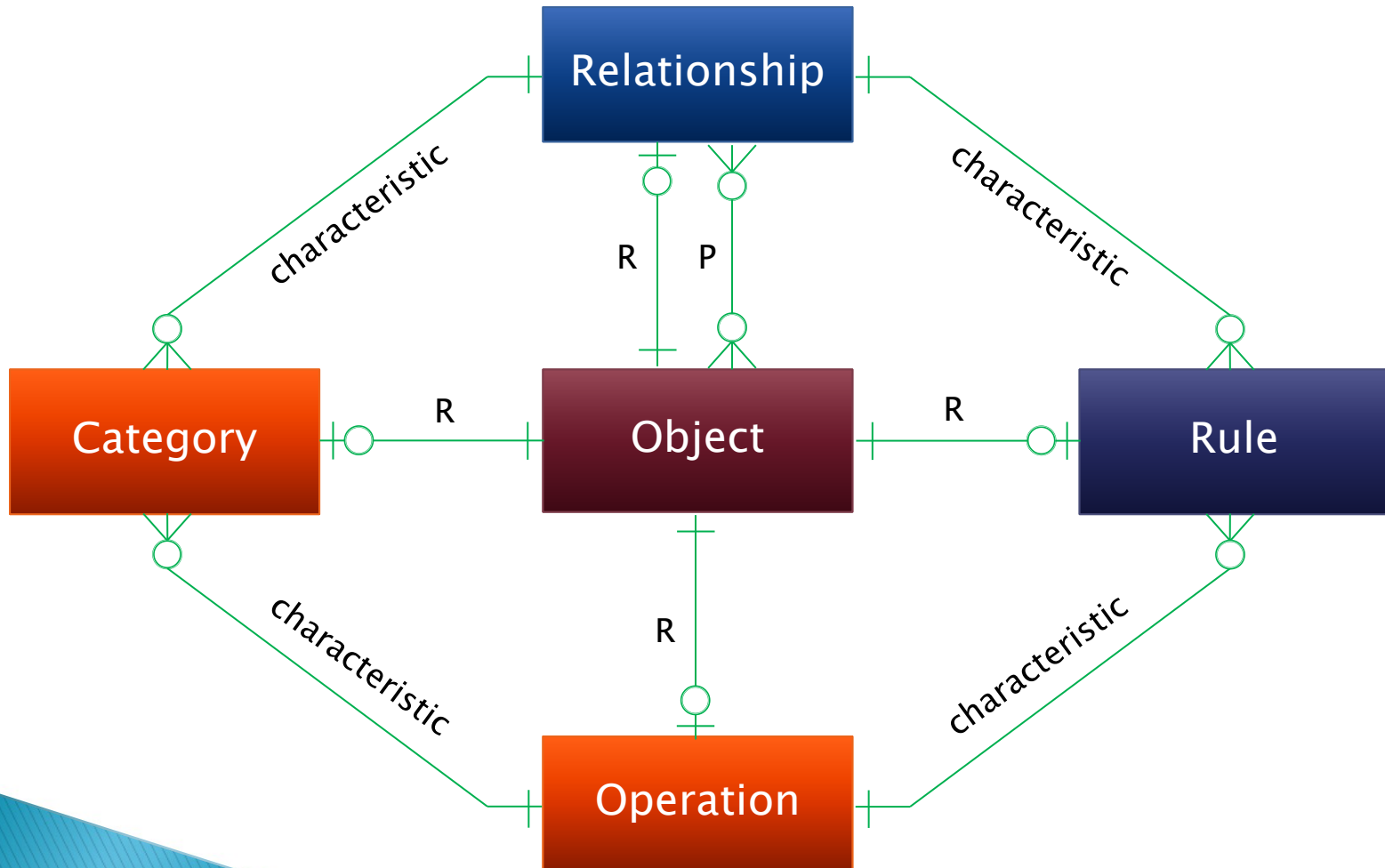
MENTION – USE duality



Universality Principle Implementation

- ▶ Edges R1, R2, R3 and R4 are together called R-edges. R-edges serve to perform transitions between **mentioning** and **using** and vice-versa.
- ▶ If we focus on a concrete object of the class #Object, which was spoken of in a way (MENTION), then by the operation **USE** we will pass the relevant R-edge and we will reach represented sequence (#Connection) or operation (#Operation) or category (#Category) or rule (#Rule), which could be then directly used, or we will reach nothing.
- ▶ If we focus any vertex of the Diamond graph on the other hand (which we used in a way – USE), then by the operation **MENTION** we will pass the relevant R-edge and we will reach this vertex representing object, which we can speak of directly.

Diamond of Attention Focussing



Container (#Object)

- ▶ Dmt-objects dwell in container (#Object) in Diamond. Containers used in Diamond will be defined by specification of their member elements.
- ▶ Container (#Object) is defined in such a way that it contains all such dmt-objects which can be mentioned in DMT, i.e. which *can be assigned certain properties*.

Container (#Connection)

- ▶ Container (#Connection) is defined in such a way that every of its elements is a sequence of the length n (n -tuple) of dmt-objects, where n is some finite natural number.
- ▶ Every element of the container (#Connection) is called a **connection** or a **sequence**.
- ▶ Container (#Connection) contains by definition one special element \perp called improper connection. It is not possible to determine objects constituting improper connection.

Container (#Operation)

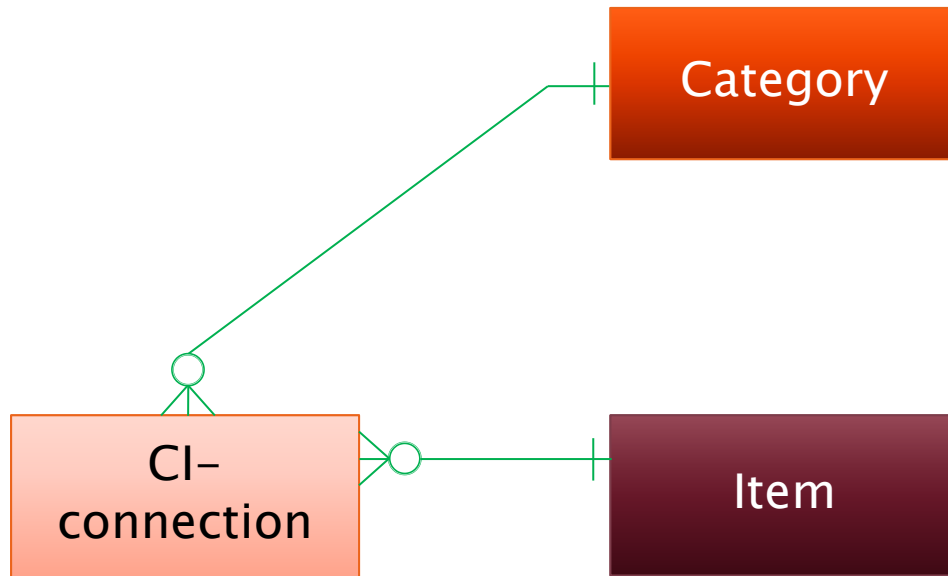
- ▶ Container (#Operation) is defined in such a way that every of its elements is an algorithmically computable transformation of one state of DMT to, generally, another state of DMT.
- ▶ By the state of DMT we mean one particular filling of DMT (as a container) by elements-instances, that may dwell in DMT.
- ▶ We assume there always are two (boolean) objects in DMT, one of them codes True, the other False.
- ▶ The elements of the container (#Operation) are called *operations*.
- ▶ Some operations may *fail*, i.e. return object coding False, under certain circumstances and *succeed under different circumstances*. Some operations always succeed.

Container (#Category)

- ▶ Container (#Category) is defined in such a way that every of its elements has the following properties:
 1. it is a container for dmt-objects,
 2. it is one-to-one mapped to the pair $\langle C_n, O_p \rangle$, where
 $C_n \in (\#Connection)$, $O_p \in (\#Operation)$, and
 3. it holds about the operation O_p that by means of the connection C_n it can recognize whether a given object is or is not in this container.
- ▶ The elements of the container (#Category) are called **categories**.
- ▶ The connection C_n is called a *defining connection* of this category, the operation O_p is called a *defining operation* or an *evaluator* of this category.

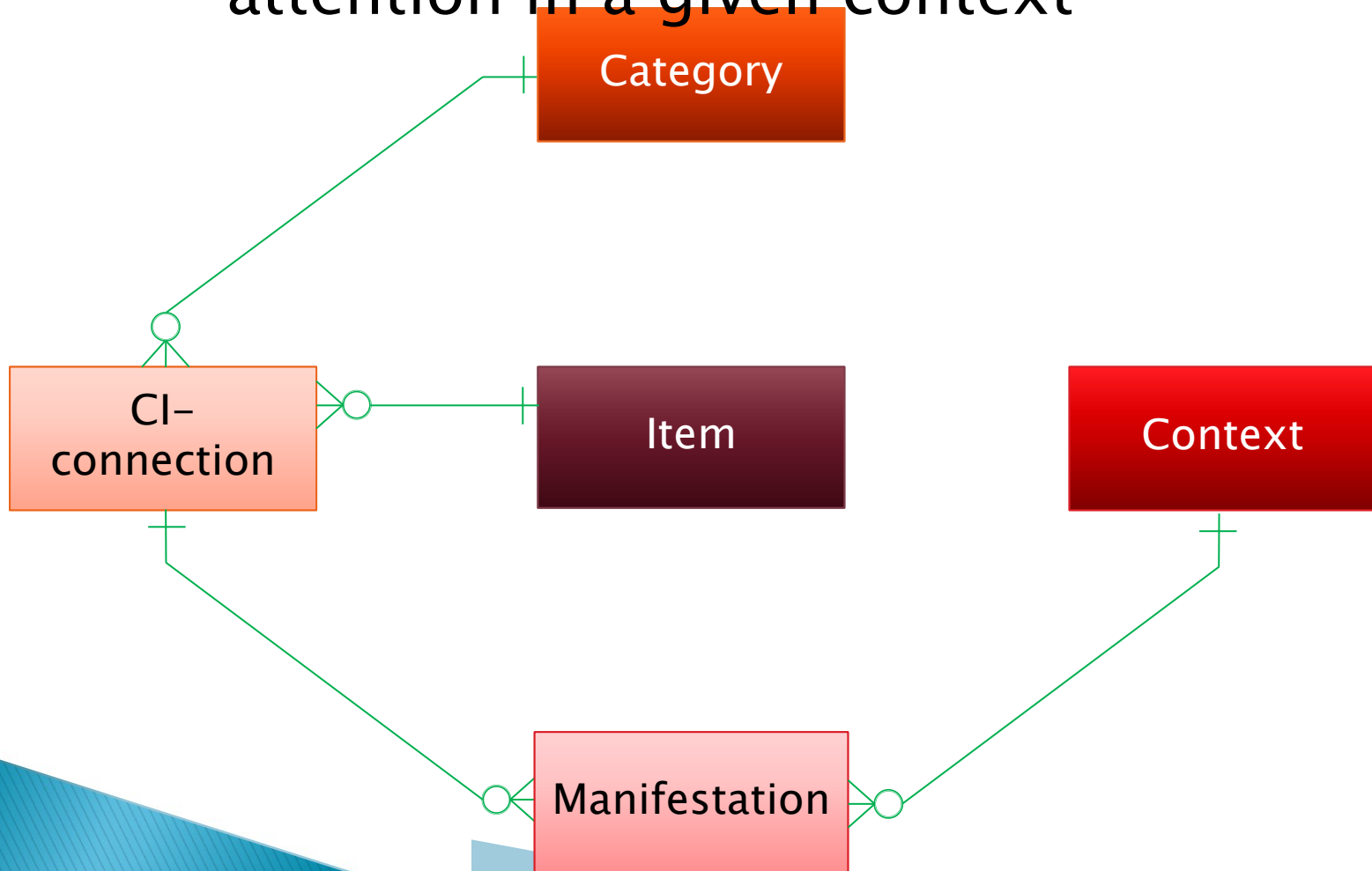
Certainty

- Items (= objects as such, not their constructs) belongs to a category with a given certainty

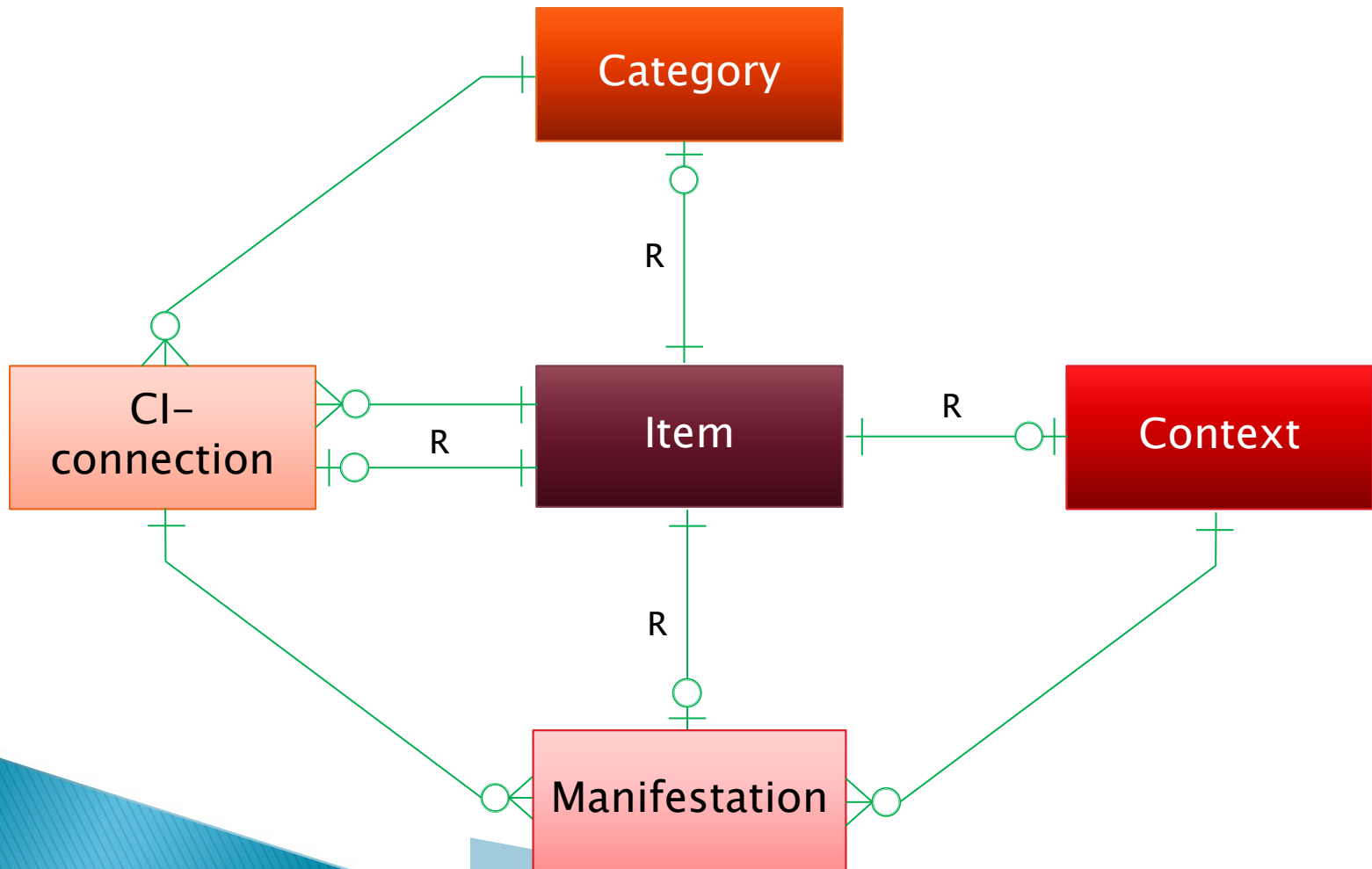


Attention

- ▶ The fact is manifested with a certain attention in a given context

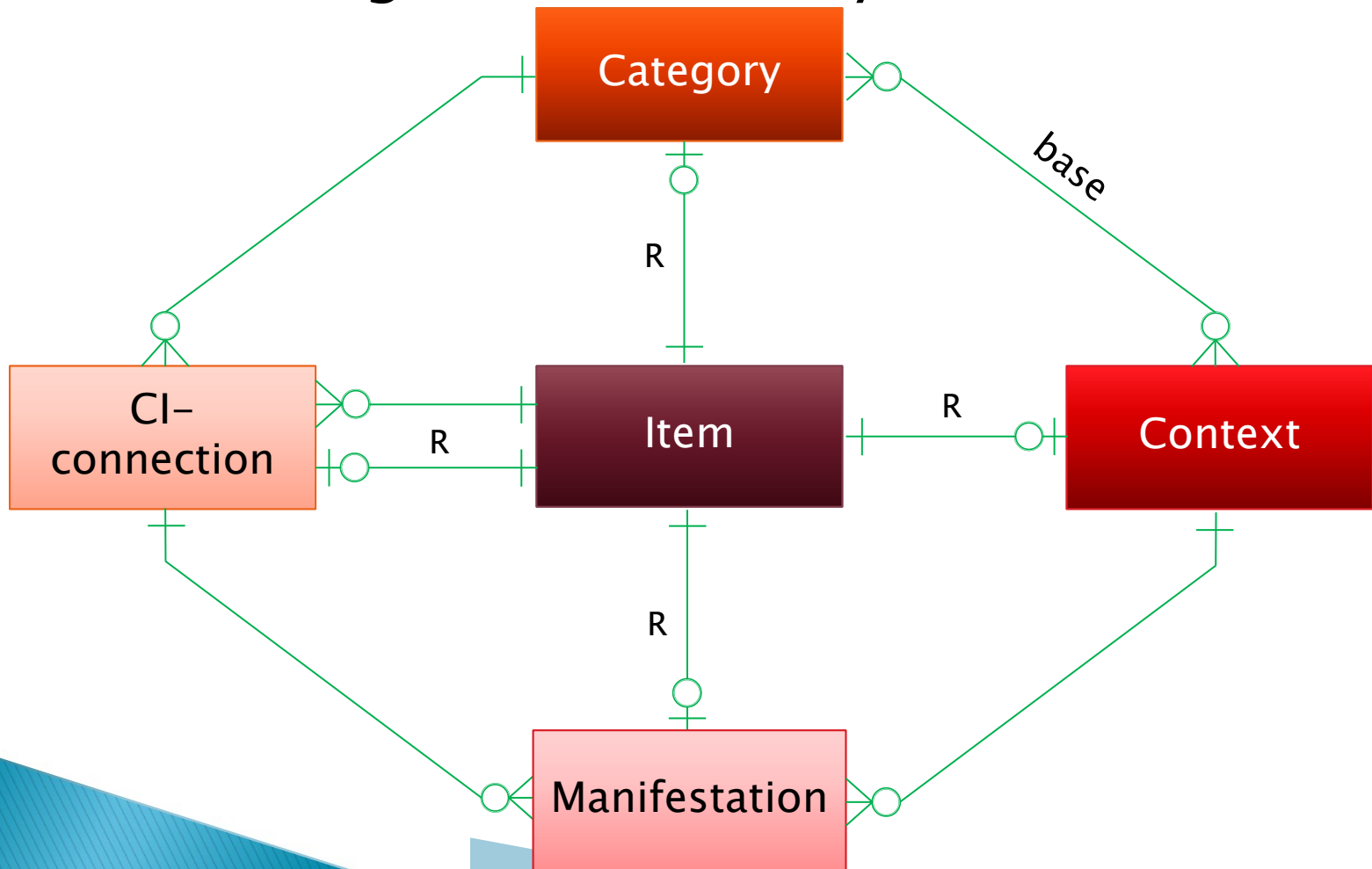


R-edges



Context base

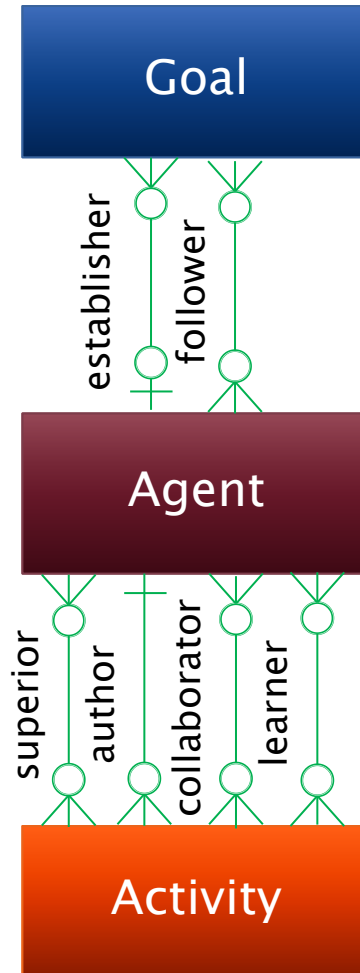
- ▶ Context serves as a model. The base edge defines the set of categories to classify its items to

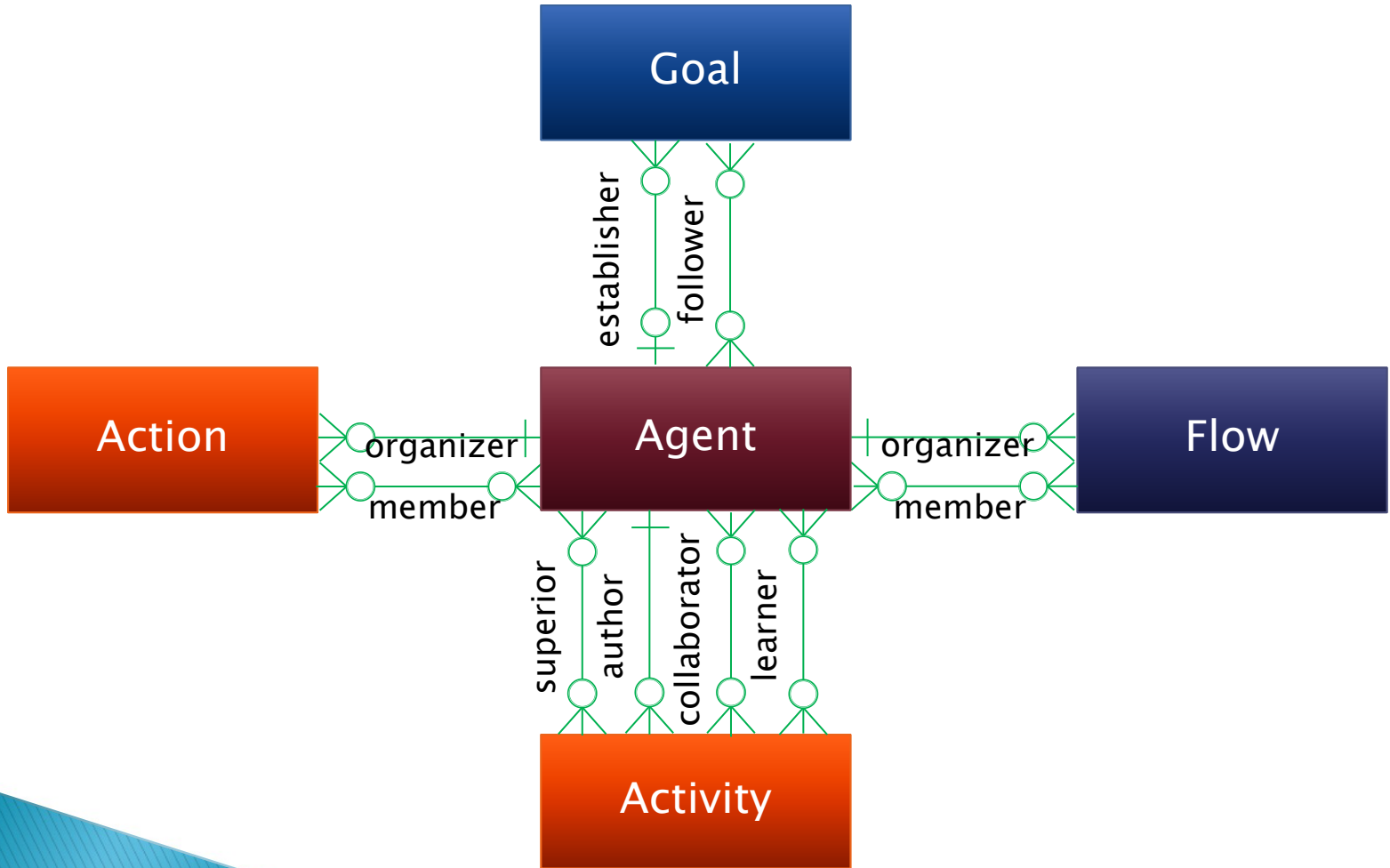


3rd diamond

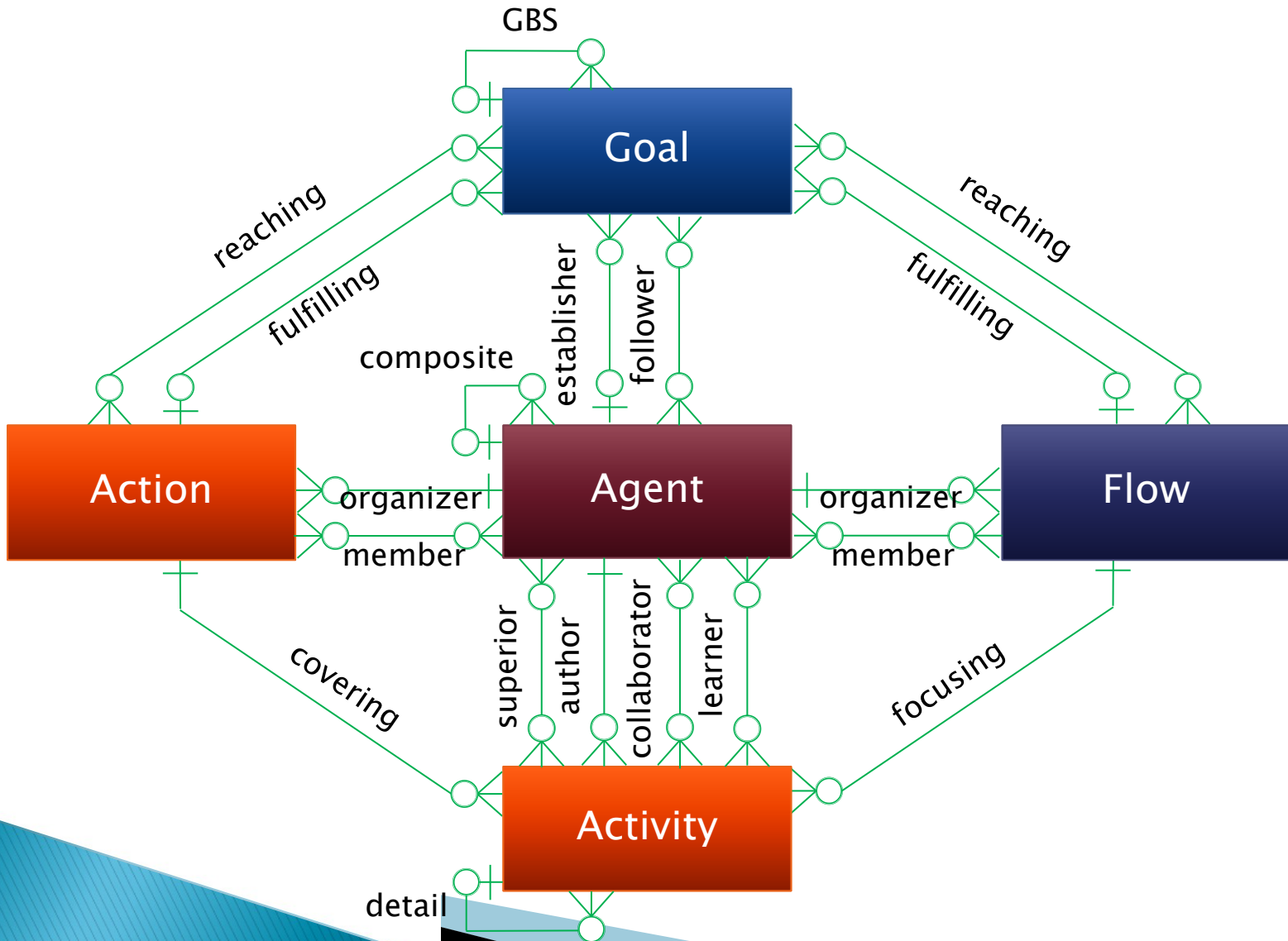
- ▶ Organizing diamond
- ▶ How is your life / position / work organized
- ▶ How can be some agent
 - Member of some team
 - Work on projects
 - Be educated or teach

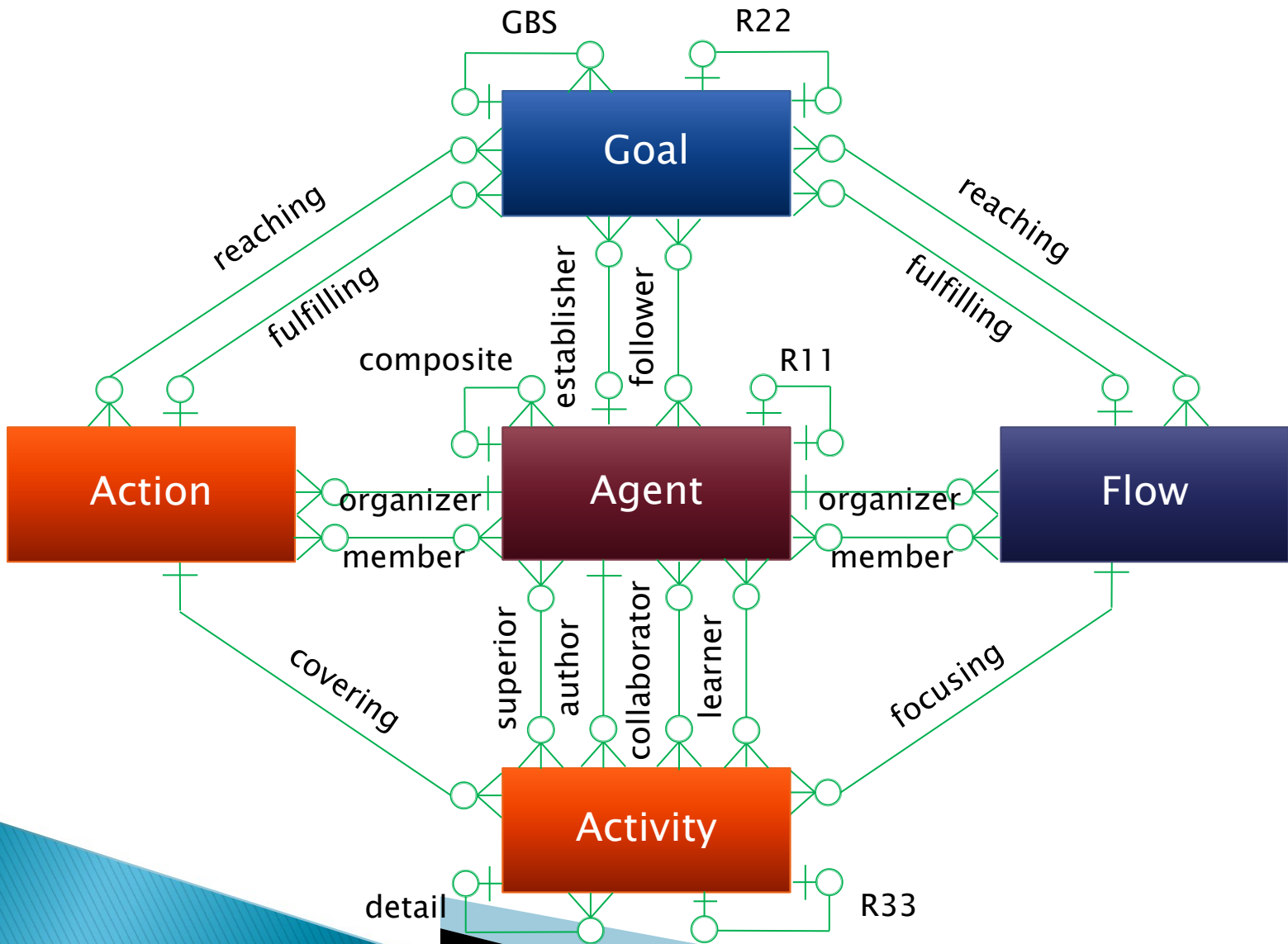
Diamond of Agent-Team Organization

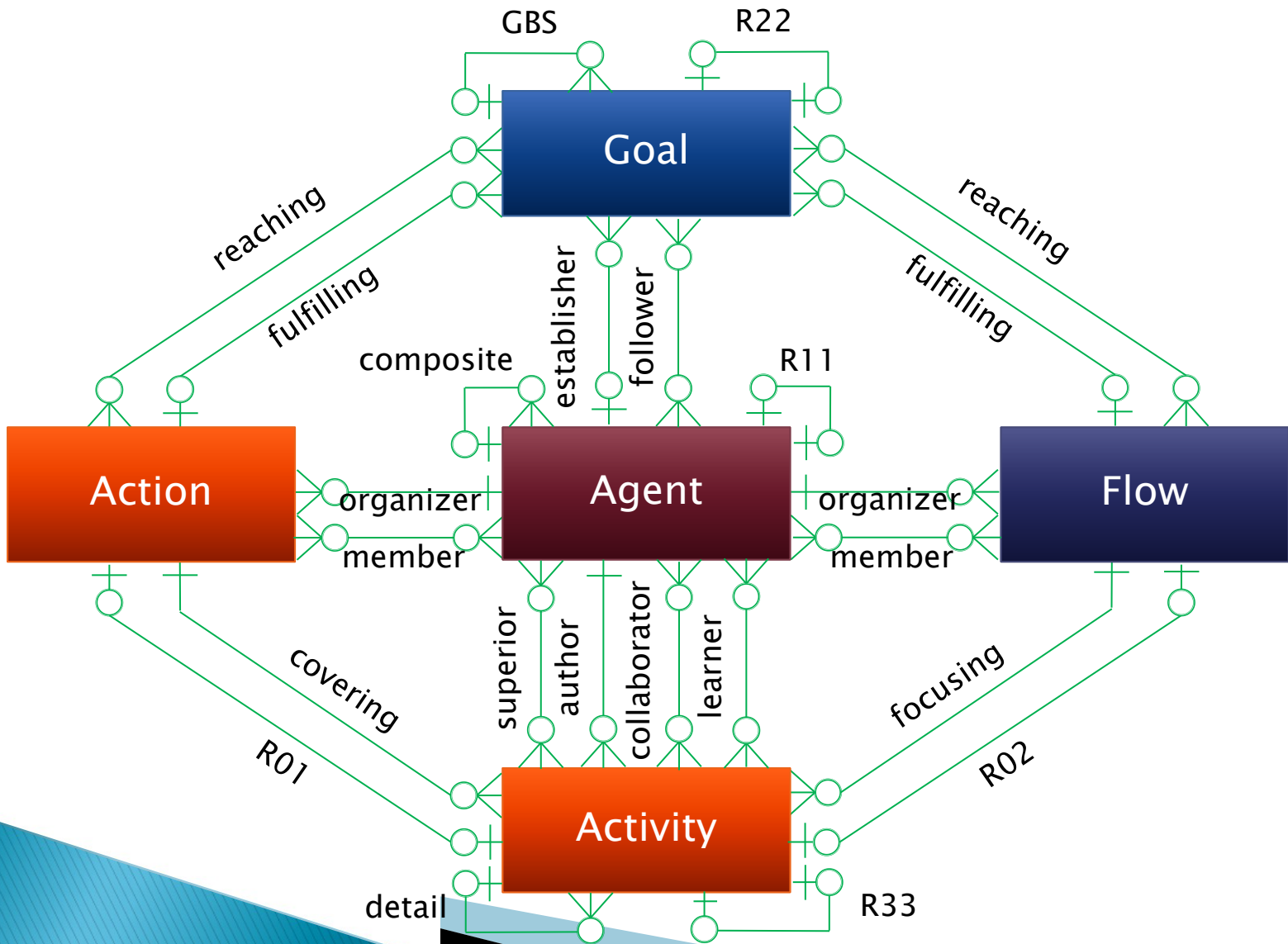








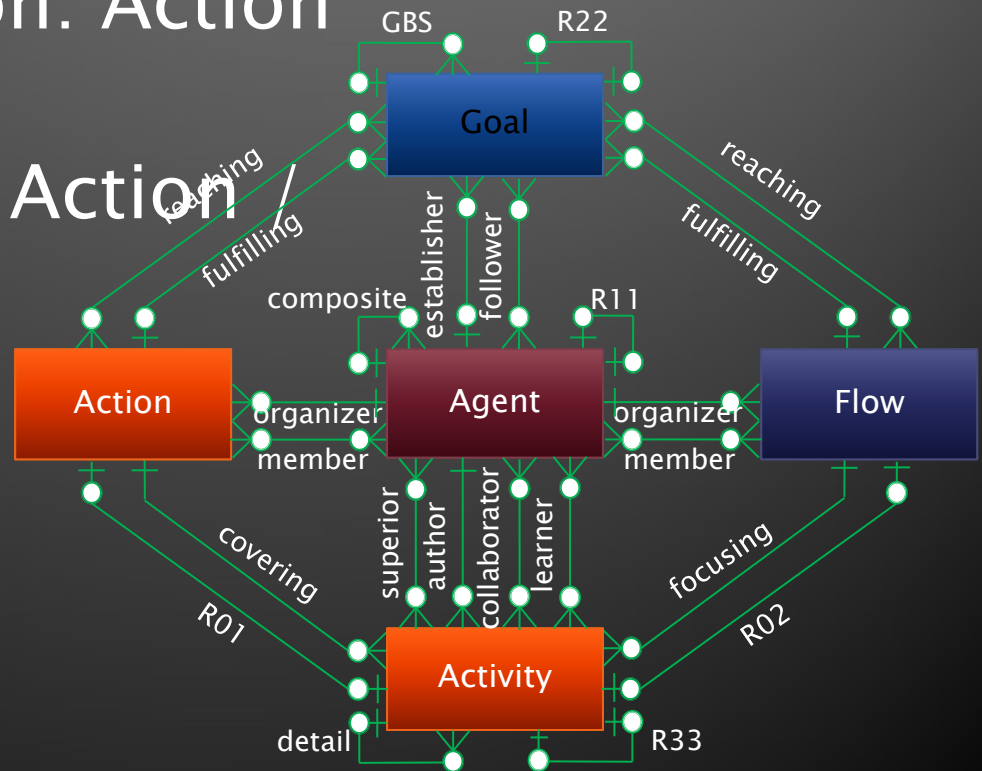




Diamond of Organization

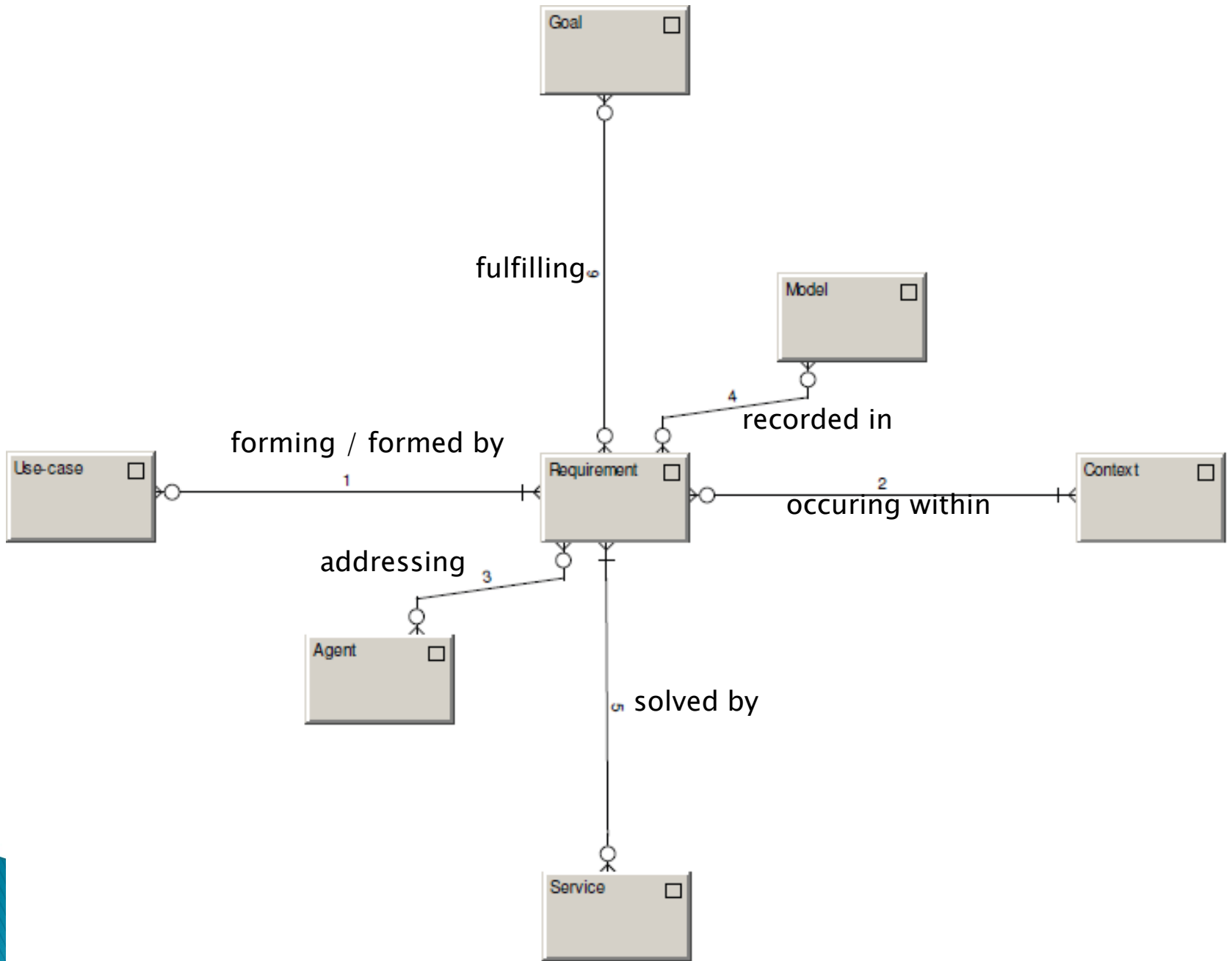
Summary

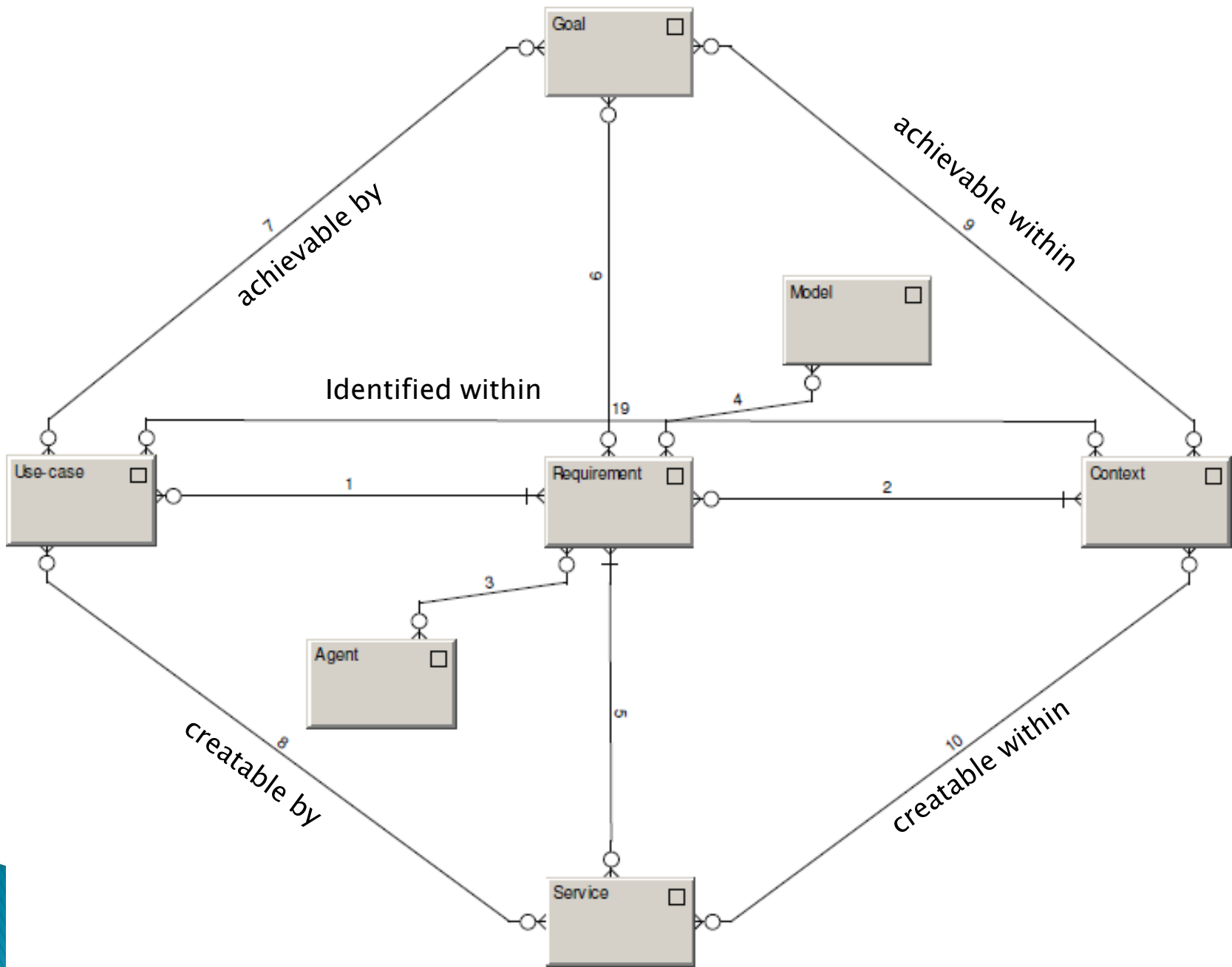
- ▶ Matrix-based organization: Action vs. Flow
- ▶ Activity vs. Action
- ▶ R-edges

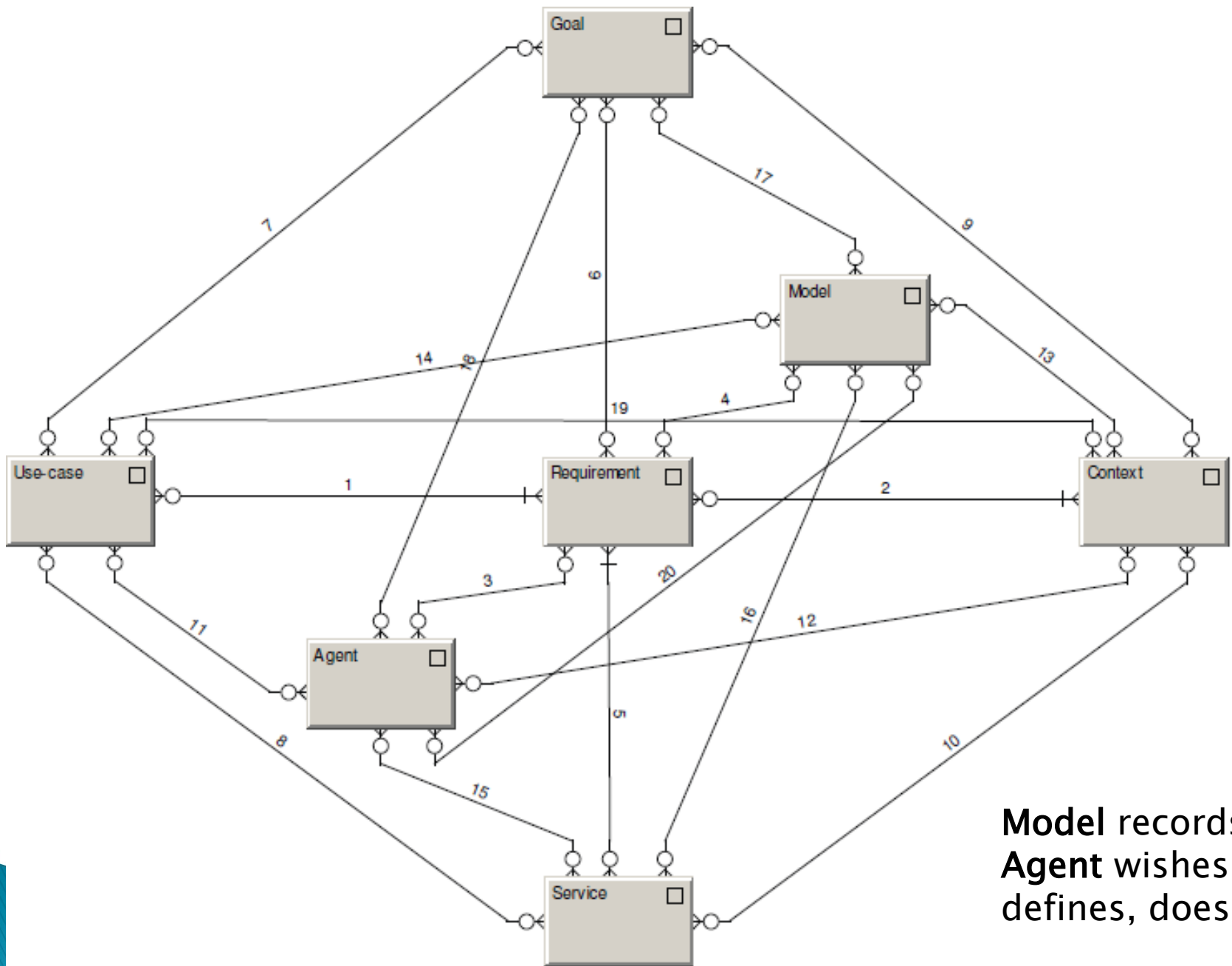


4th Diamond – DO

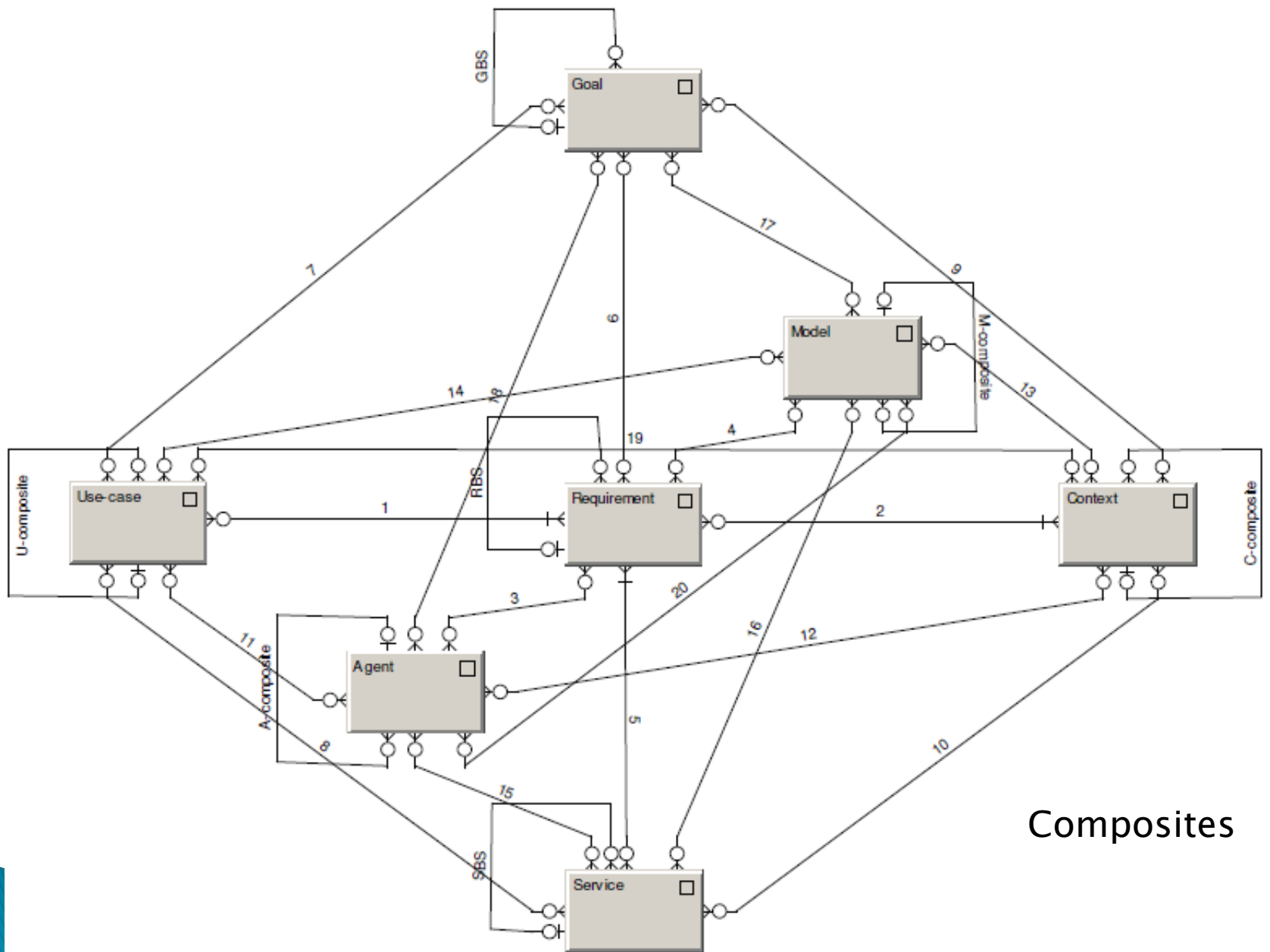




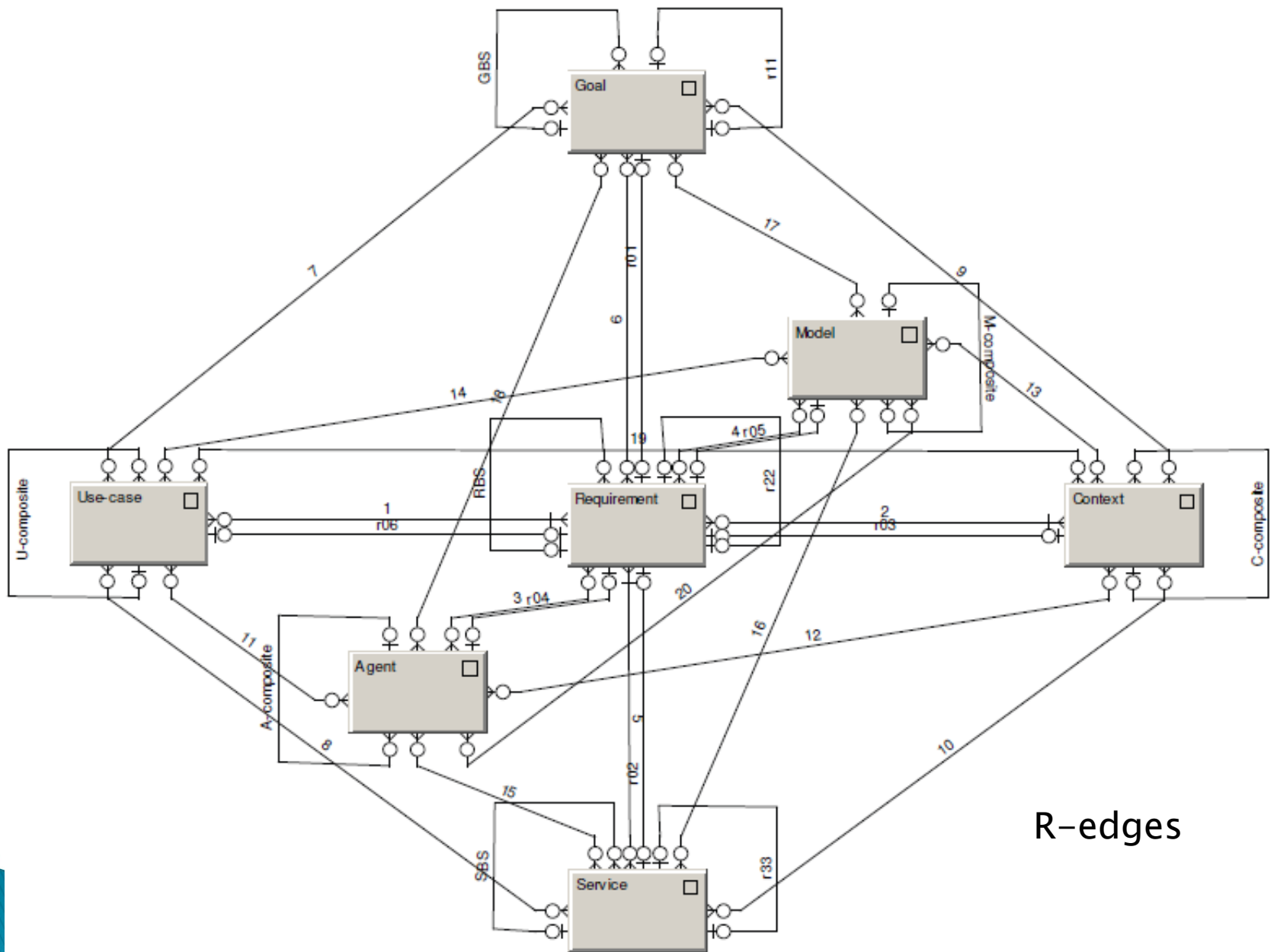




Model records.
Agent wishes,
 defines, does.



Composites



R-edges

Diamond of Predictive Behaviour

- ▶ Depicts the motivation of agents to DO
- ▶ Everything can be seen as a requirement
- ▶ Forming and being formed by behavioral patterns
- ▶ Models as a system memory

