Service Modeling II

© Leonard Walletzký

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

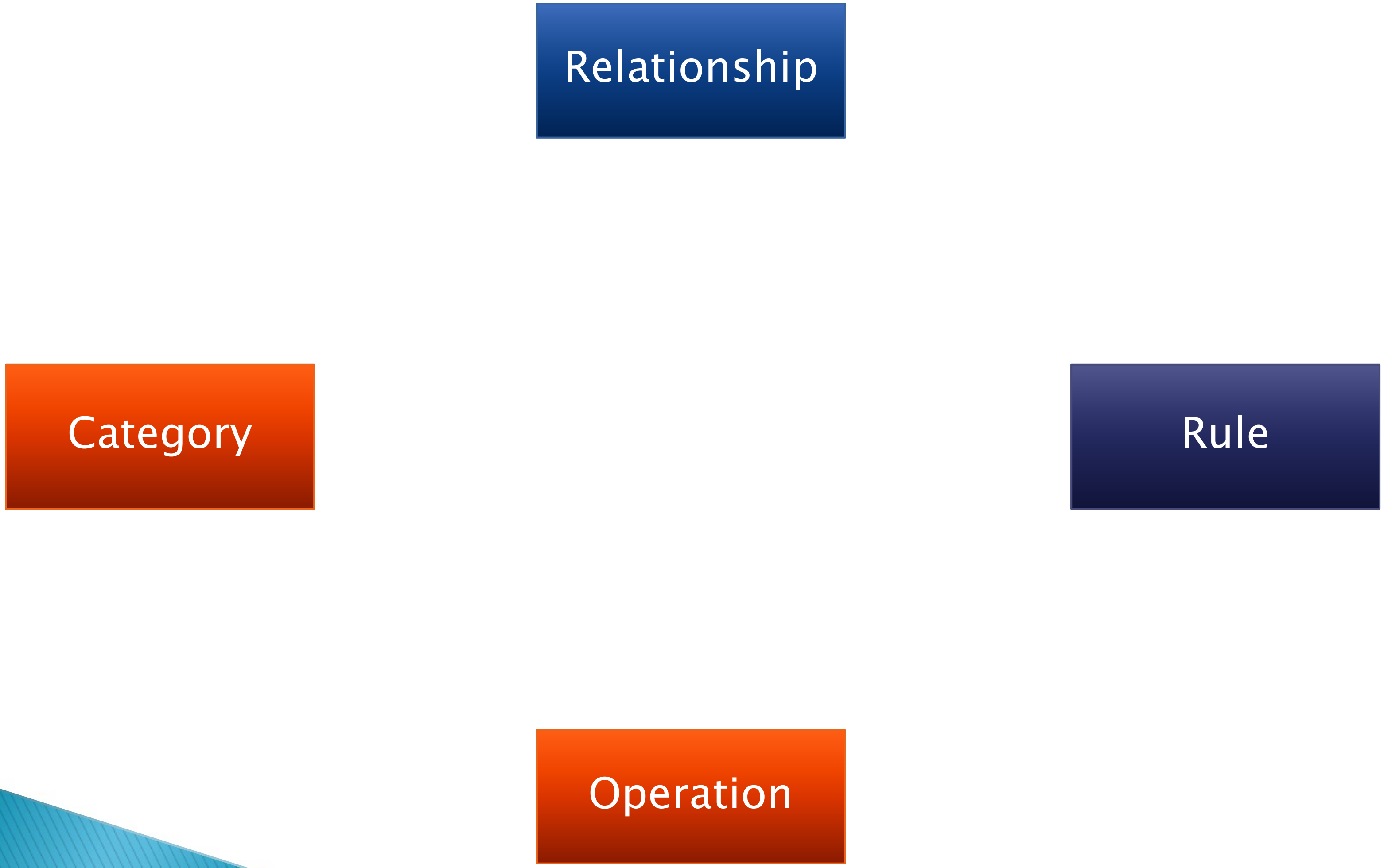
4 diamonds

- 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 variets 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?



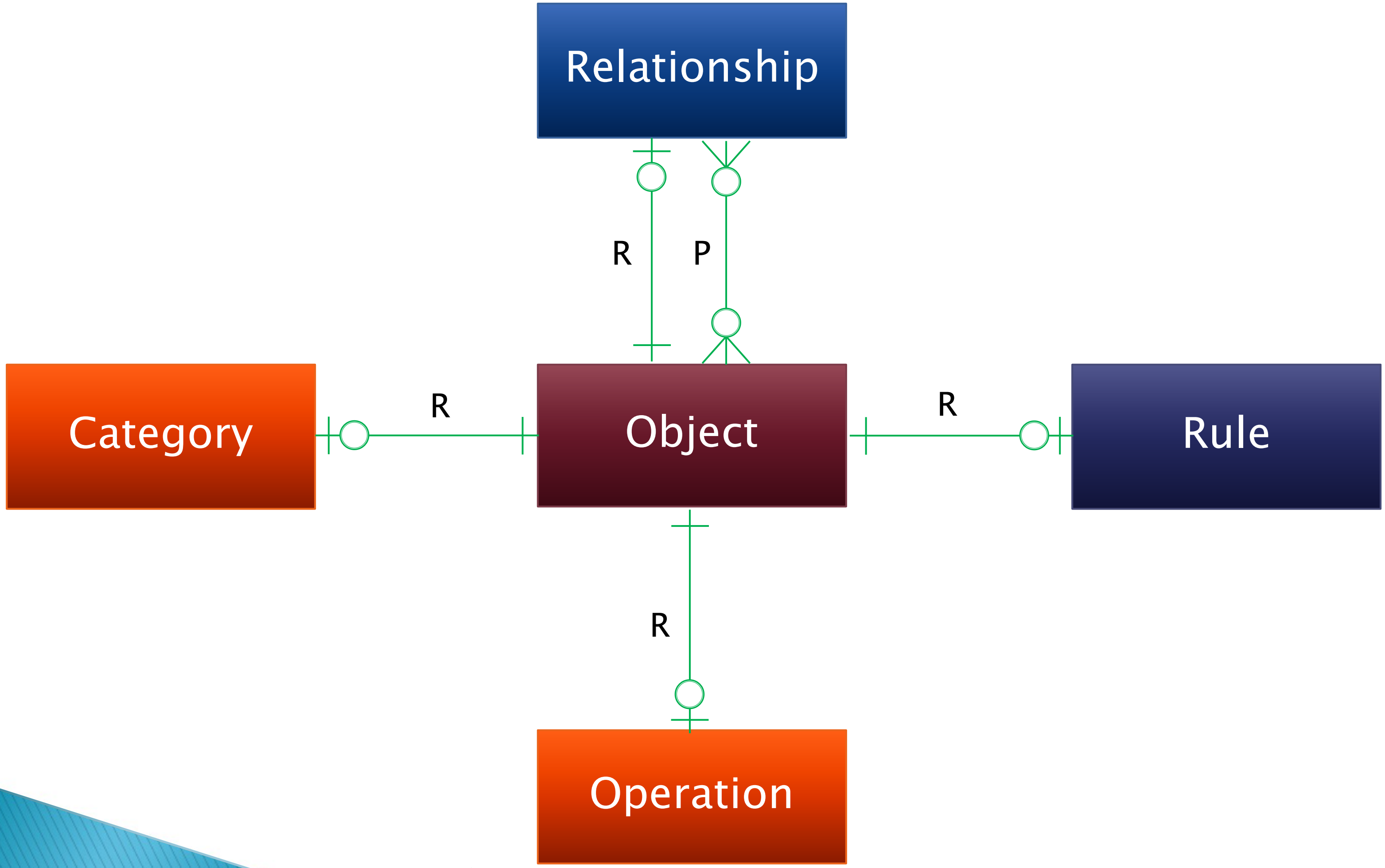
Relationship

Category

Rule

Operation

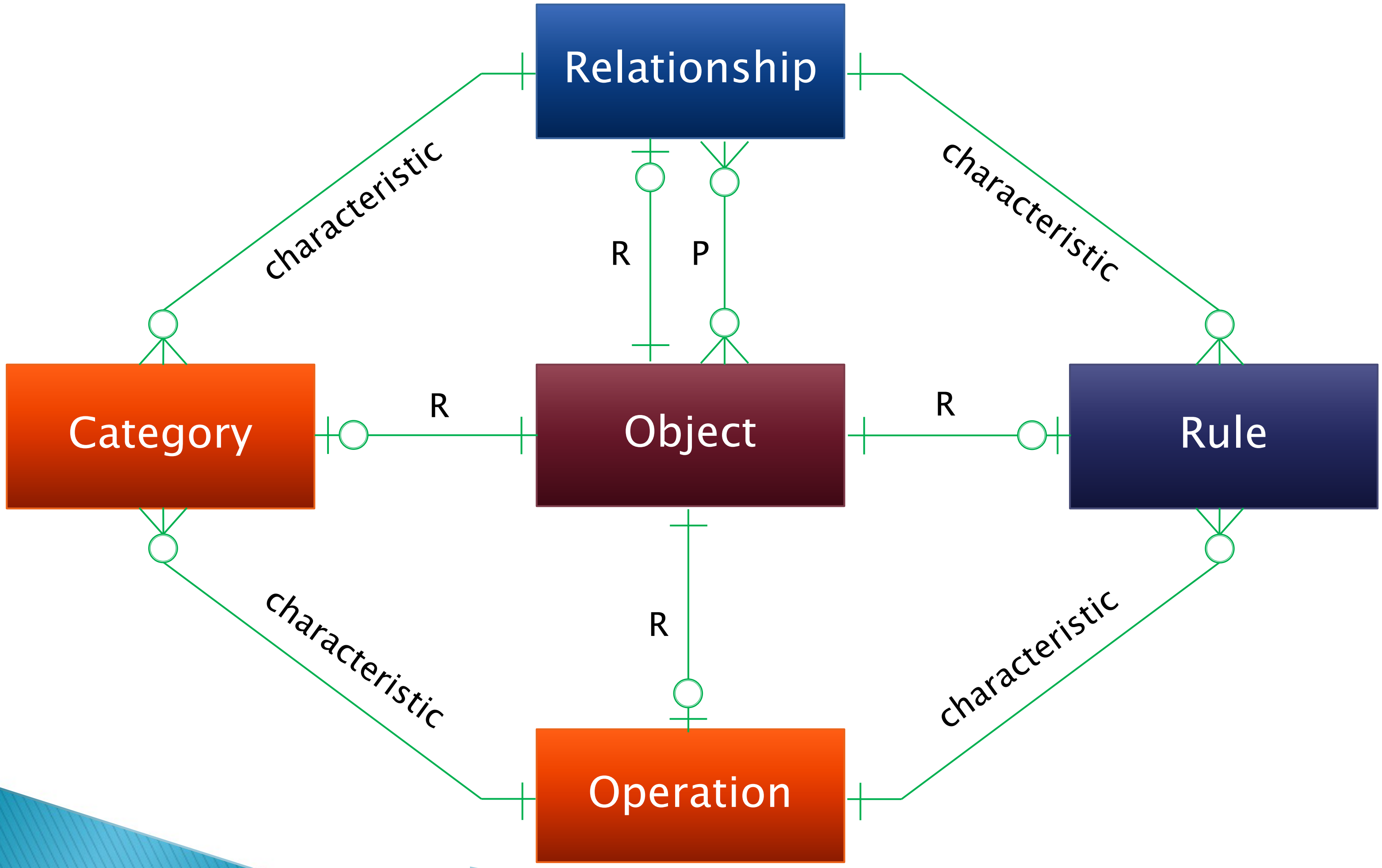
MENTION – USE duality



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 ⊥ 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.

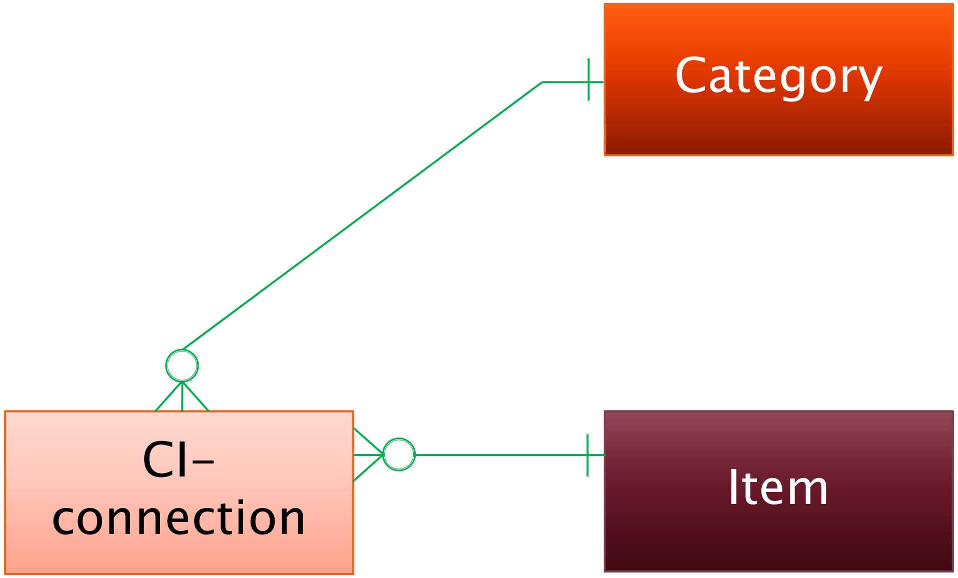
Container (#Category)

- 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 <Cn, Op>, where

     Cn ∈ (#Connection), Op ∈ (#Operation), and
  3. it holds about the operation Op that by means of the connection Cn 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 Cn is called a *defining connection* of this category, the operation Op is called a *defining operation* or an *evaluator* of this category.

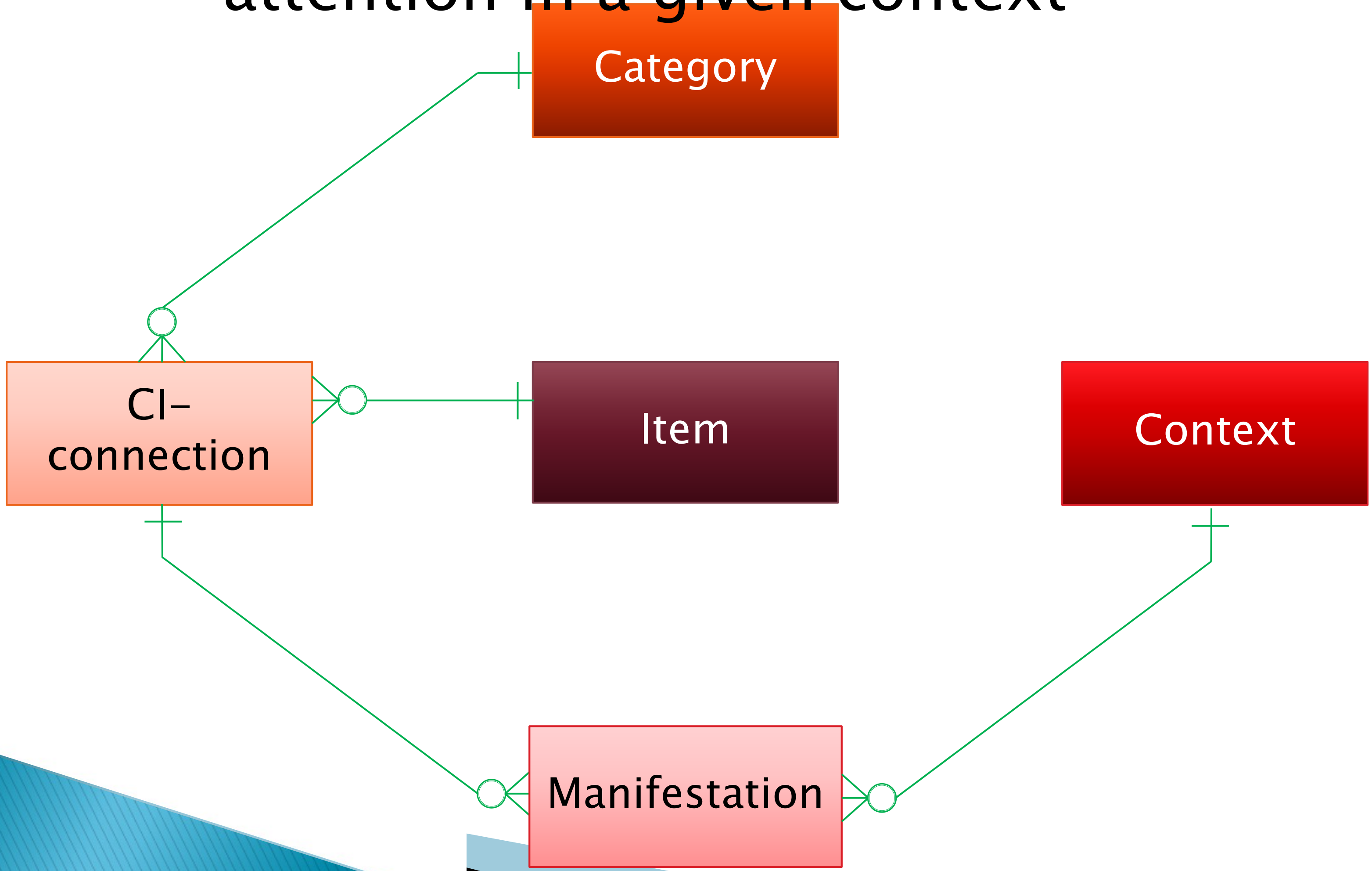
Certainty

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

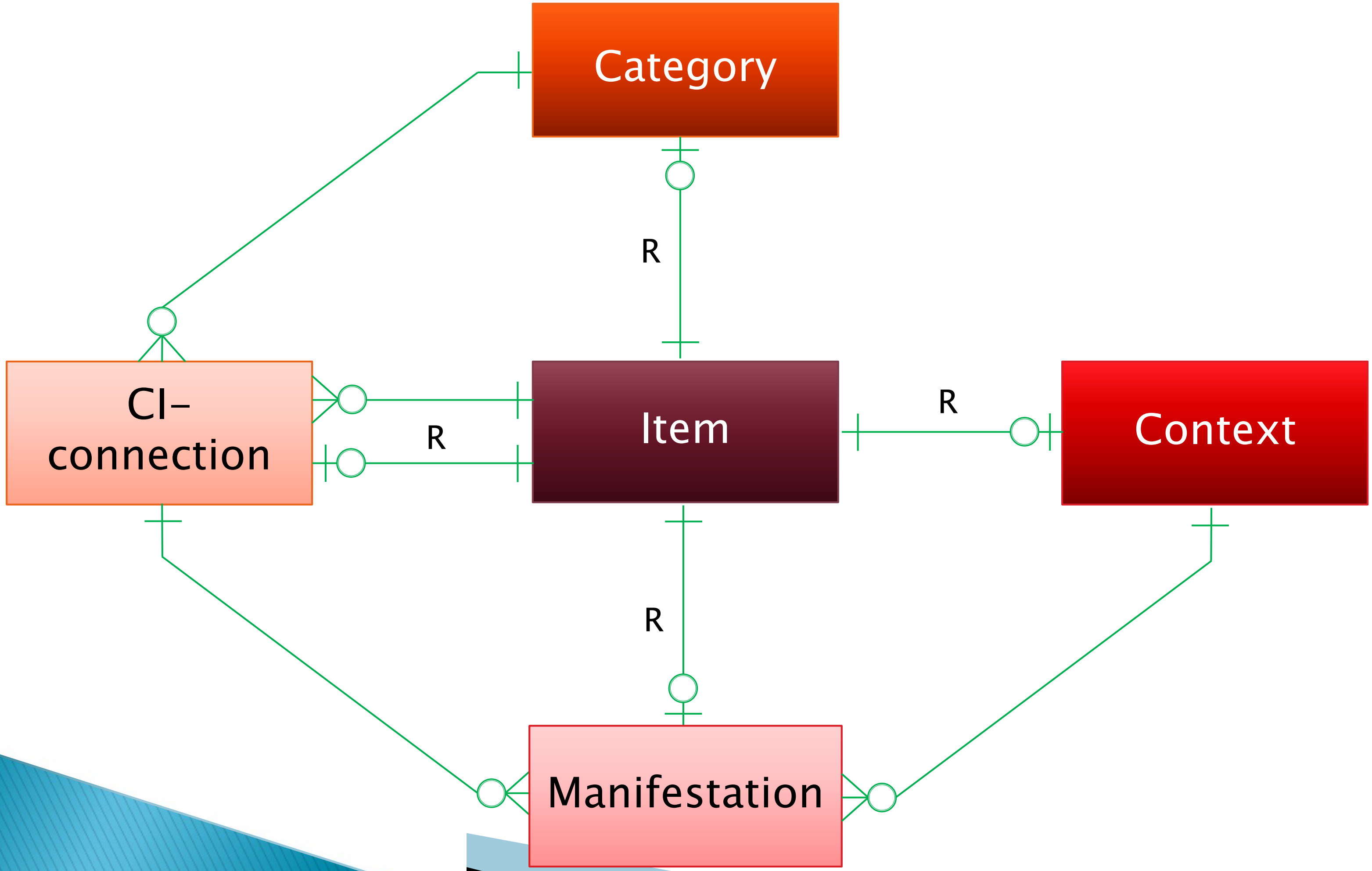


Attention

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

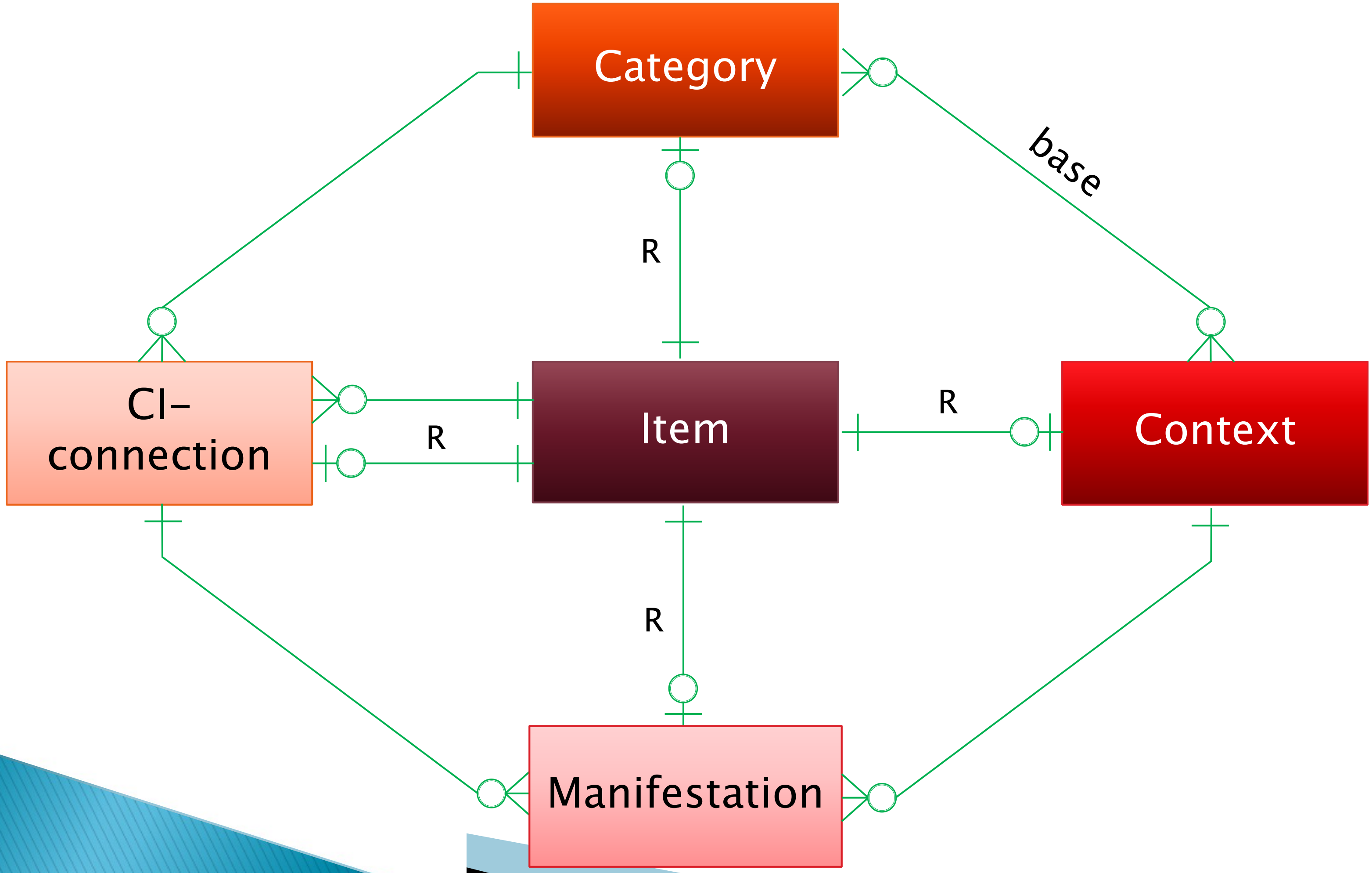


R-edges



Context base

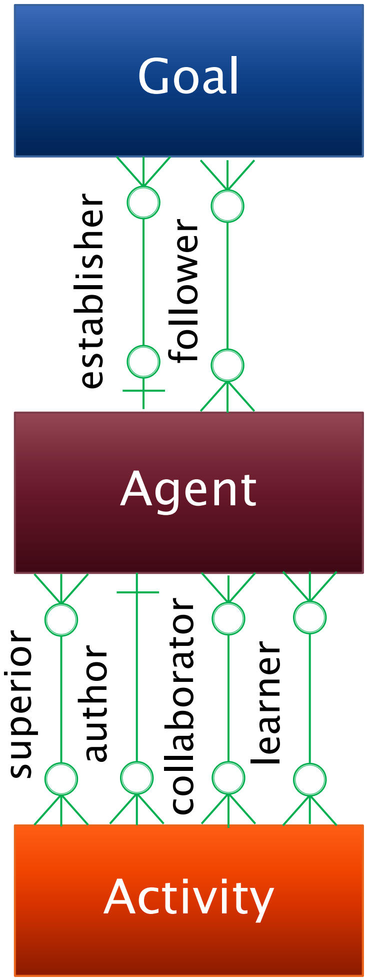
- Context serves as a model. The base edge defines the set of categories to classify its items to

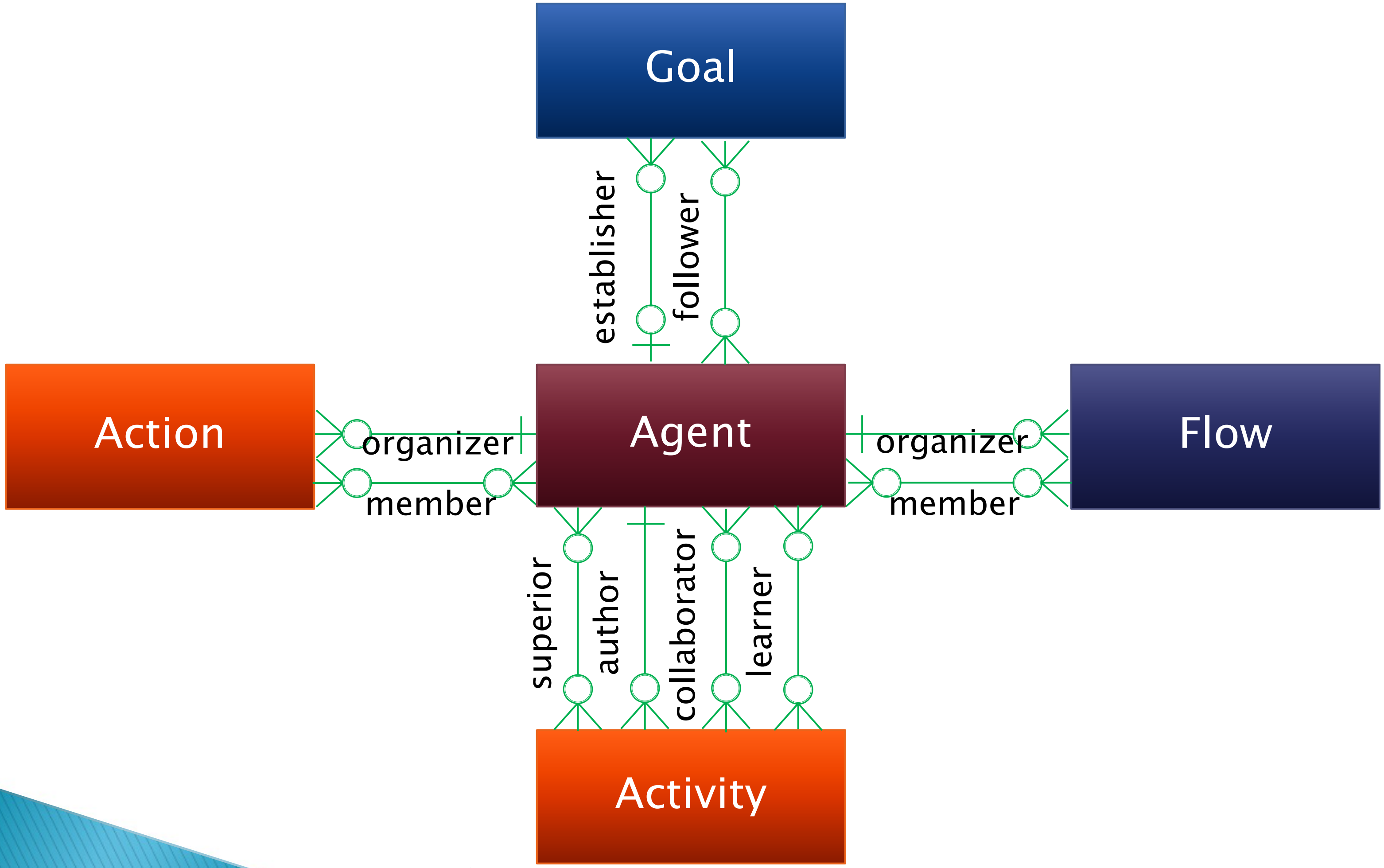


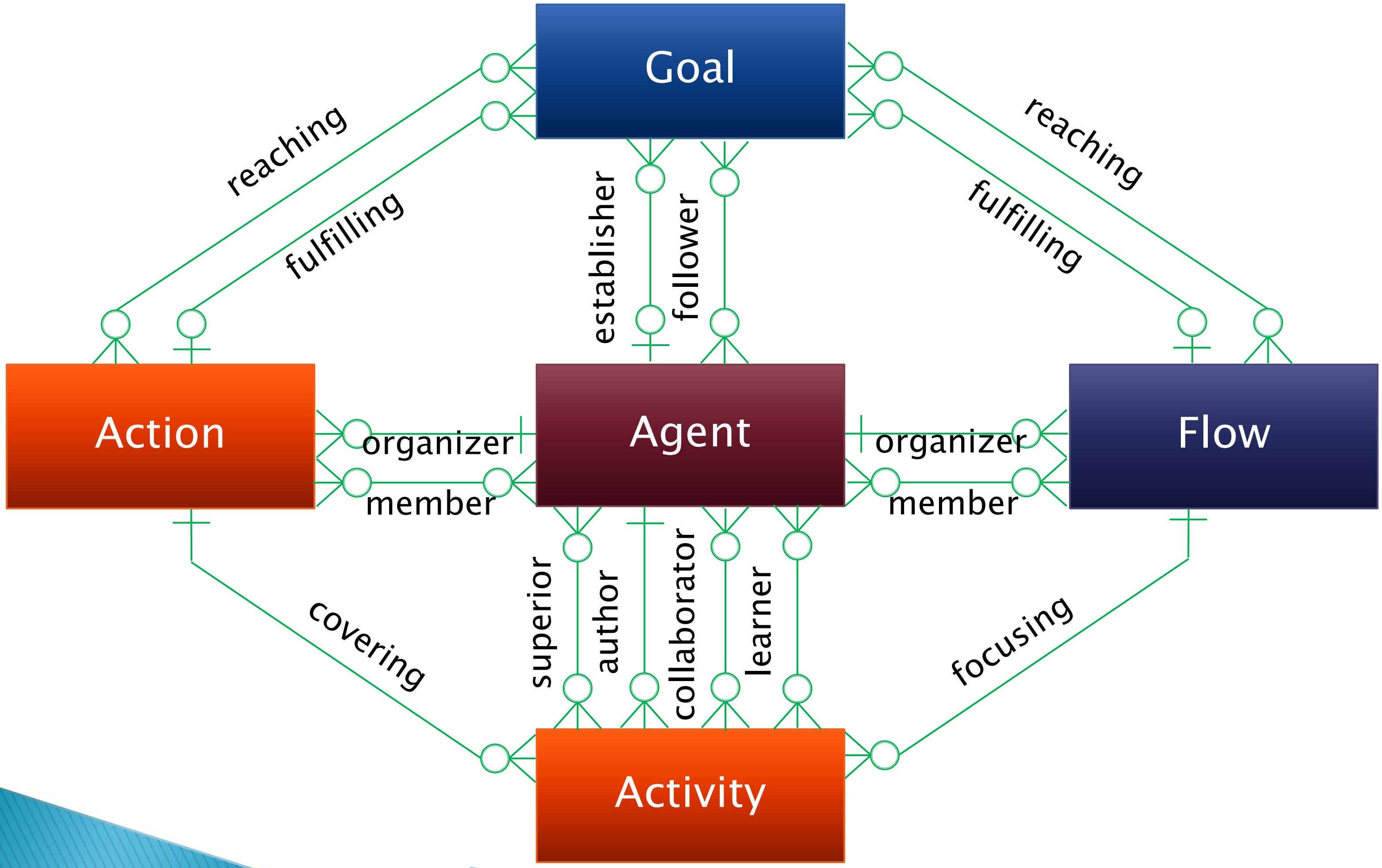
3rd diamond

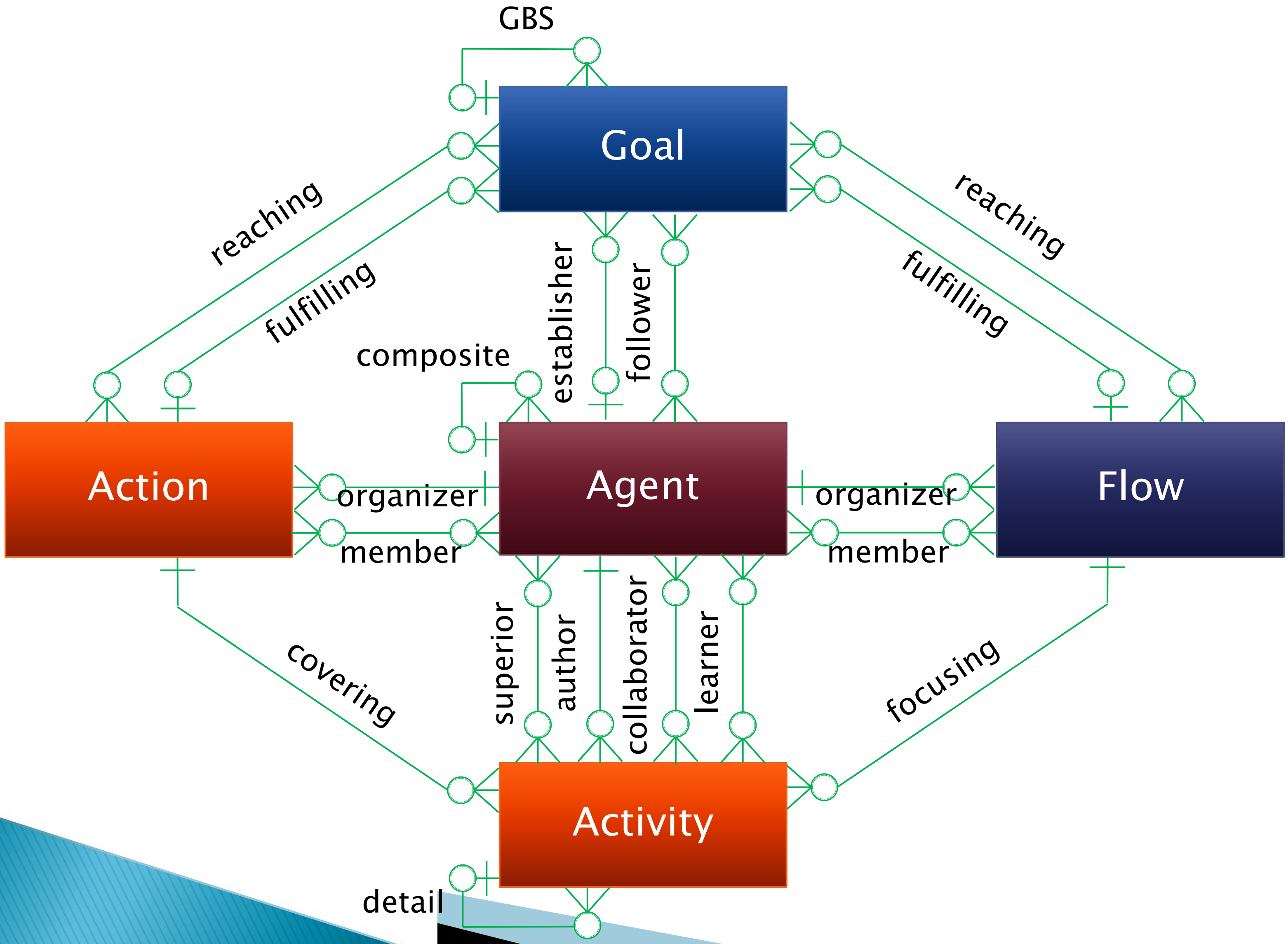
- Organizing diamond
- How is your life / position / work organized
- How can be some agent
  - Member of some team
  - Work on projects
  - Be educated or teach

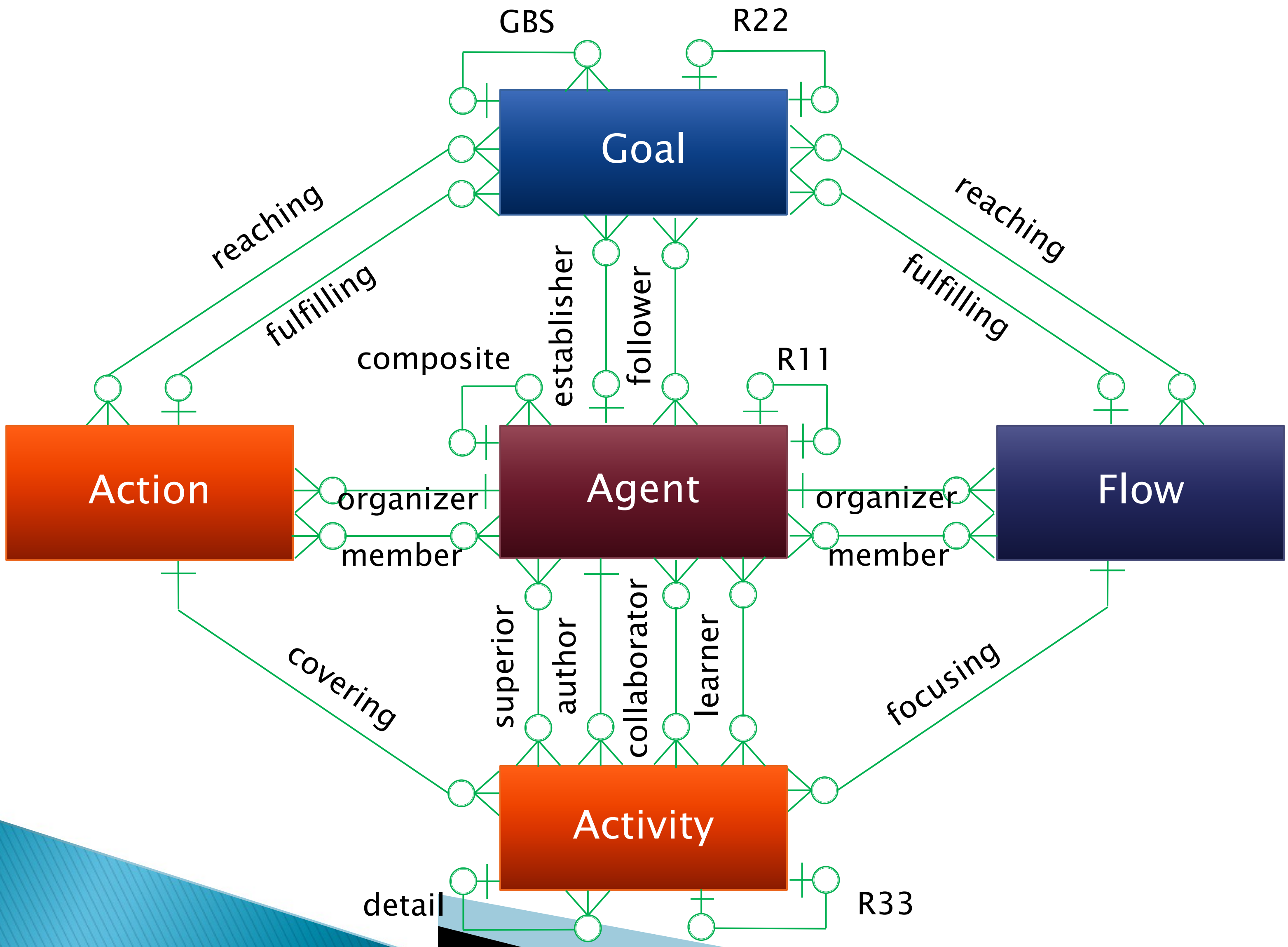
Diamond of Agent-Team Organization

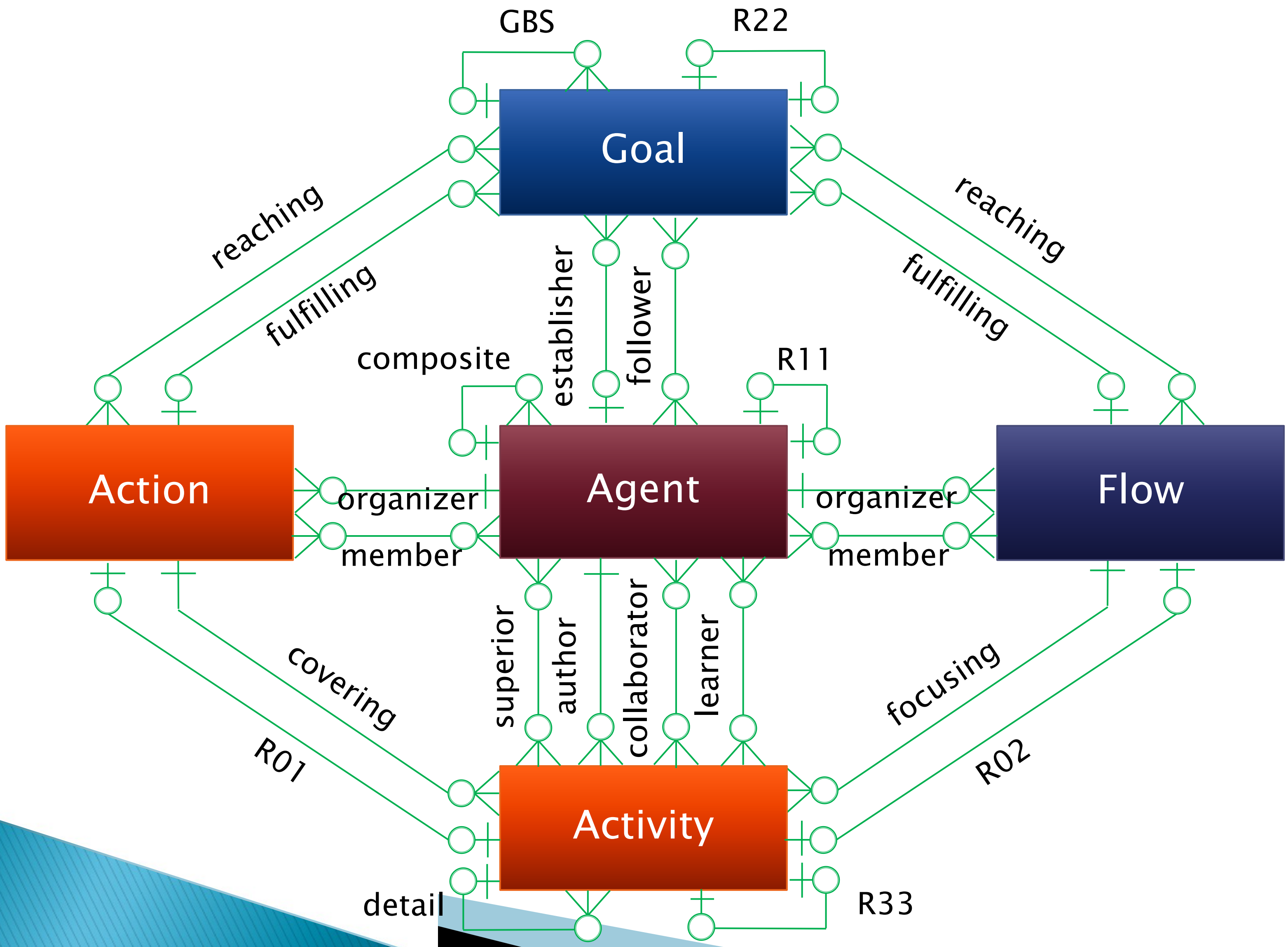



Goal
establisher
follower
Action
organizer
member
Agent
organizer
member
Flow
superior
author
collaborator
learner
Activity


Goal
reaching
fulfilling
reaching
fulfilling
establisher
follower
Action
organizer
member
Agent
organizer
member
Flow
superior
author
collaborator
learner
covering
focusing
Activity


GBS
Goal
reaching
fulfilling
reaching
fulfilling
establisher
follower
composite
Action
organizer
member
Agent
organizer
member
Flow
superior
author
collaborator
learner
covering
focusing
Activity
detail

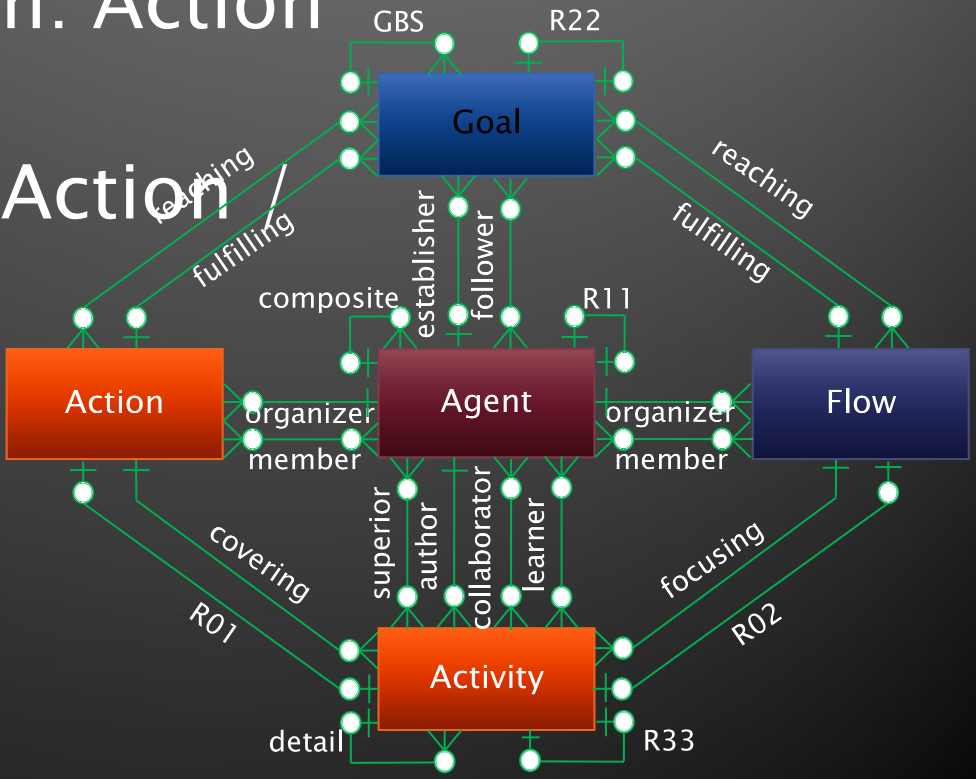

GBS
R22
Goal
reaching
fulfilling
reaching
fulfilling
establisher
follower
composite
R11
Action
Agent
Flow
organizer
member
organizer
member
superior
author
collaborator
learner
covering
focusing
Activity
detail
R33


GBS
R22
Goal
reaching
fulfilling
reaching
fulfilling
establisher
follower
composite
R11
Action
Agent
Flow
organizer
member
organizer
member
superior
author
collaborator
learner
covering
R01
focusing
R02
Activity
detail
R33

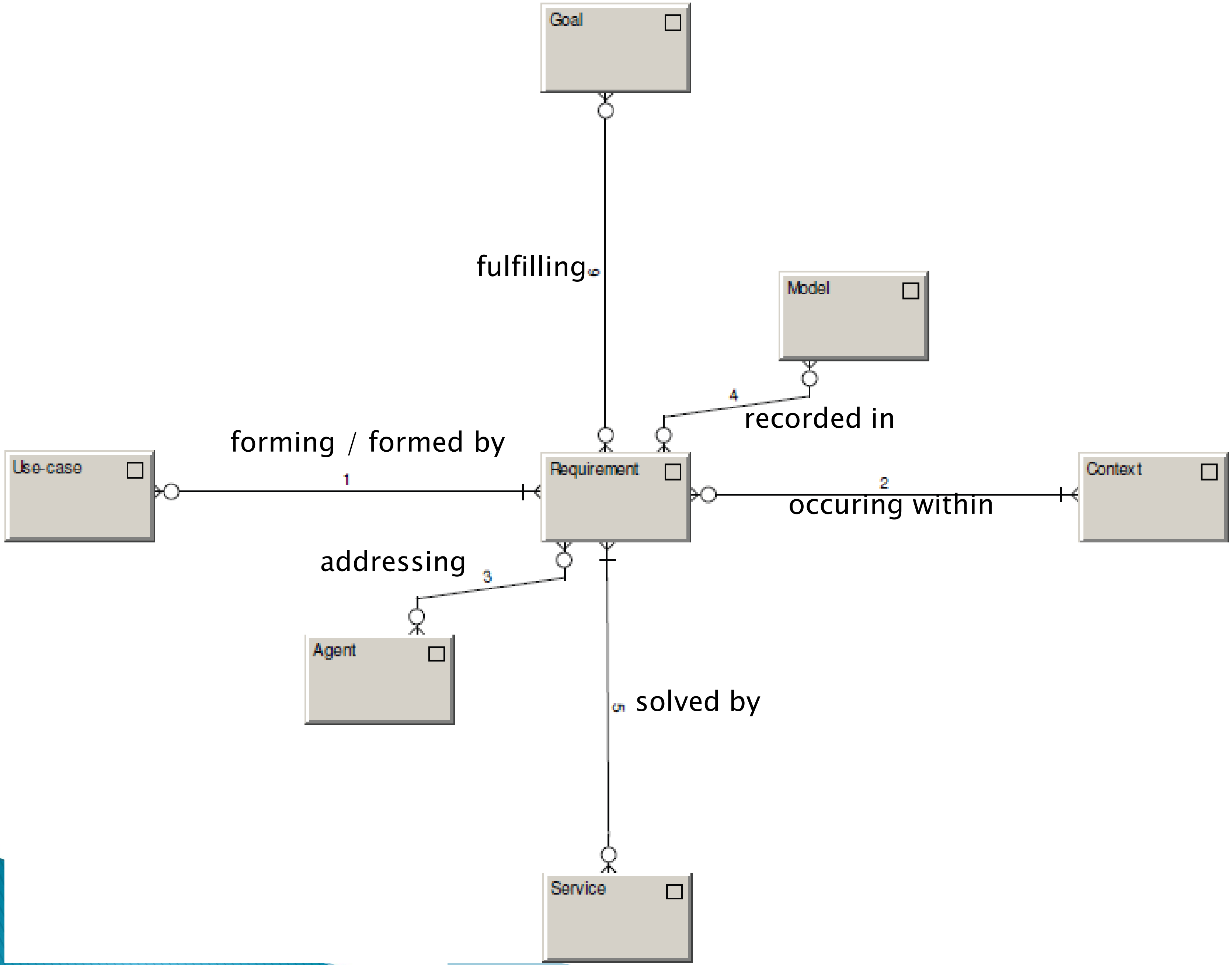
Diamond of Organization

Summary

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



4th Diamond – DO



fulfilling

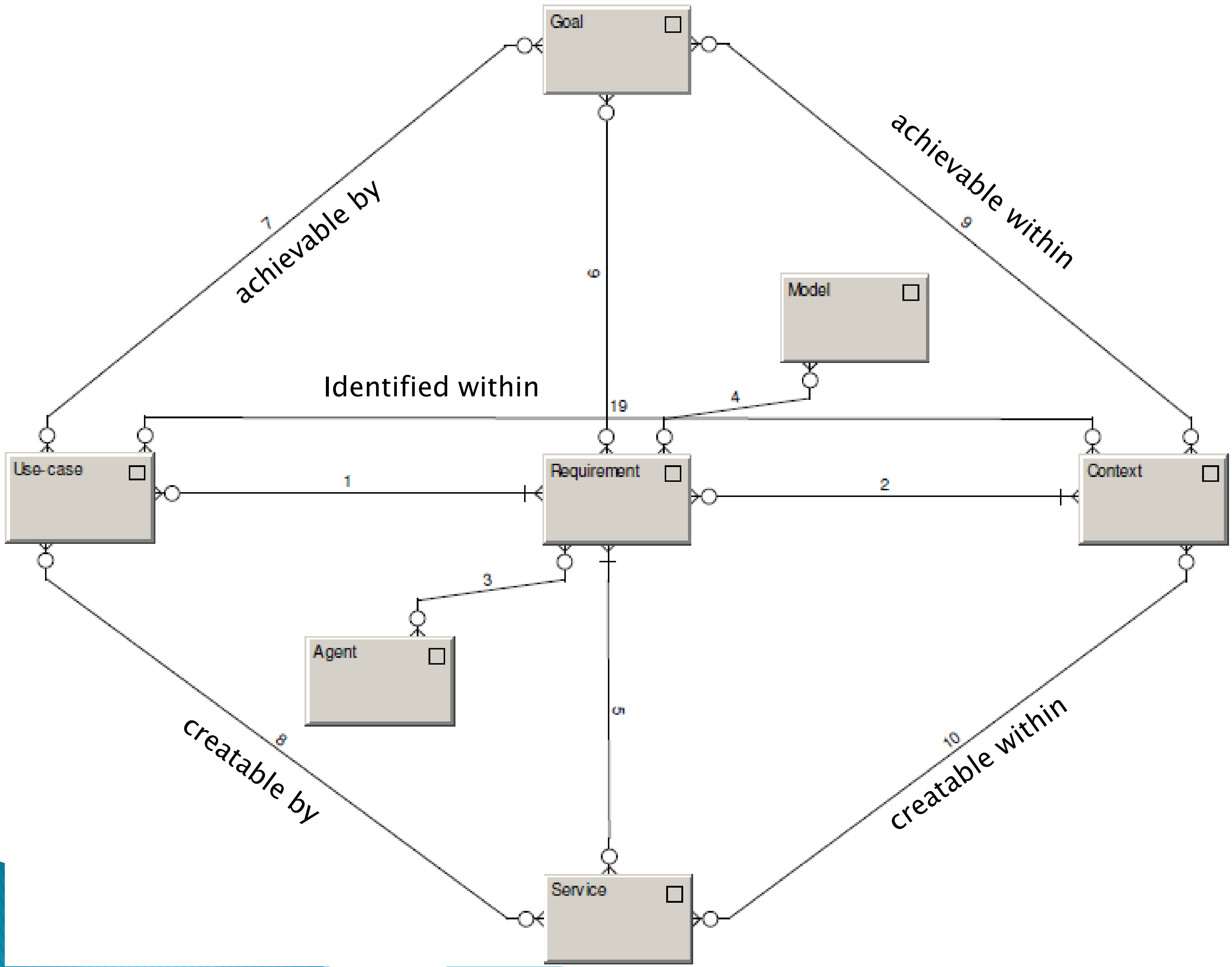
recorded in

forming / formed by

occuring within

addressing

solved by



Goal □

achievable by

7

achievable within

9

6

Model □

Identified within

19

4

Use-case □

1

Requirement □

2

Context □

3

Agent □

5

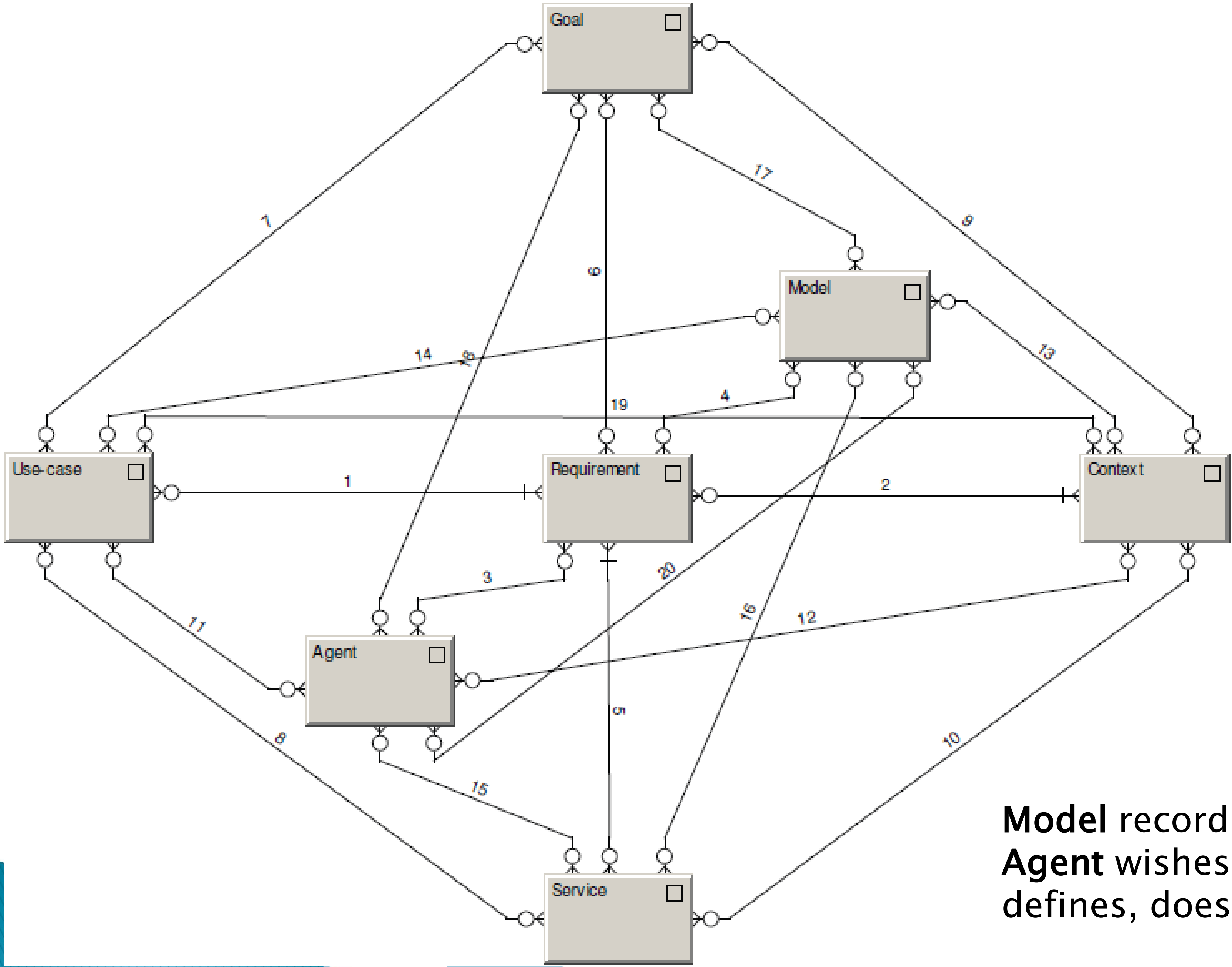
creatable by

8

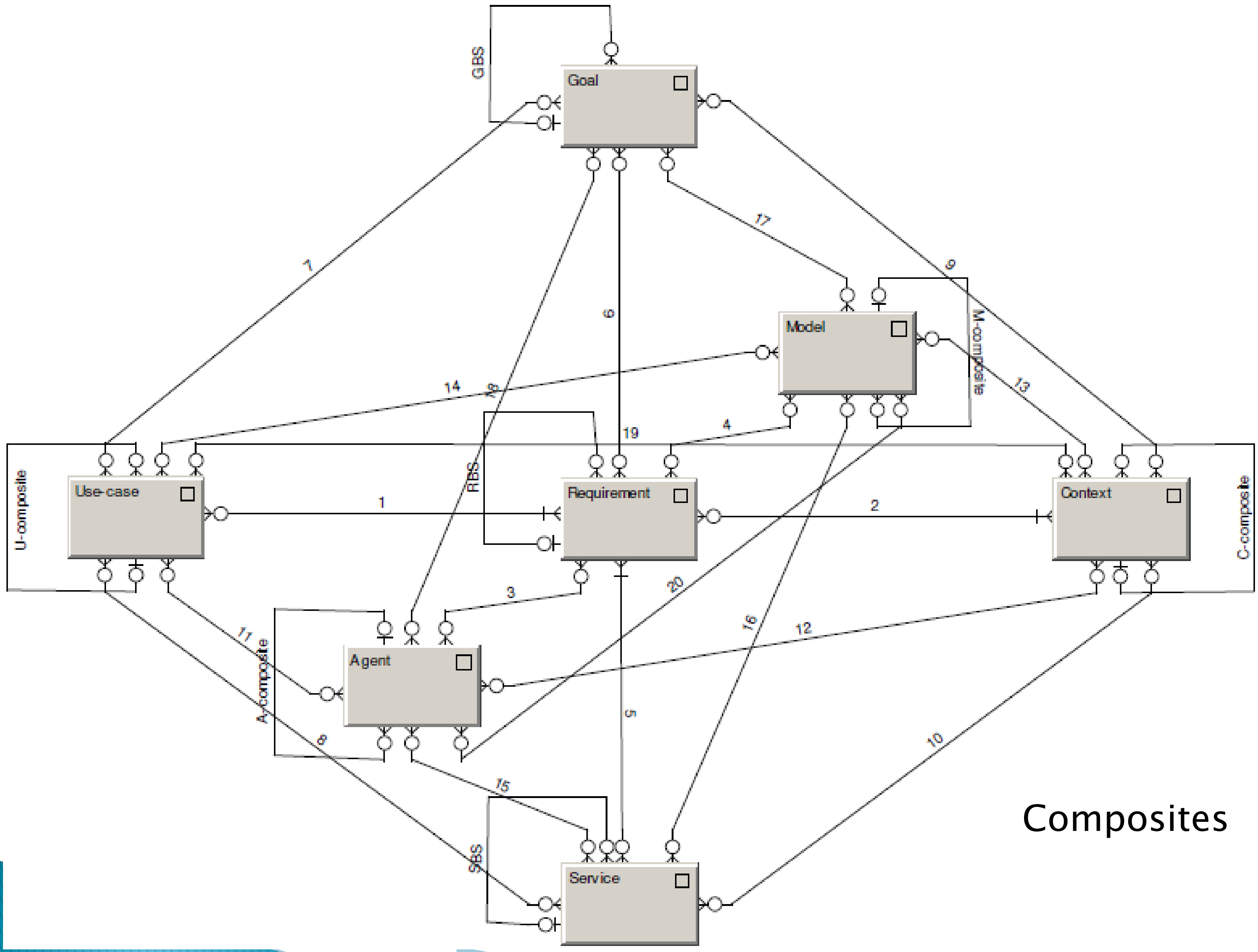
creatable within

10

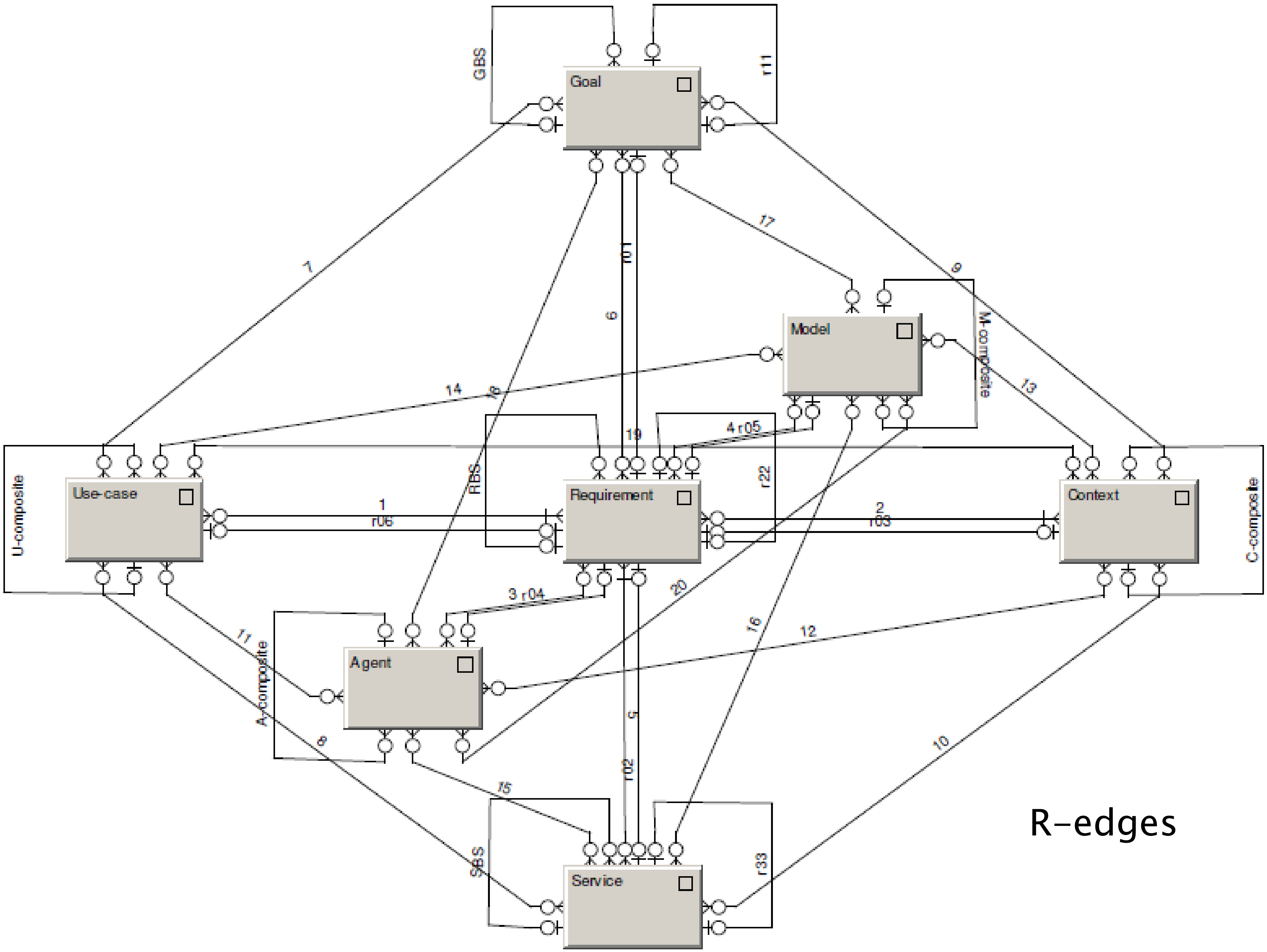
Service □



Model records.
Agent wishes, defines, does.


Goal
GBS
Model
M-composite
Use-case
U-composite
Requirement
RBS
Context
C-composite
Agent
A-composite
Service
SBS
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

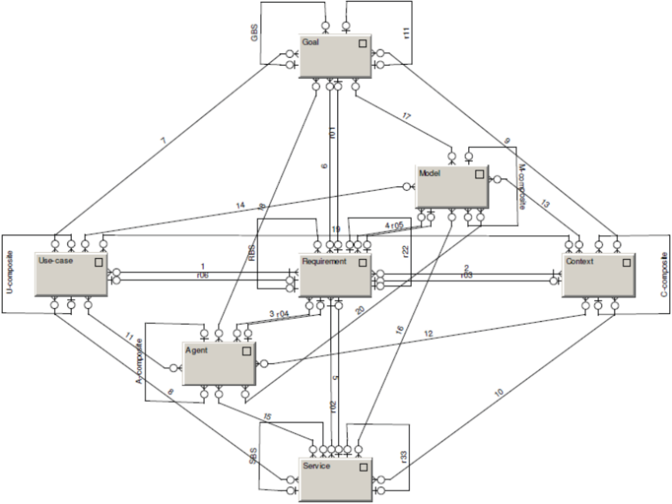
Composites


Goal
GBS
r11
7
r01
6
17
9
Model
M-composite
14
18
13
19
4 r05
r22
Use-case
U-composite
1
r06
RBS
Requirement
2
r03
Context
C-composite
3 r04
20
Agent
A-composite
11
16
12
5
8
r02
15
10
SBS
Service
r33

R-edges

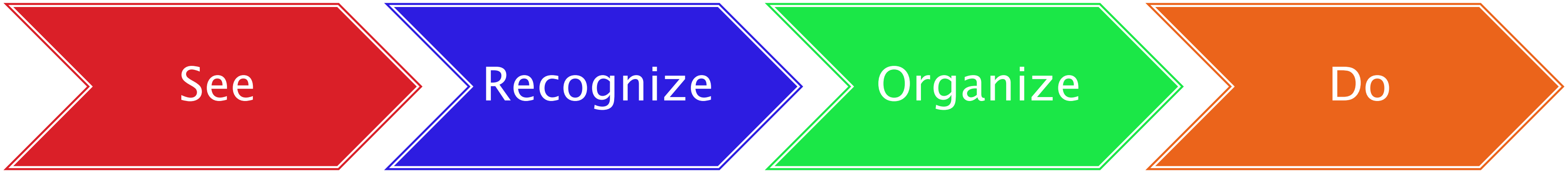
Diamond of Predictive Behaviour

- 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



Diamond-Path Framework

Overview

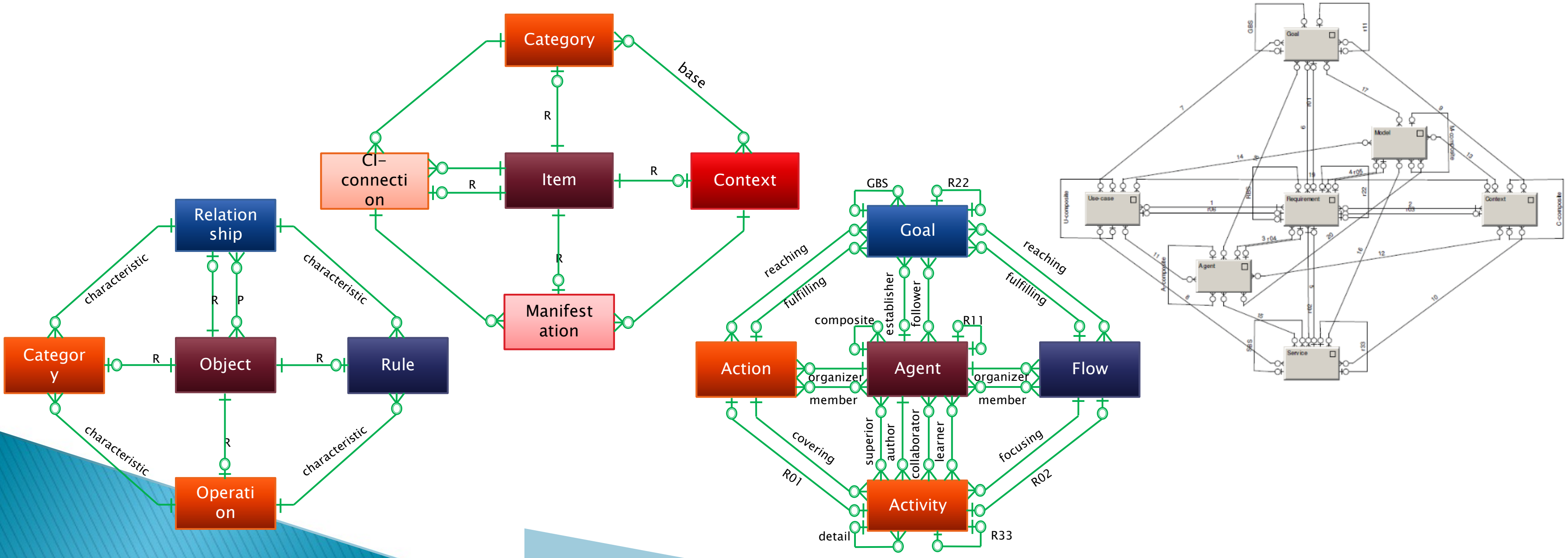


Attention Focussing

Cognitive Elements

Agent-Team Organization

Predictive Behaviour



Reflection

- Do you find it interesting?
- And useful?
- Why has it remained a pure theoretical concept?
- Is it too complex?

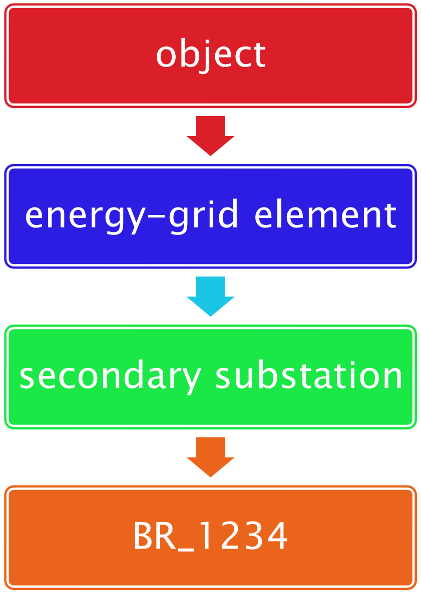
Follow-up Efforts

- Several academical works, none has made it to the business practice

➨ ...with one exception

GridMind

- Simulation environment designed to benchmark technological solutions of future-energy grids
- Combines objects on various levels of abstraction



- Comprises a number of mental contexts: distribution network, communication network, information scope, devices, technical processes, ...
- Very complex, difficult environment – worth systematical thinking