Behavior Modeling

(based on Alistair Cockburn book)

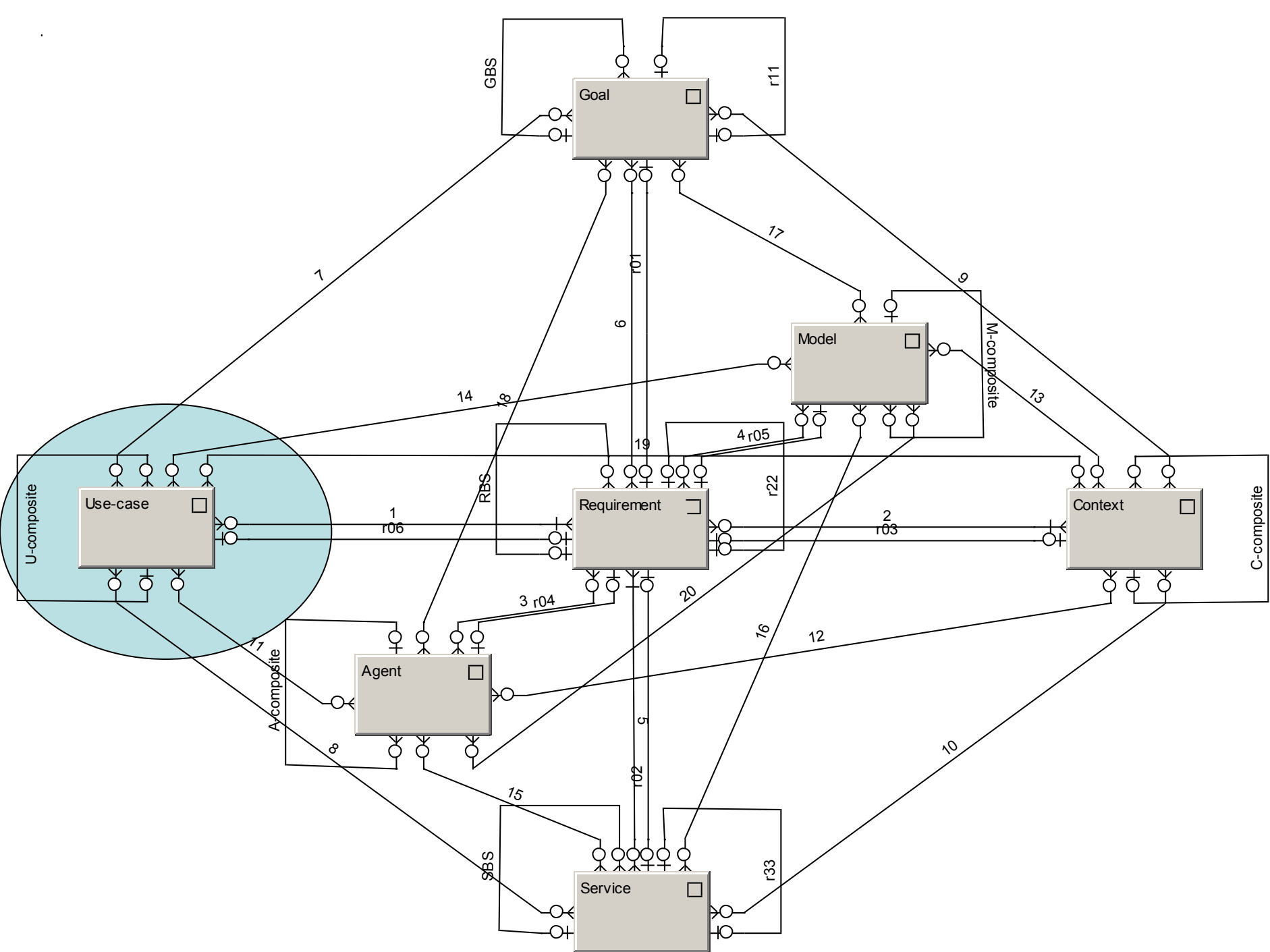## PA116 – L10

(c) Zdenko Staníček, Sept 2010

INVESTMENTS IN EDUCATION DEVELOPMENT

Use Cases

- Philosophy

- Principles

- Recommendations

Alistair Cockburn:
The Use-Case is a contract for System behavior

How to describe contract for system behavior?

- Diagram
- Diagram with NL descriptions
- NL
- NL constrained by several rules
- Artificial Language (BPML, BPMN, BPEL, …)

- What is the best way?

How to describe contract for system behavior

- NL constrained by several rules
- …coupled with diagram, if necessary

- Again in a form of a model
- … a special model, purposely not included into other models from Diam4
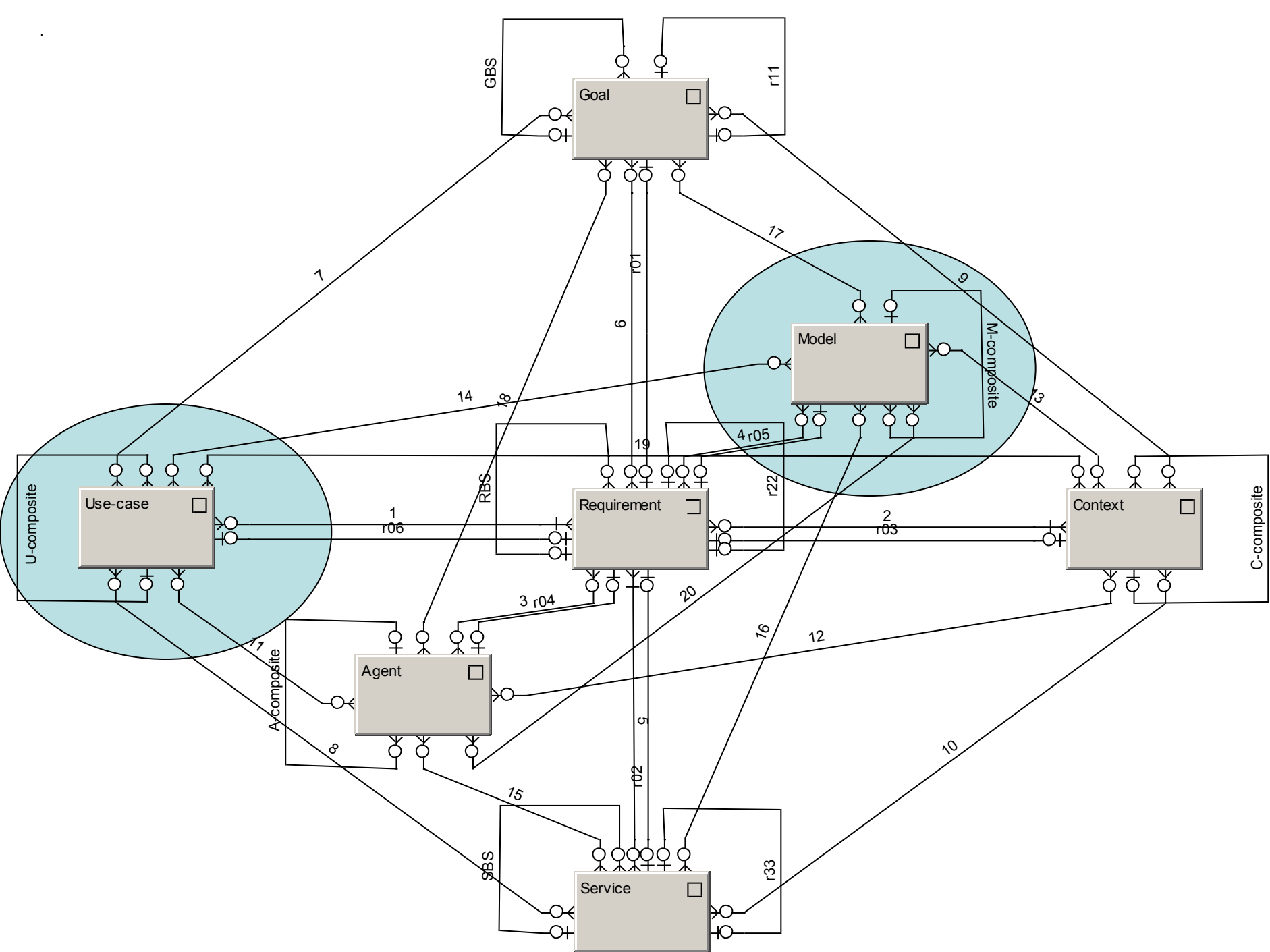
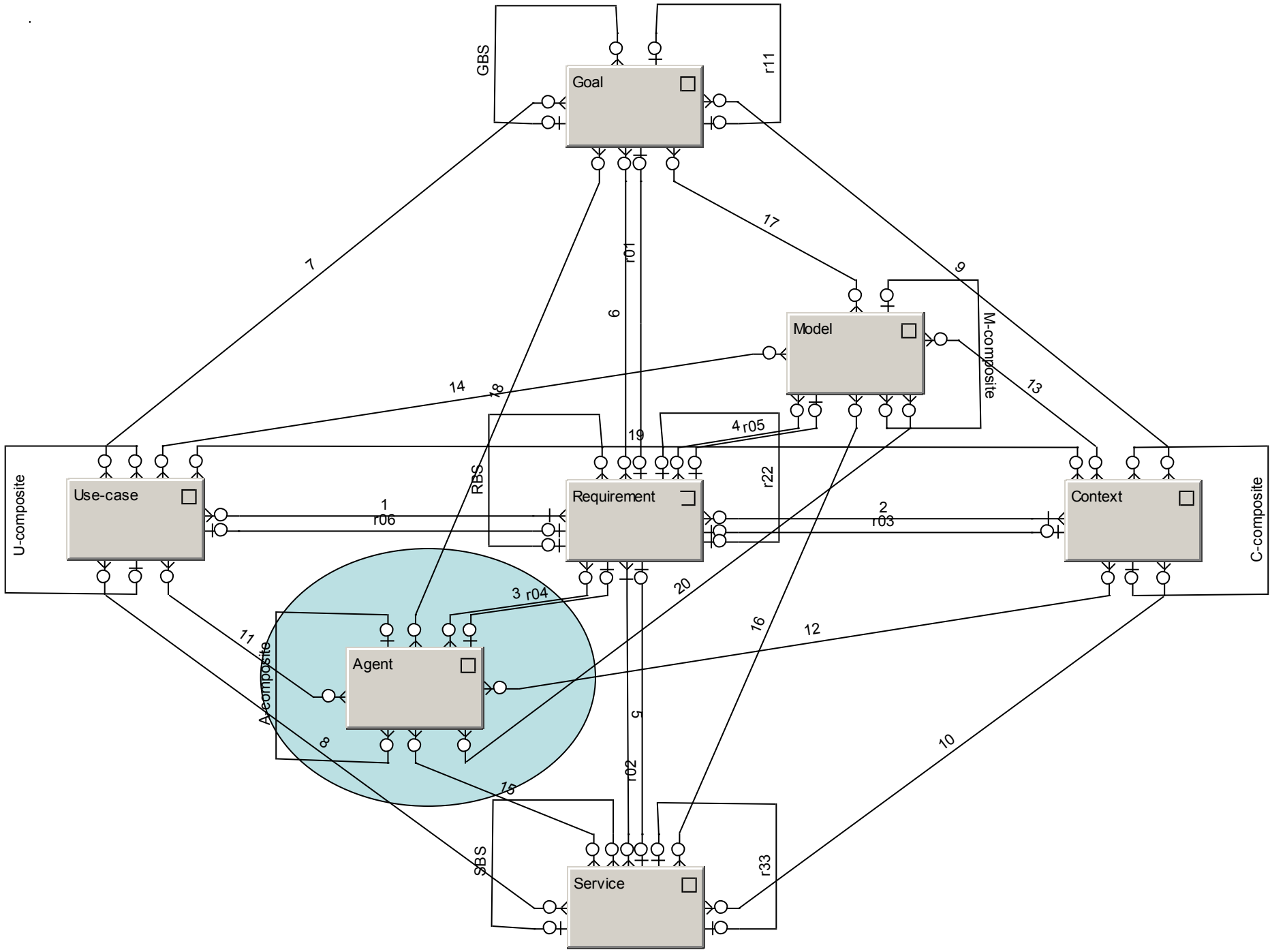# Key questions of Use Case writing
according to Alistair Cockburn

- Scope
  - What is really the system under discussion (SuD)?

- Primary Actor
  - Who has the goal?

- Level
  - How high- or low-level is the goal?
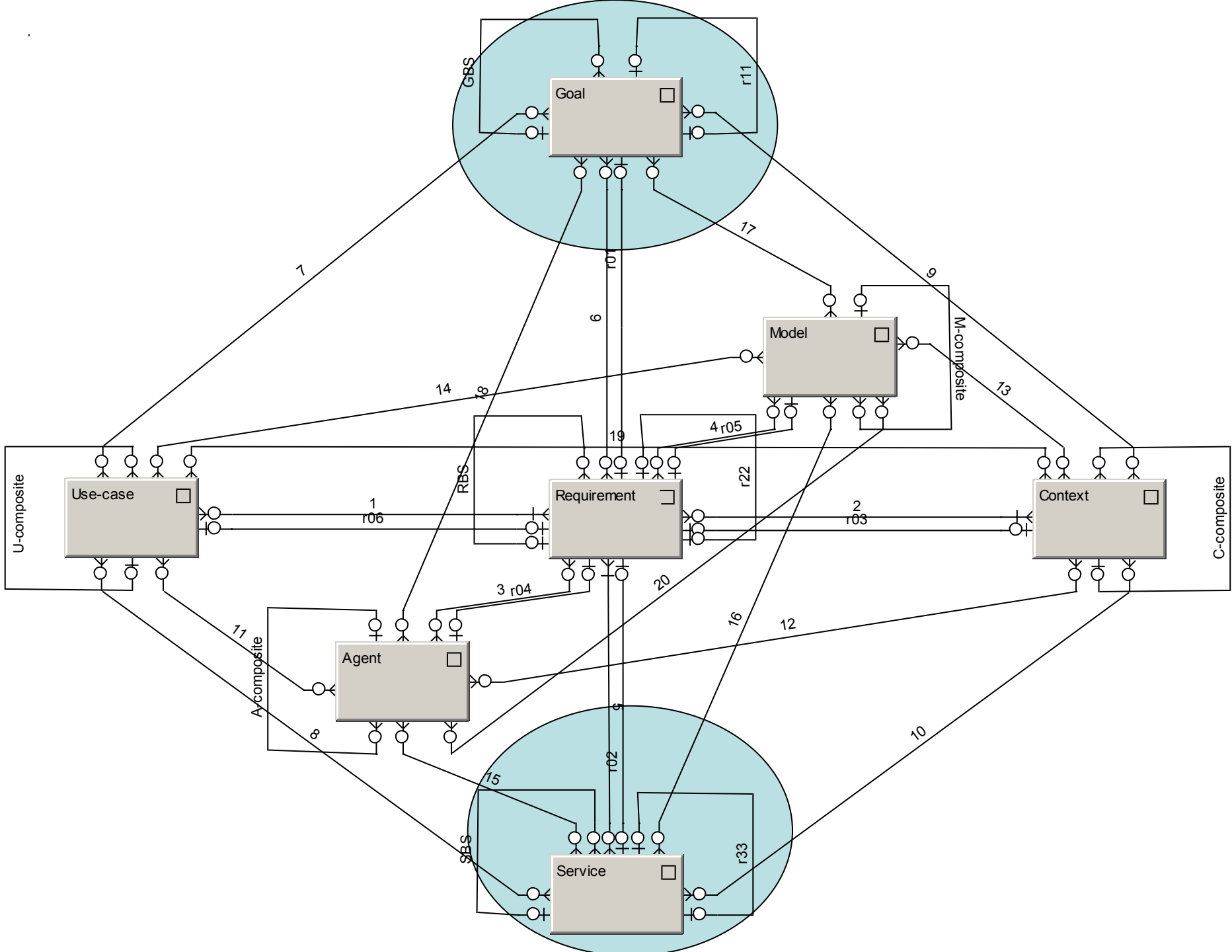
Summary definitions
(A. Cockburn)

- *Use Case*: a contract for the behavior of the SuD
- *Actor:* anyone or anything with behavior
- *Stakeholder:* someone or something with a vested interest in the behavior of the system under discussion (SuD)
- *Primary actor:* the stakeholder who or which initiates an interaction with the SuD to achieve a goal

GBS

Goal

r11

7

17

9

M-composite

13

6

Model

14

18

19

4 r05

r22

U-composite

RBS

Use-case

Requirement

Context

C-composite

1

r06

2

r03

3 r04

20

16

12

11

A-composite

Agent

8

5

15

r02

SBS

Service

r33

10

# Summary definitions (cont.)
(A. Cockburn)

- *Scope*: identifies the system that we are discussing.

- *Preconditions*: what must be true before the use case runs

- *Guarantees*: what must be true after the use case runs

Summary definitions (cont.)
(A. Cockburn)

- *Main success scenario*: a case in which nothing goes wrong

- *Extensions*: what can happen differently during that scenario

- Numbers in the extensions refer to the step numbers in the main success scenario at which each different situation is detected (e.g. steps 4a and 4b indicate two different conditions that can show up at step 4)

- When a use case references another use case, the referenced use case is <u>underlined</u>
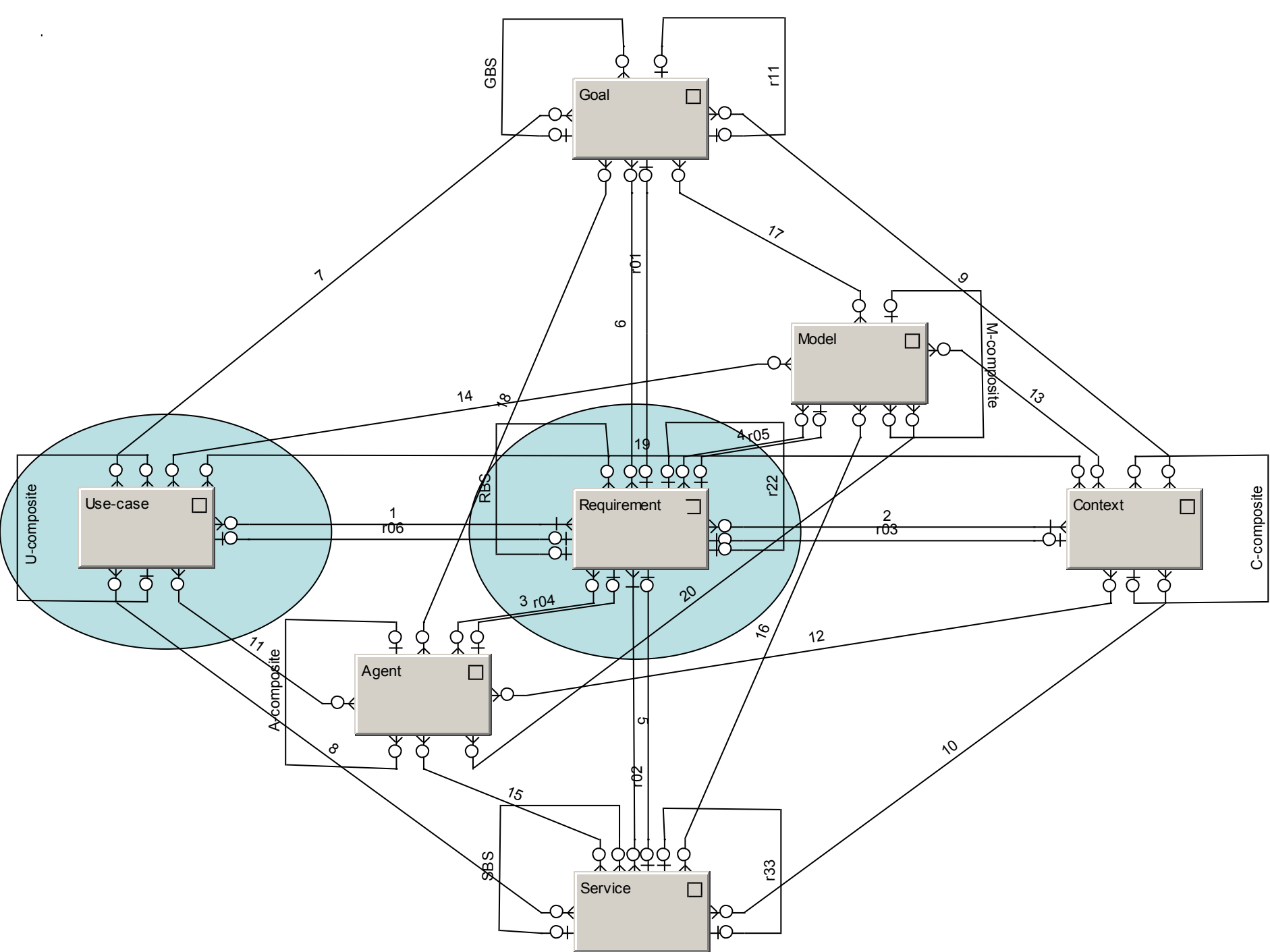
Use Case Template
(A. Cockburn)

<the name (the goal as a short active verb phrase)>

Context of Use: <a longer statement of the goal, if needed, its normal occurrence conditions>

Scope: <design scope, what system is being considered black-box under design>

Level: <one of: summary, user-goal, subfunction>

Primary Actor: <a role name, if needed, with description>

Stakeholders and Interests: <list of stakeholders and key interests in the use case>

Preconditions: <state of the world>

Minimal Guarantees: <how the interests are protected under all exits>

Use Case Template (cont.)

Success Guarantees: <the state of the world if goal succeeds>

Trigger: <what starts the use case, may be time event>

Main Success Scenario:

<put here the steps of the scenario from trigger to goal delivery and any cleanup after>

<step#><action description>

Extensions:

<put here the extensions, each referring to the step of the main scenario>

<step altered><condition>: <action or sub use case>
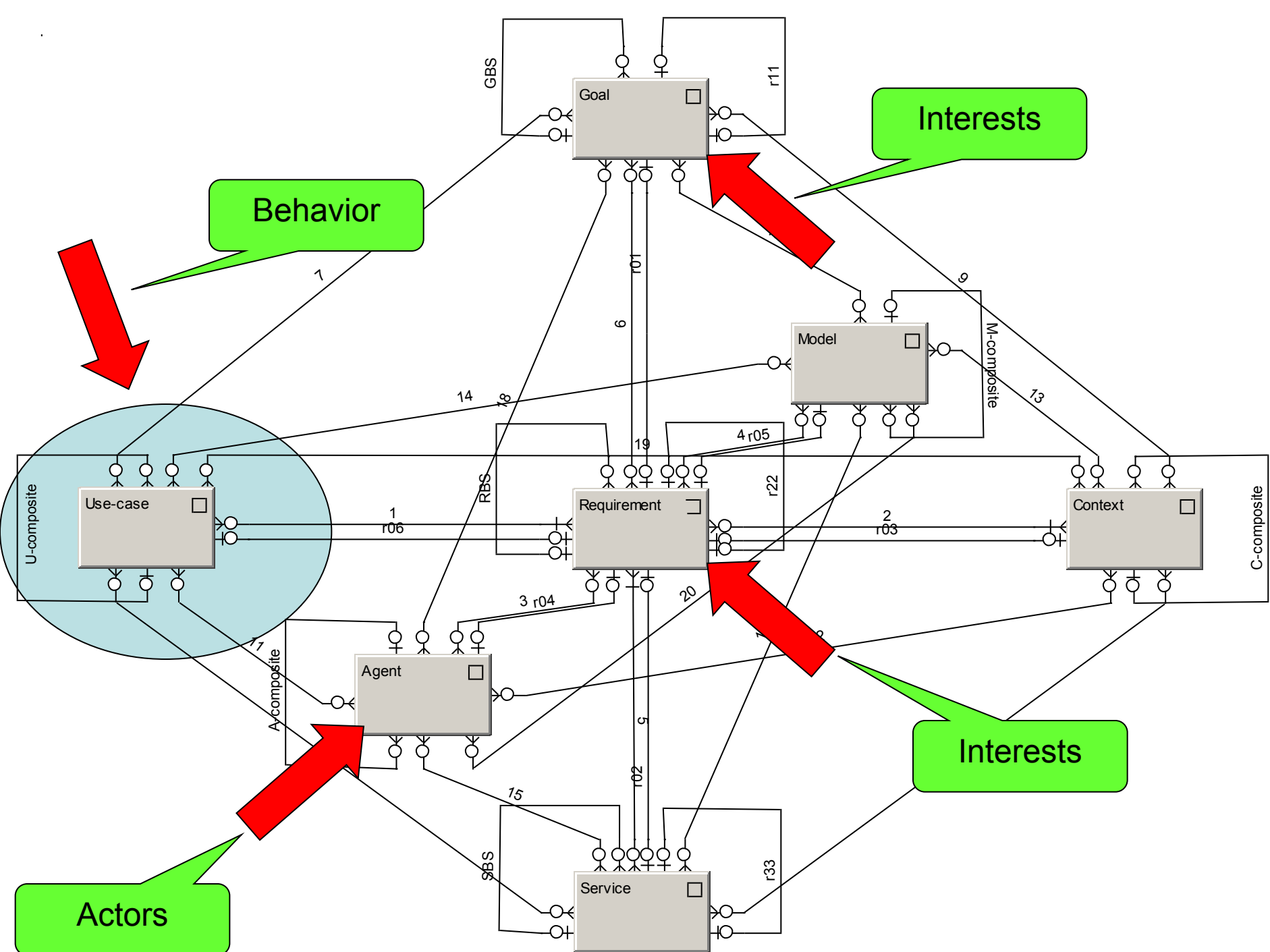
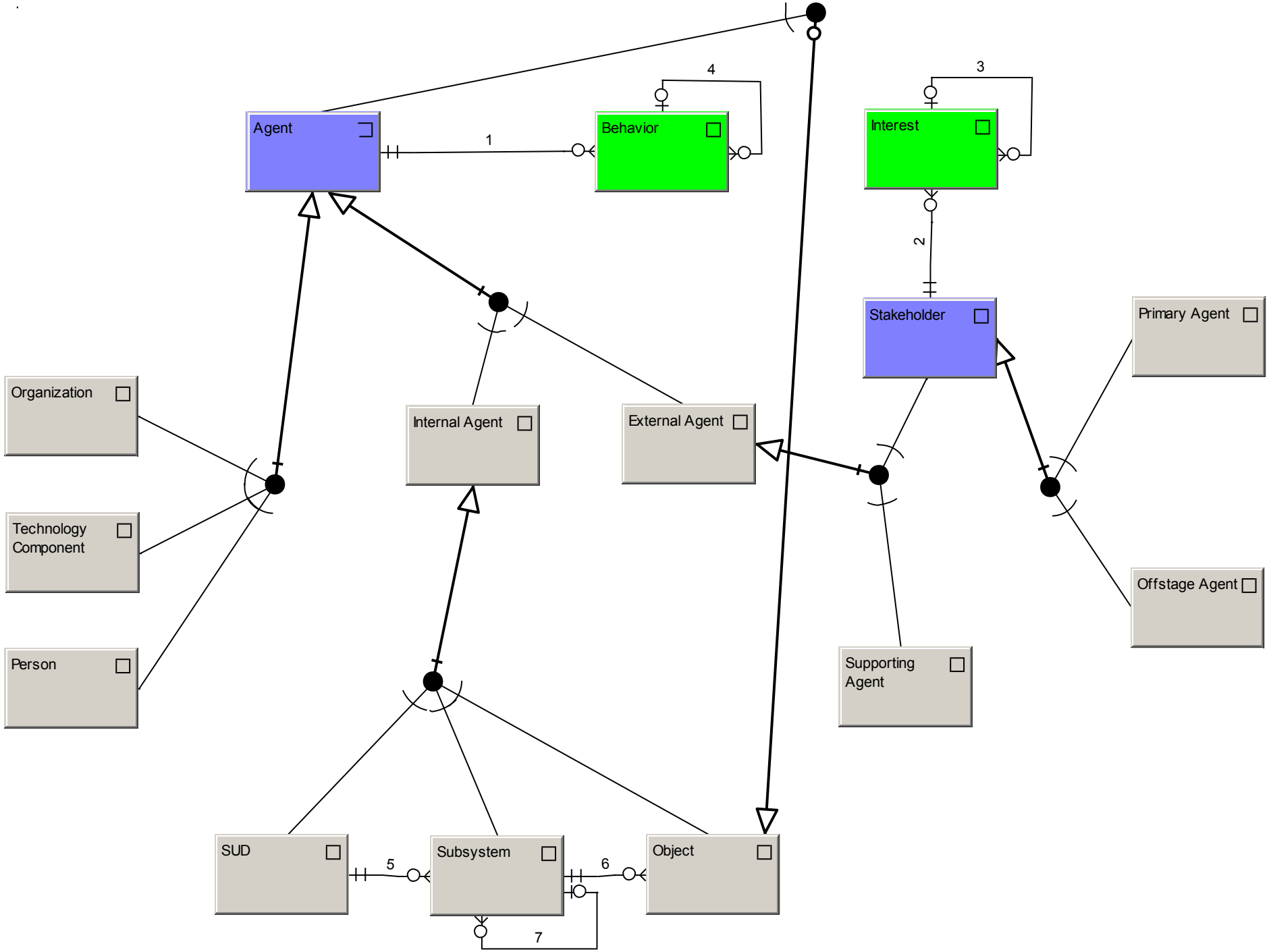**Related Information:**

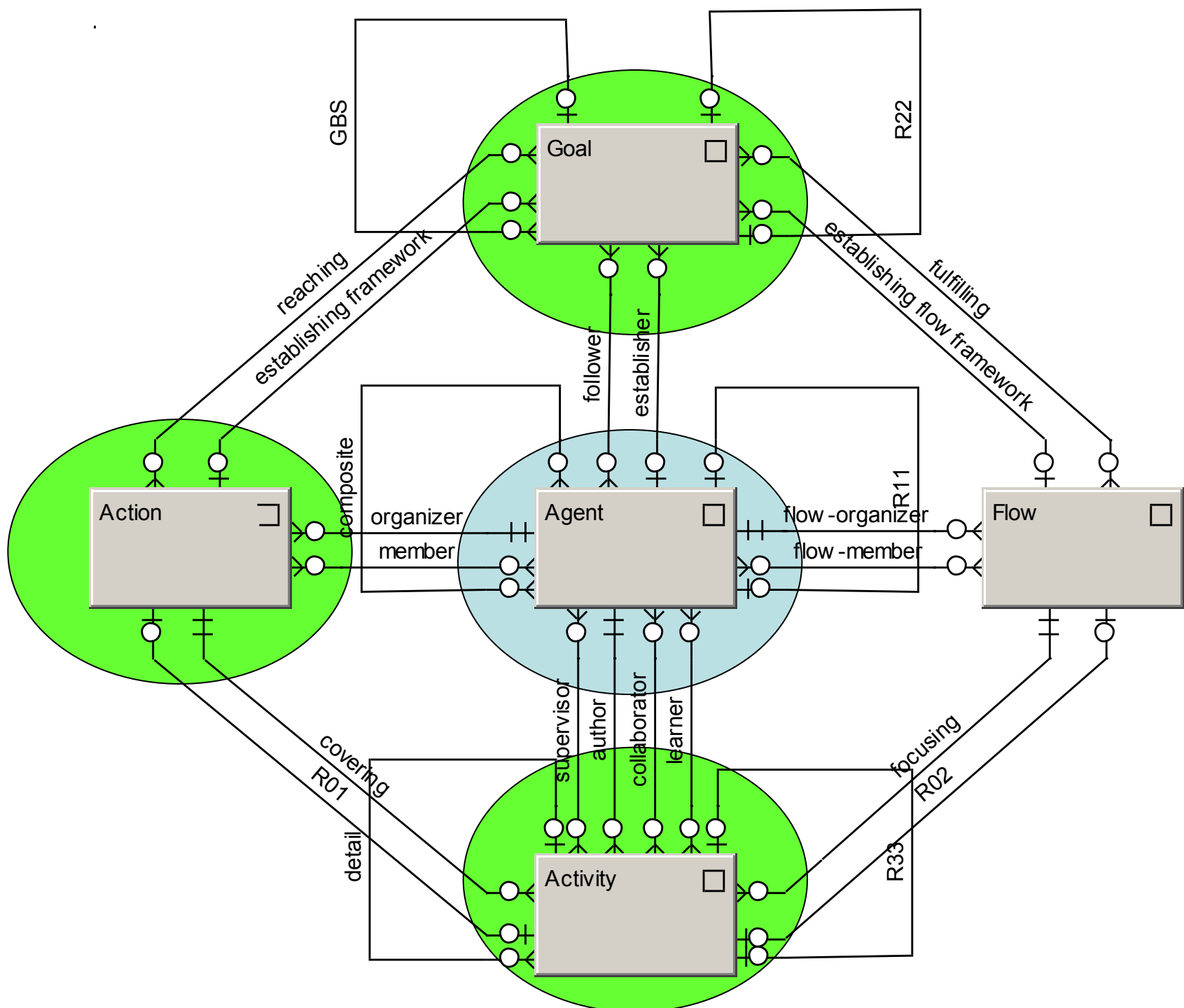# Readings

Alistair Cockburn:

Writing Effective Use Cases

… there you will find lot of examples and explanations …

Actors=Agents and Stakeholders

- A Stakeholder has interests
- An Actor or generally an Agent has behaviors
- The Primary Actor (the Agent from its perspective the Use Case is described) is also a Stakeholder

Behavior and Interactions as composite

- Goal-oriented Behavior consists of Responsibilities, Goals, and Actions
- The Private Actions we write are those that forward or protect the interests of Stakeholders.
- Interactions connect the Actors=Agents.
- Several Agents can participate in an Interaction.
- Interactions decompose into Use Cases, Scenarios, and Simple Messages.
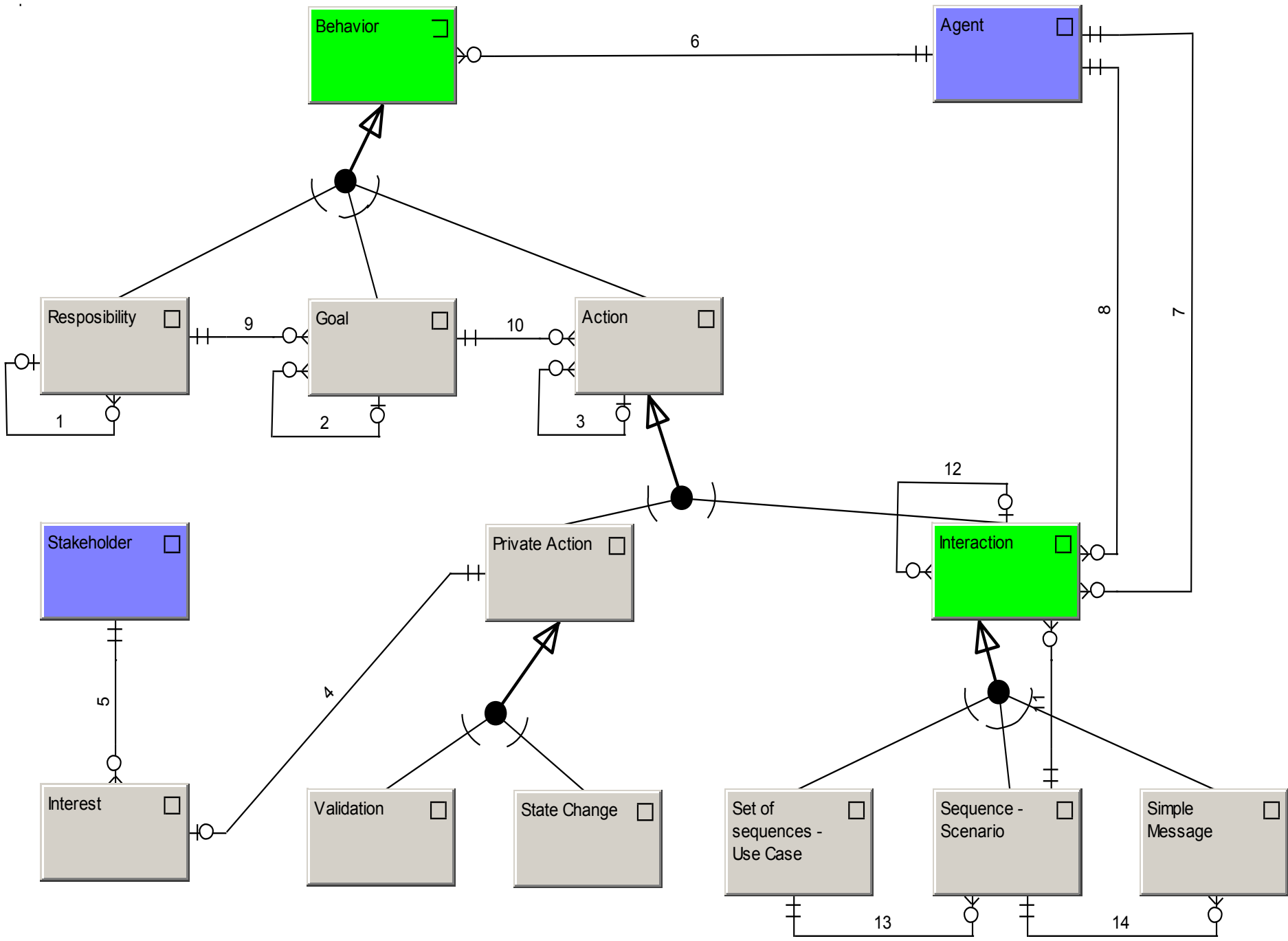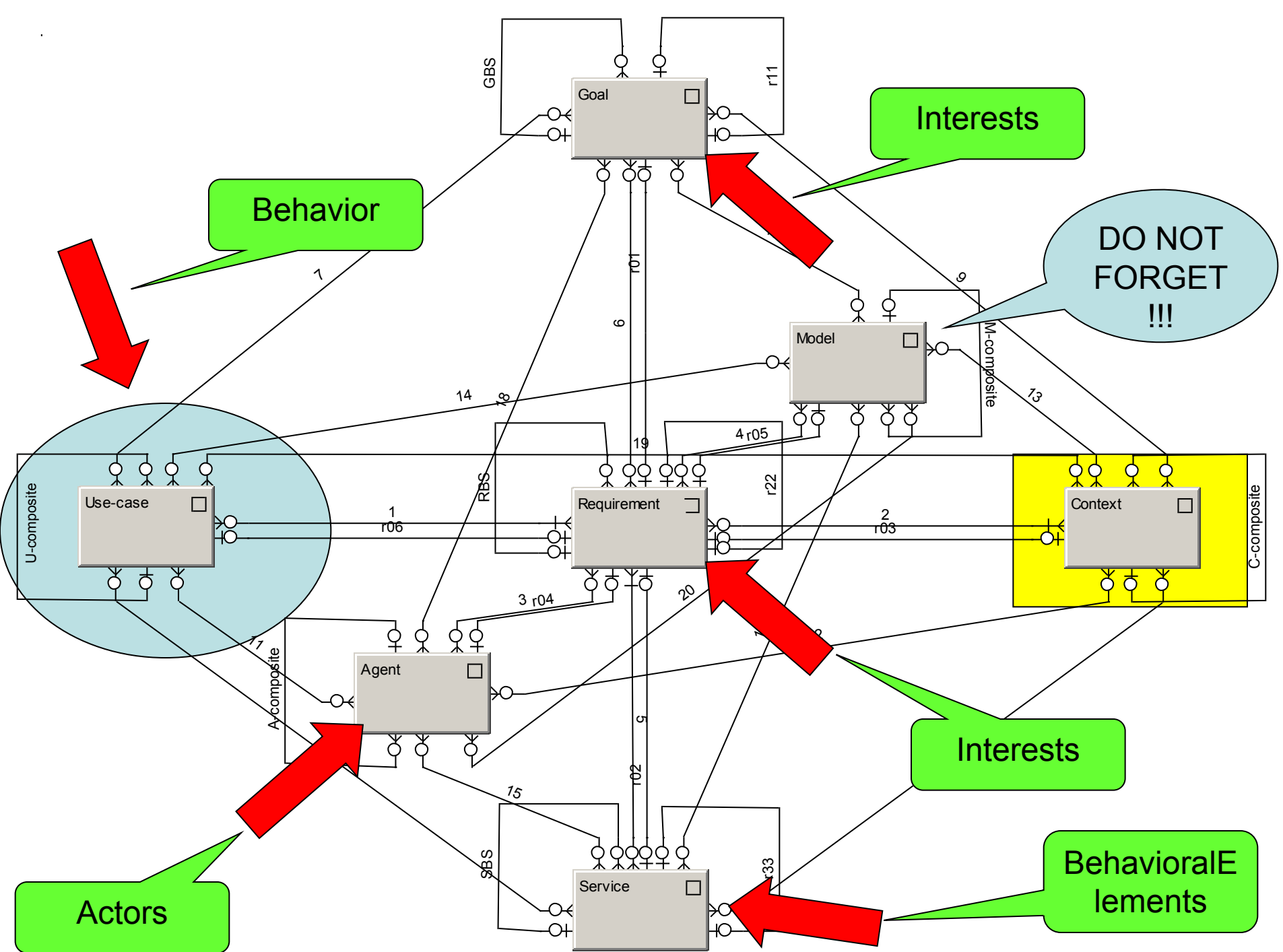
The Writing Process
(A. Cockburn)

1. Name the system scope and boundaries.
    Track changes to this initial context diagram with the in/out list.
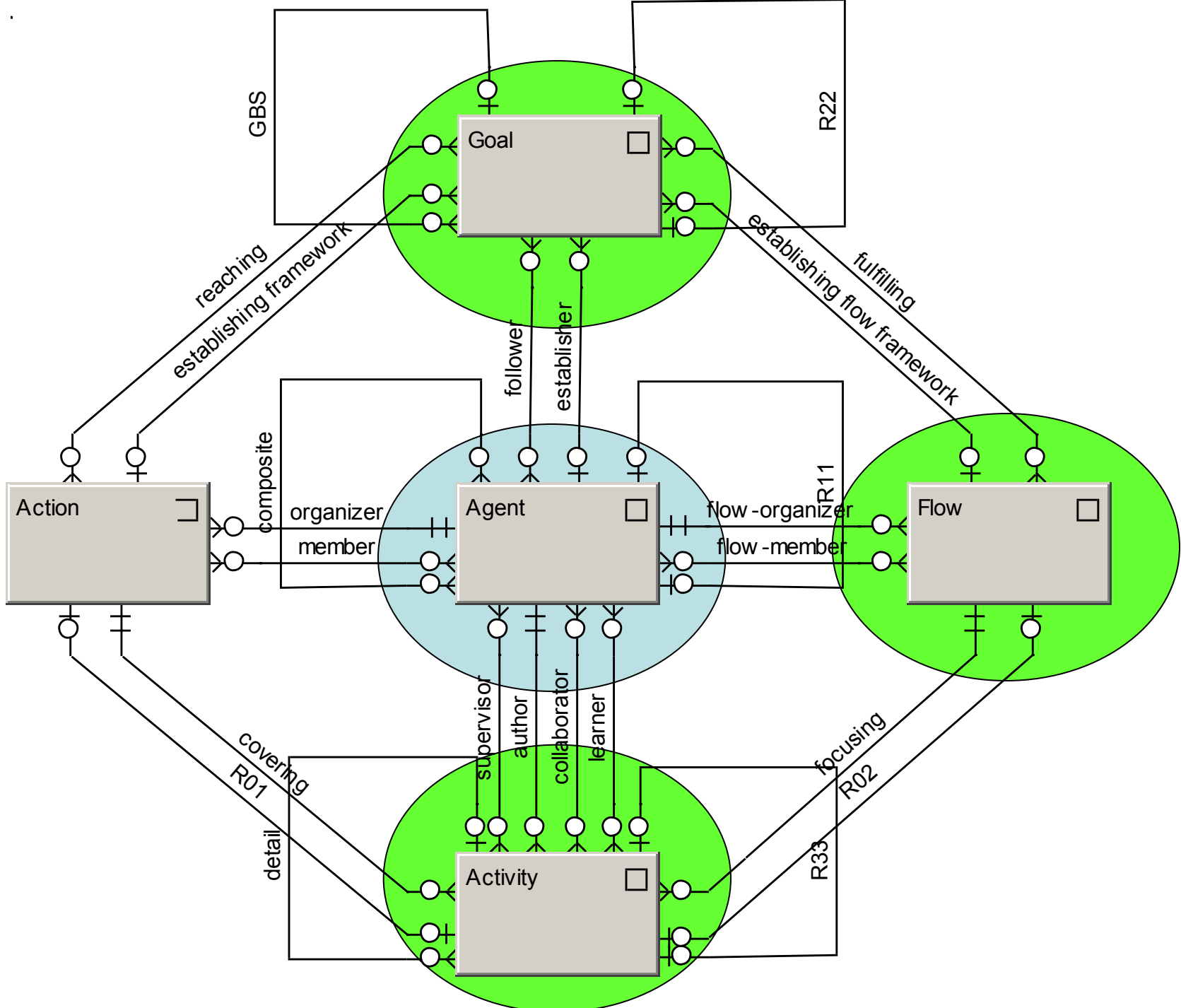
2. Brainstorm and list the primary actors.
    Find every human and non-human primary actor, over the life of the system.

3. Brainstorm and exhaustively list user goals for the system.
    The initial Actor-Goal List is now available.

4. Capture the outermost summary use cases to see who really cares.
    Check for an outermost use case for each primary actor.

The Writing Process (cont.)
(A. Cockburn)

5. Reconsider and revise the summary use cases. Add, subtract, or merge goals.
   Double-check for time-based triggers and other events at the system boundary.

6. Select one use case to expand.
   Consider writing a narrative to learn the material.

7. Capture stakeholders and interests, preconditions and guarantees.
   The system will ensure the preconditions and guarantee the interests.

8. Write the main success scenario.
   Use steps 3 – 9 to meet all interests and guarantees.

The Writing Process (cont.)
(A. Cockburn)

9. Brainstorm and exhaustively list the extension conditions.
Include all that the system can detect and must handle.

10. Write the extension-handling steps.
Each will end back in the MSS, at a separate success exit, or in failure.

11. Extract complex flows to sub use cases; merge trivial sub use cases.
Extracting a sub use case is easy, but it adds cost to the project.

12. Readjust the set: add, subtract, merge, as needed.
Check for readability, completeness, and meeting stakeholders' interests.

Reminders
(A. Cockburn)

- Write something readable !
- Work breadth-first: from lower precision to higher precision
  - Precision level 1: Primary actor's name and goal
  - Precision level 2: The use case brief, or the main success scenario
  - Precision level 3: The extension conditions
  - Precision level 4: The extension handling steps

Reminders (cont.)
(A. Cockburn)

- For each step:
  - Show a goal succeeding
  - Capture the actor's intention (not the user interface details)
  - Have an actor pass information, validate a condition, or update state
  - Write between-step commentary to indicate step sequencing (or lack of)
  - Ask "why" to find a next-higher level goal

Reminders (cont.)
(Z. Stanicek)

- For data description:
  - Precision level 1: Names of entity sorts involved
  - Precision level 2: Definitions of entity sorts
  - Precision level 3: The conceptual model
  - Precision level 4: Descriptive attributes
- Think in systematic top-down manner
  - E.g. using Diam4 – Diam1

S

R

O

D

S diagram

Connection

Category — Object — Rule

Operation

01, R1, P, 03, R2, R4, 02, R3, 04

R diagram

Category

CI-connection — Item — Context

Manifestation

ci2category, base, r1, r3, ci2item, r2, r4, manif2ci, manif2context

O diagram

Goal

Action — Agent — Flow

Activity

GBS, R22, reaching, establishing framework, establishing flow framework, fulfiling, composite, organizer, member, follower, establisher, flow-organizer, flow-member, R11, supervisor, author, collaborator, learner, covering, R01, detail, focusing, R02, R33

D diagram

Goal

Use-case — Requirement — Context

Model

Agent

Service

GBS, r11, 7, 17, 9, M-composite, 6, r01, 14, r8, 19, 4 r05, 13, RBS, 1, r06, r22, 2, r03, C-composite, 3 r04, 20, A-composite, 11, r21, 16, 12, 8, 15, SBS, r02, r33, 10