SE463
Software Requirements Specification & Analysis

Lightweight Modelling

Readings:

Robertson, S. and Robertson, J., *Mastering the Requirements Process, 3ed.*, Chapter 6 “Scenarios”

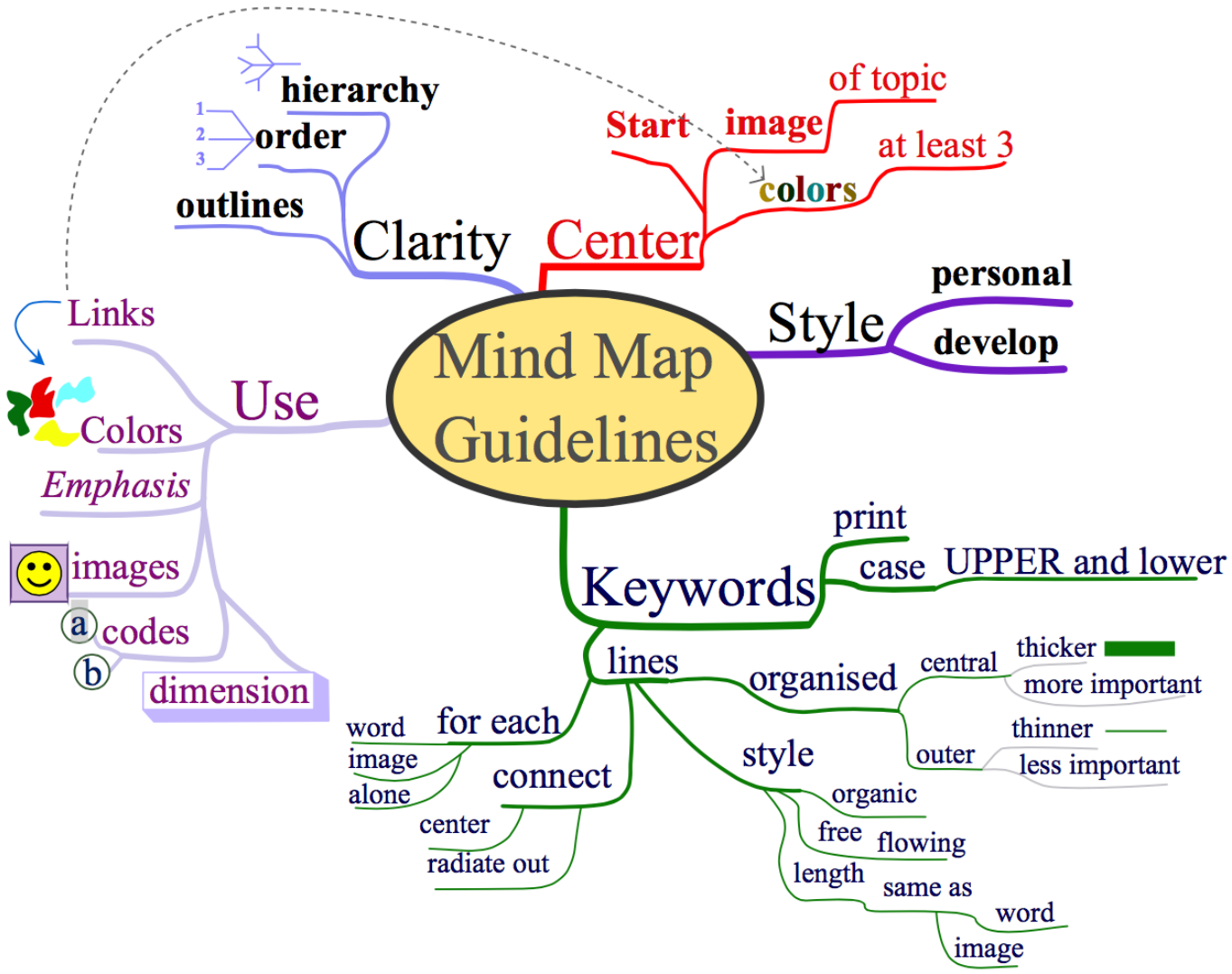
Lightweight Modelling

- Mind maps
- Scenarios
- Activity diagrams
- Process models

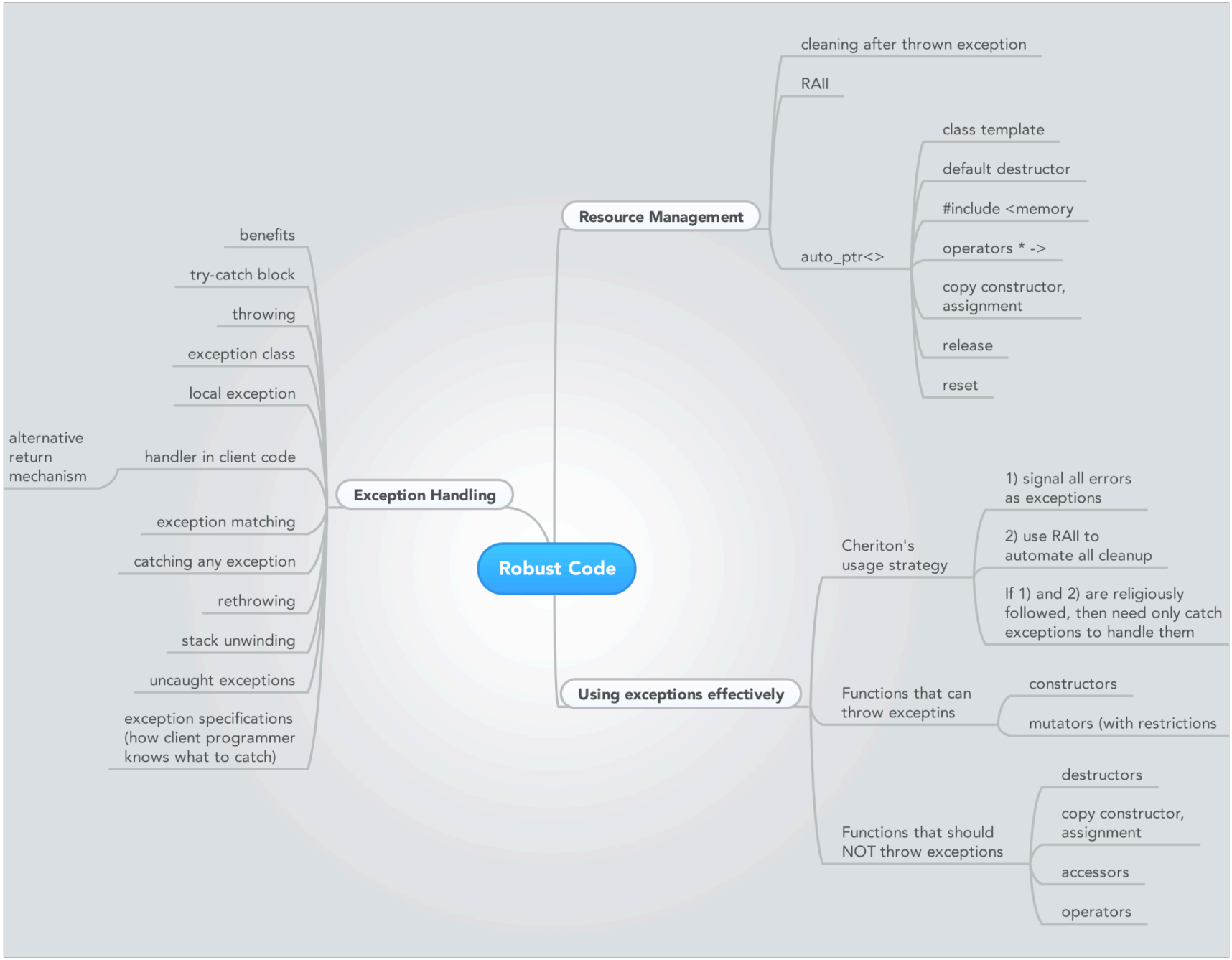
Act of modelling encourages requirements analyst and stakeholders to take an active role in constructing descriptions

- of current work processes
- of functional requirements

Mind Maps



https://en.wikipedia.org/wiki/Mind_map



Scenarios

A scenario is one full execution path through a use case, listing only observable actions.

Banking Scenario

1. User requests to withdraw funds, specifies amount
2. Bank authenticates user
3. Bank checks that the account has sufficient funds
4. Bank dispenses cash and receipt
5. Bank records the transaction
6. User takes cash and receipt and leaves

More Complex Actions

- If (conditional statement)
- For (iteration expression)
- While (conditional iteration)
- Go To UC*n*

Example:

3. While the Bank checks that account has sufficient funds

3.1 Show advertisements to the customer

These constructs are not needed very often and may be a sign that the scenario is becoming too detailed or too much like pseudo-code.

Variations

- Alternative: a sub use case that achieves the main goal of a use case through a different intended sequence of actions

  Example:

  1. User requests to withdraw funds, specifies amount

  A1.1 User cancels transaction

  6. User takes cash and receipt and leaves

  A6.1 User takes cash and receipt, initiates new transaction

- Exception: a sub use case that captures an unwanted but inevitable deviation

  Example:

  3. Bank checks that account has sufficient funds

  E3.1 Bank cancels withdrawal due to insufficient funds

Negative Scenarios

- Negative scenario: a scenario that is to be prohibited by the system

Example:

Banking Scenario

1. User requests to withdraw funds, specifies amount
2. Bank authenticates the user
3. Bank checks that the account has *insufficient* funds
4. Bank dispenses cash and receipt

- Misuse case: a scenario that captures undesirable inputs from the user or environment

Example:

2. Bank authenticates user

M2.1 User is an impostor

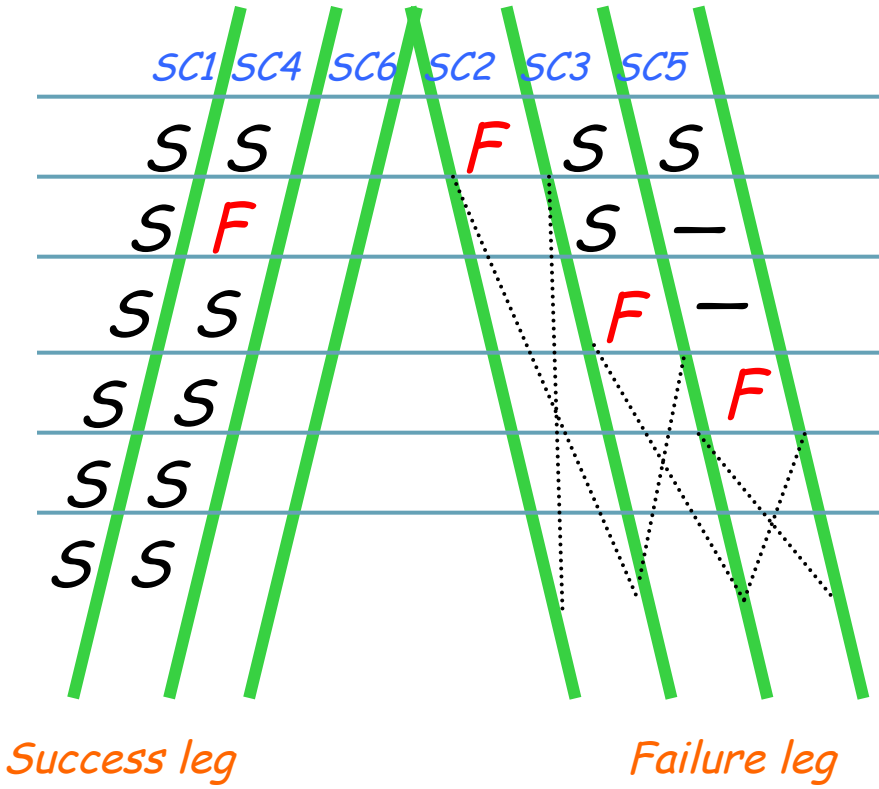
Example: Meeting Scheduler

Meeting Scheduler Scenario

1. User requests meeting; specifies participant (important, active, ordinary), agenda, date range
2. Invitations sent to participants, asking about their availability
3. The initiator is asked about room and AV requirements
4. Participants reply with their availability
5. Initiator replies with room and AV requirements
6. AV equipment is reserved
7. When all participants have replied, the system finds a conflict-free date, time and room for meeting
    7.1 Room is reserved
8. Confirmation message is sent to all participants and initiator

Use cases vs. Scenarios

A use case is a collection of success & failure scenarios initiated by an external actor, to achieve a particular goal



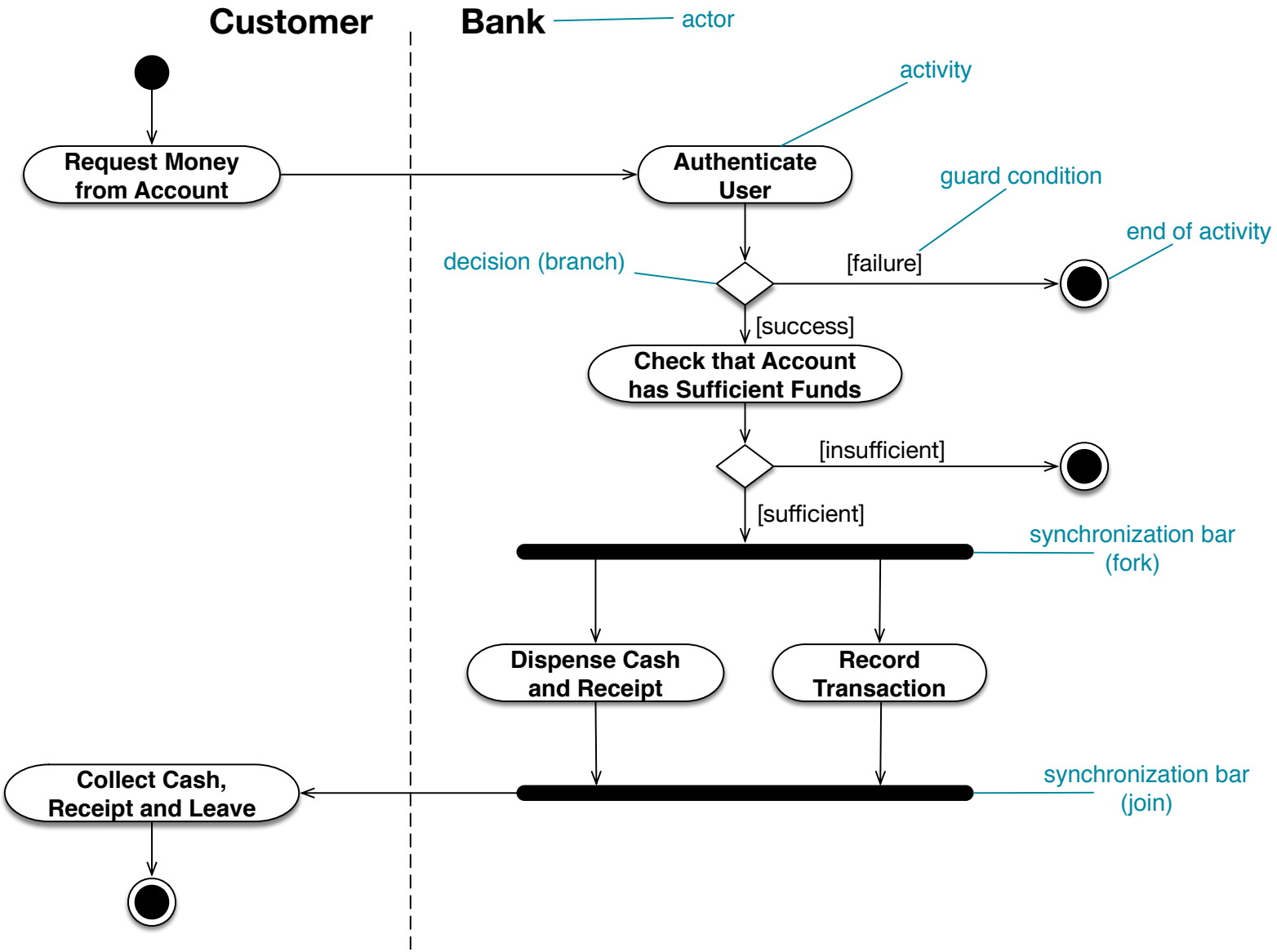
1. User authenticates
2. User requests withdraw
3. Bank checks account balance
4. Bank dispenses cash and receipt
5. Bank records the transaction
6. User takes cash and receipt

S	Success
F	Failure
—	Don't care

Template

- **Business Event Name:**
- **Business Use Case Name and Number:**
- **Trigger:**
- **Preconditions:** Sometimes certain conditions must exist before the use case is valid.
- **Interested Stakeholders:**
- **Active Stakeholders:**
- **Normal Case Steps:**
  - Step 1 . . .
  - Step 2 . . .
  - Step 3 . . .
- **Alternatives**:
- **Exceptions**:
- **Outcome**:

Activity Diagram



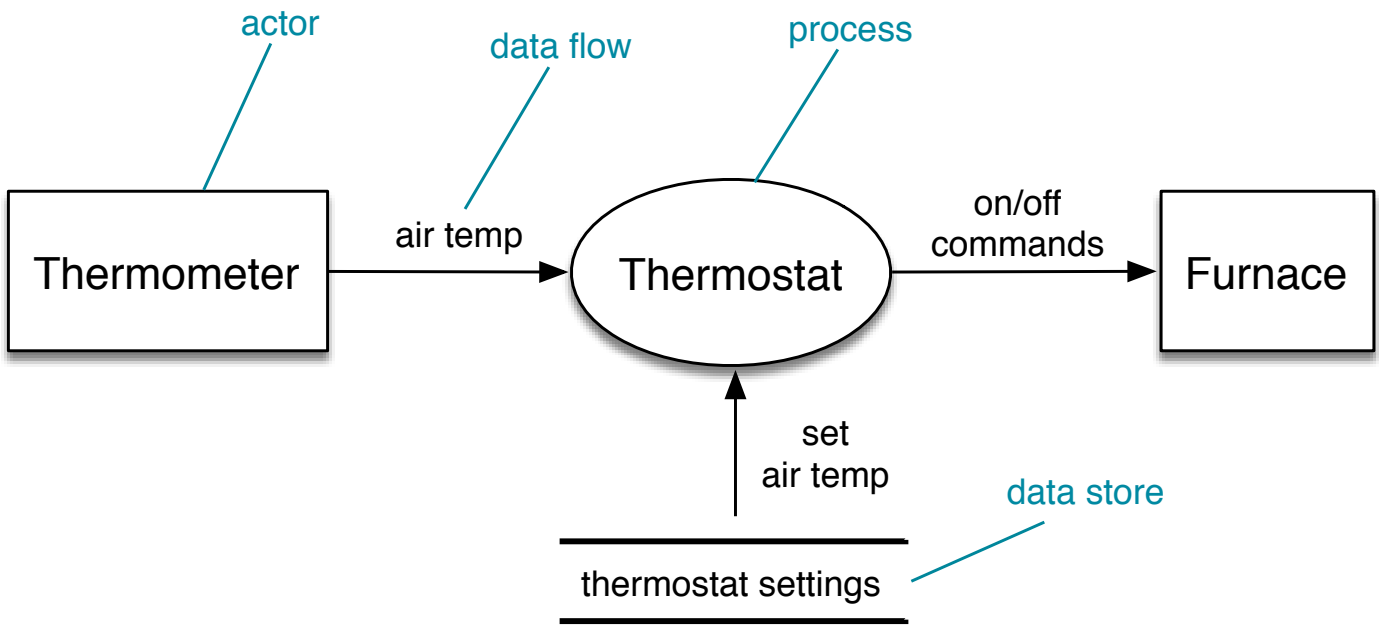
Example: Job Mine

Create an activity diagram to describe the work of a co-op student applying for a job

- Starting with applying for a job
- Ending with a job offer (or not)

Process Model

A process model is a functional decomposition of the work, and the data dependencies between functions.



Summary

Act of modelling encourages requirements analyst and stakeholders to take an active role in constructing descriptions

Want requirements modelling notations to have simple syntax and semantics, for nontechnical stakeholders to be able to read them, understand them, and contribute to their construction

- Mind maps
- Scenarios
- Activity diagrams
- Process models