Goal Modeling

We Will Cover

- What is a goal?
- Where do goals come from?
- What is a goal model?
- When to use goal models?
  - How do goal models relate to UML models?
- Why use goal models?
- Capturing the goal model

What is a goal?

- A stakeholder objective for the system
  - The system includes the software and its environment
- Who are stakeholders?
  - Anyone who has an interest in the system
  - Customers, end users, system developers, system maintainers...
- What is a goal model?
  - A hierarchy of goals
  - Relates the high-level goals to low-level system requirements

Goal or Not a Goal?

- Goal Examples:
  - – Credit card information is kept private
  - – Credit card information is accurate
  - – Safe transportation
  - – Highly reliability
- Non-goal Examples:
  - – The system will be implemented in C++
  - – The paint colors for the cars will be yellow, orange, and red

Goal Exercise

- Order the list of goals from high-level concern to low-level concern
  - User receives a request for a timetable from a system
  - Collect timetables by system
  - Schedule meeting
  - System collect timetables from user
  - Collect timetables

Goal Exercise

- Order the list of goals from high-level concern to low-level concern
  - – Schedule meeting
  - – Collect timetables
  - – Collect timetables by system
  - – System collect timetables from user
  - – User receives a request for a timetable

Types of Goals

- Functional (Hard)
  - Describe functions the system will perform
  - Well defined criteria for satisfaction
  - E.g., System collects timetables from user
- Non-functional (Soft or fuzzy)
  - Describe desired system qualities
  - Hard to define; satisficed rather than satisfied
  - Reliability
    - E.g., System should be reliable
  - Quality
    - E.g., System should be high quality.

Where do goals come from?

- Conveyed by stakeholders
- Disclosed in requirements documents
- Analysis of similar or current system
- Elaborating other goals

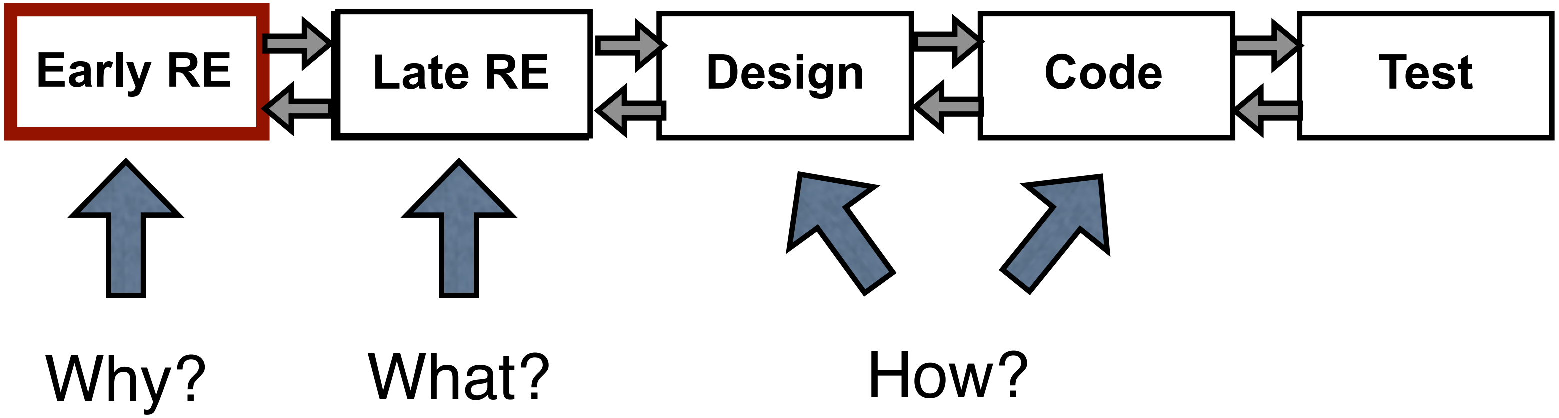
Goal Exercise

- Identify goals in the following paragraph

An Adaptive Regional Forecasting Ecological Observatory (ARFEO) is proposed as means to implement an operational, run-time configurable, and adaptable ecological observatory to be used for regional ecological forecasting. ARFEO has three main dimensions. First, using a holistic perspective of environment, we will use sensors that are analogous to human and organism senses to monitor the environment and its changes. As such, ARFEO can be configured to observe and answer ecological questions specific to one class of sensory input or across multiple sensory inputs. Second, ARFEO will use a cyberinfrastructure comprising smart, heterogeneous sensor networks, small-scale and GRID-scale distributed computing [6]. ARFEO's architecture will be a distributed design which will enable users to perform small and complex computations and transparent integration of data and analysis. ARFEO will enable a user to adapt monitoring capabilities at run-time, thus allowing a user to customize the configuration to specific needs and questions. Finally, ARFEO will emphasize the reuse and synergistic integration of existing analysis and visualization techniques to include in the computational toolkit for processing the sensor and ancillary data, as well as metadata. A key part of ARFEO will be the development of processes to make use of the toolkit element to support the analysis and visualization capabilities.

When to use goal models

- Early requirements engineering
  - Focus on identifying problems
  - Exploring system solutions and alternatives
  - Done before UML modeling



Why use goal models?

- Give rationale for requirements
- Identify stable information
- Guide requirement elaboration

The Goal Model

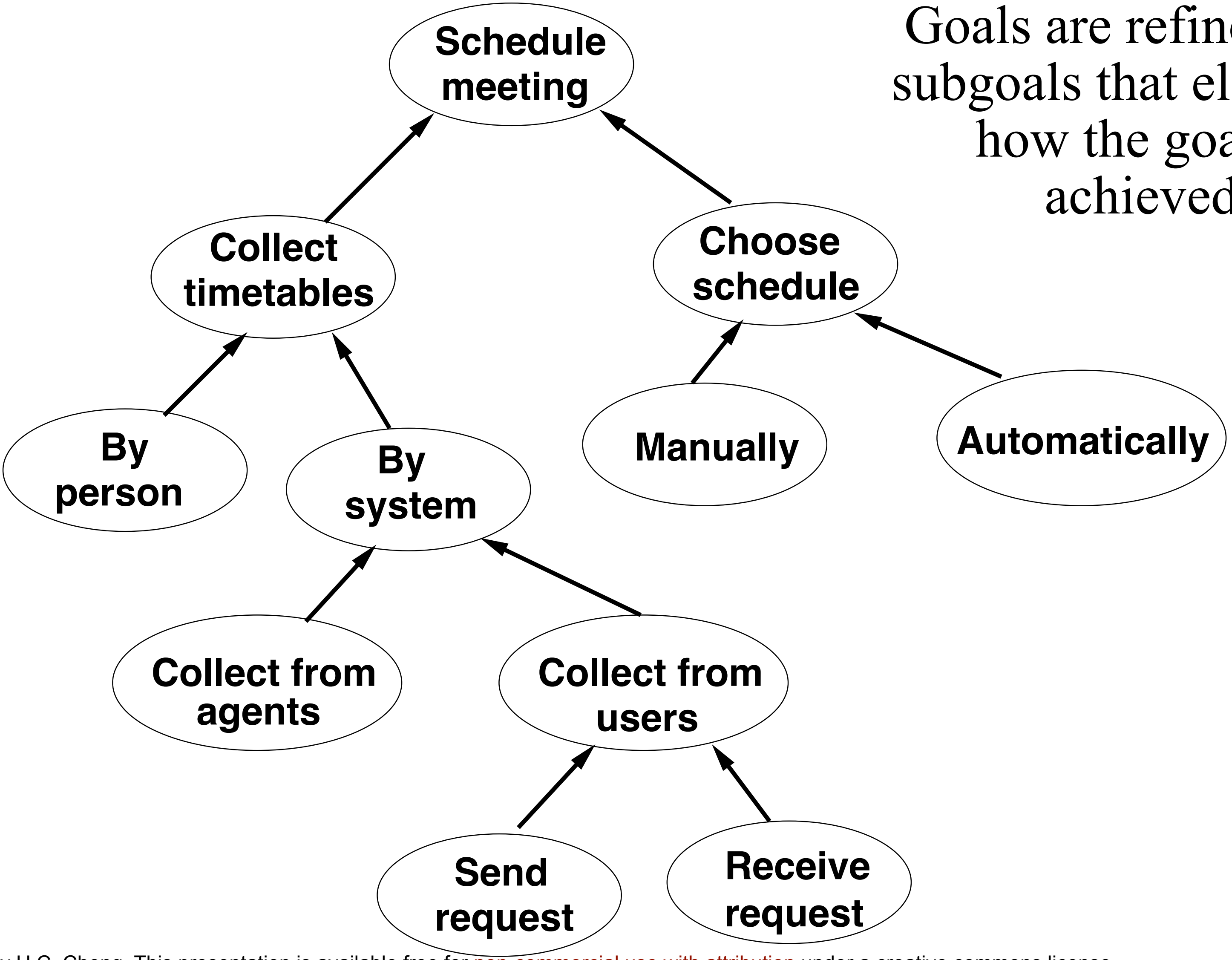
Running Example

- Meeting Scheduler
  - – Assists the initiator in scheduling a meeting
  - – Meeting should be convenient for participants
    - Participants should be available

- Modeled using the i* goal notation

Example adapted from the RE06 keynote given by John Mylopoulos

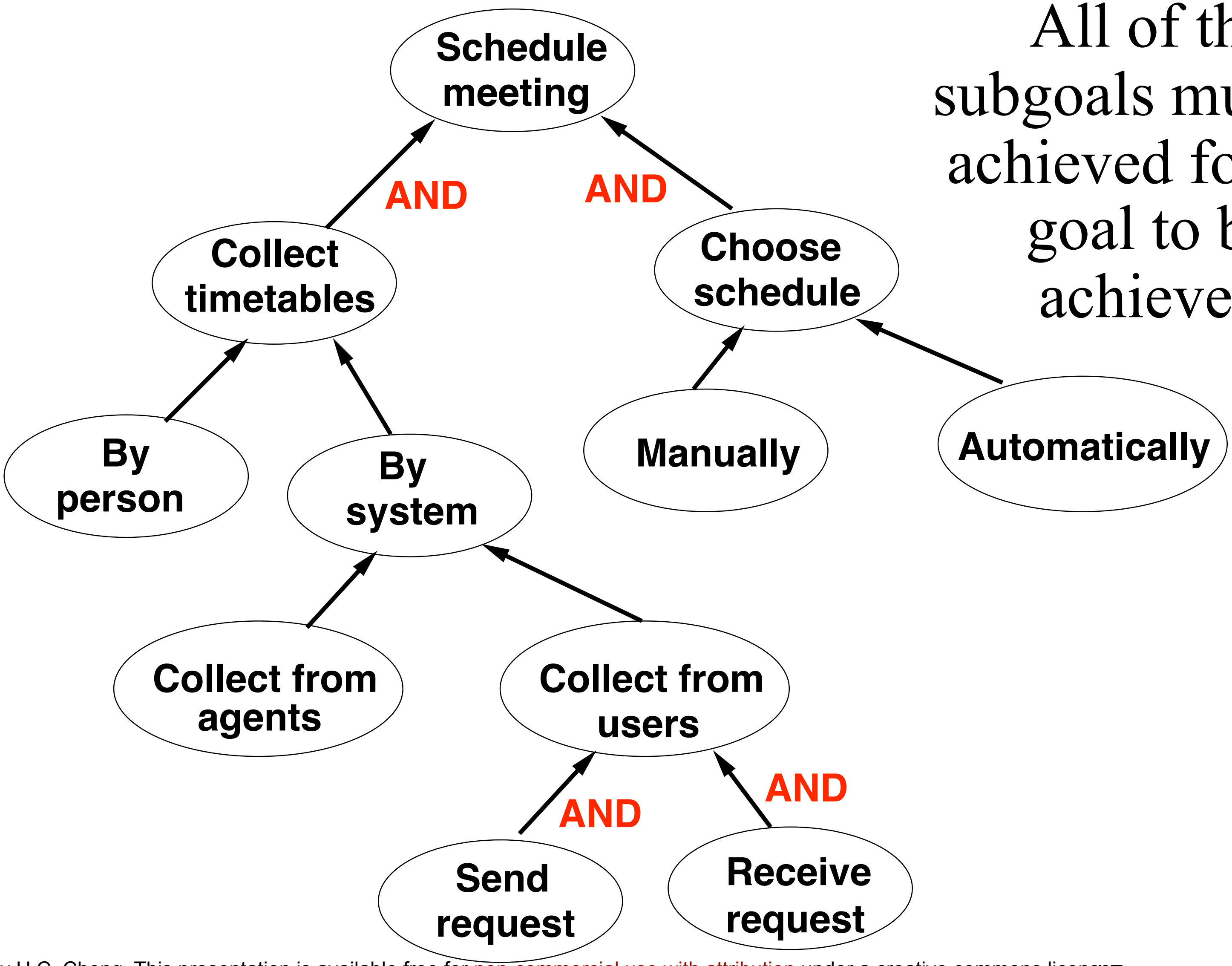
Goal Model



Goals are refined into subgoals that elaborate how the goal is achieved

Example adapted from the RE06 keynote given by John Mylopoulos

AND Refinement

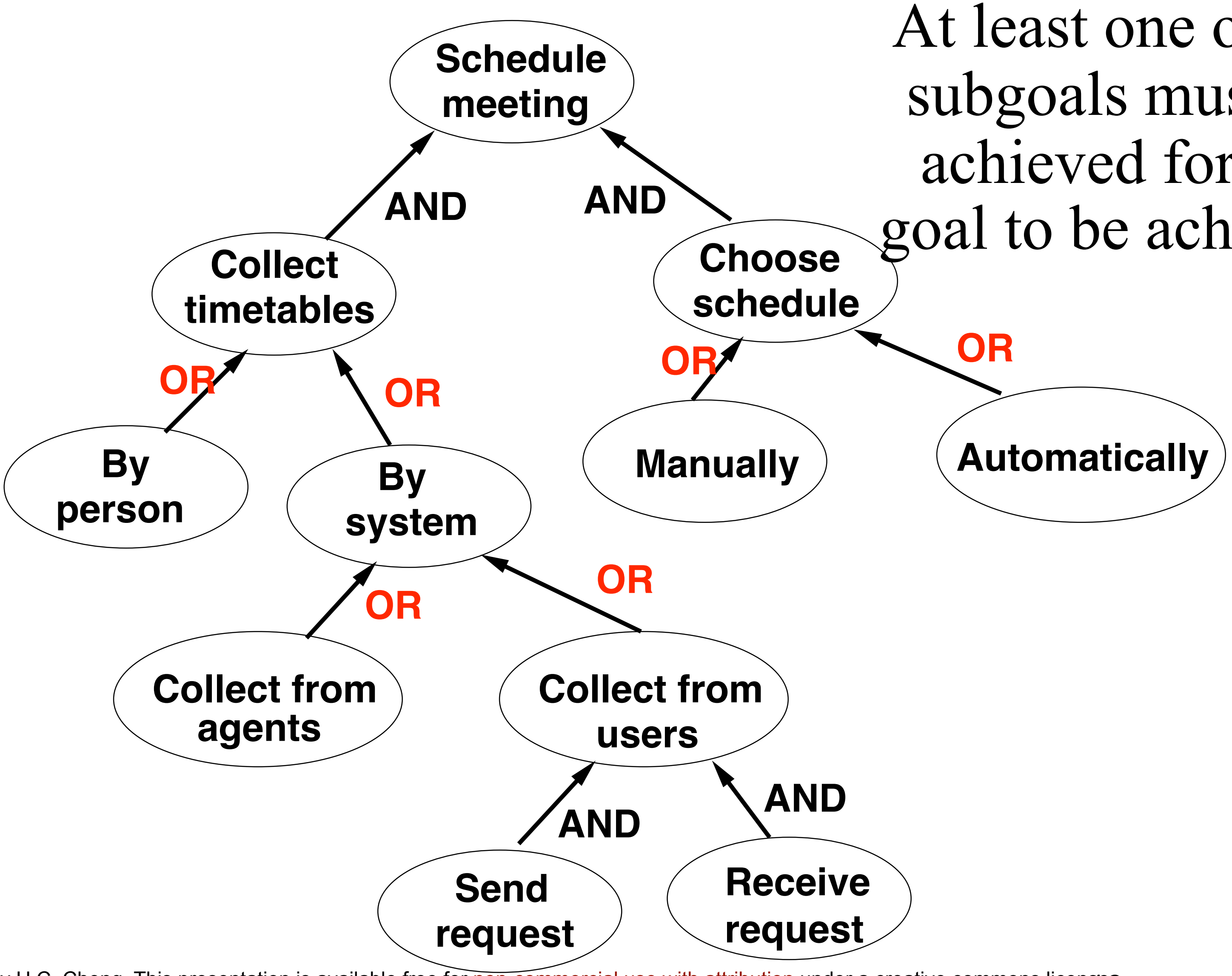


All of the subgoals must be achieved for the goal to be achieved

Example adapted from the RE06 keynote given by John Mylopoulos

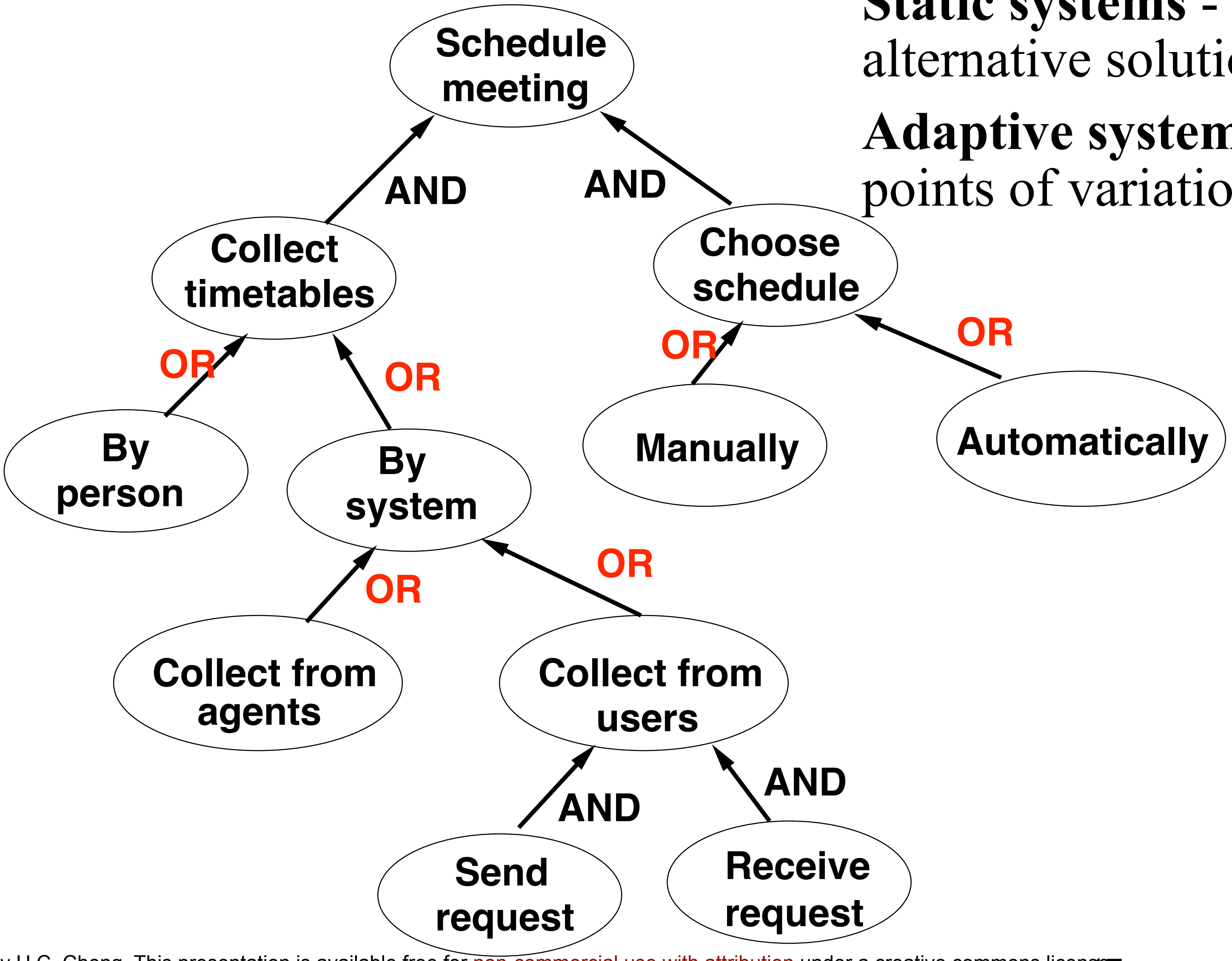
OR Refinement

At least one of the subgoals must be achieved for the goal to be achieved



Example adapted from the RE06 keynote given by John Mylopoulos

Interpretations of OR Refinements

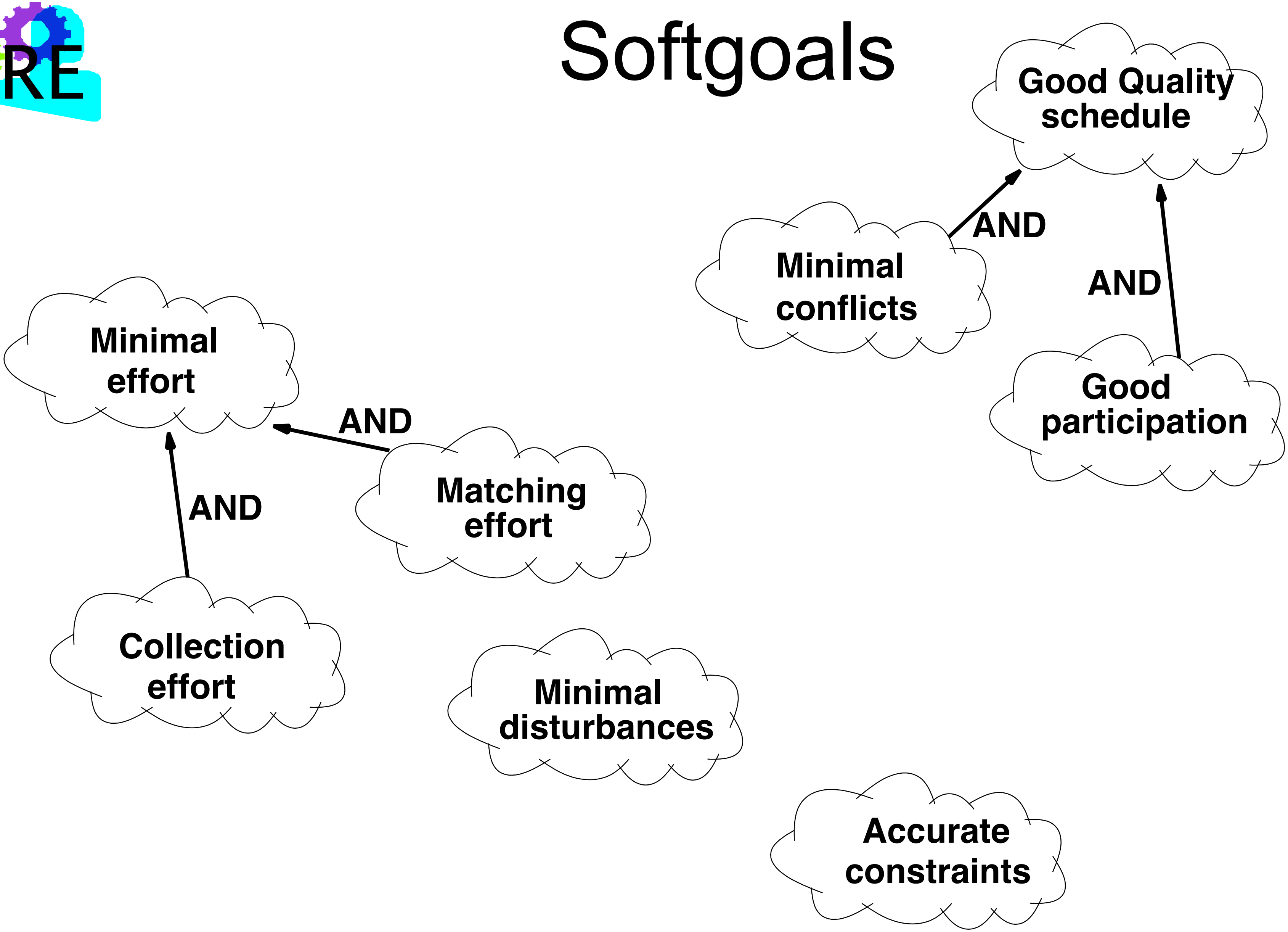


Static systems - alternative solutions

Adaptive systems – points of variation

Example adapted from the RE06 keynote given by John Mylopoulos

Softgoals



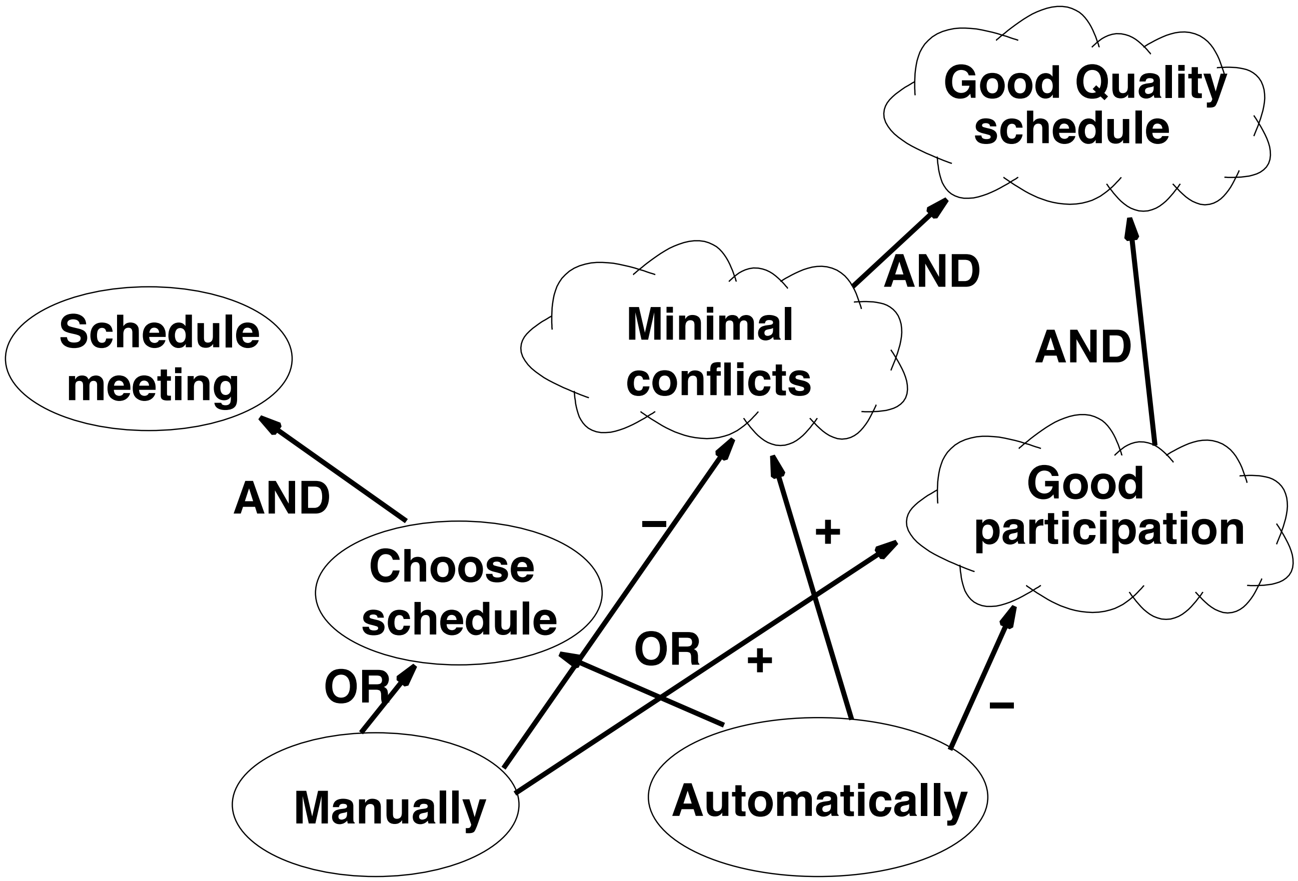
Example adapted from the RE06 keynote given by John Mylopoulos

Modeling Softgoals

- Used to evaluate alternatives
- Helps (+)
- Makes (++)
- Hurts (-)
- Breaks (--)

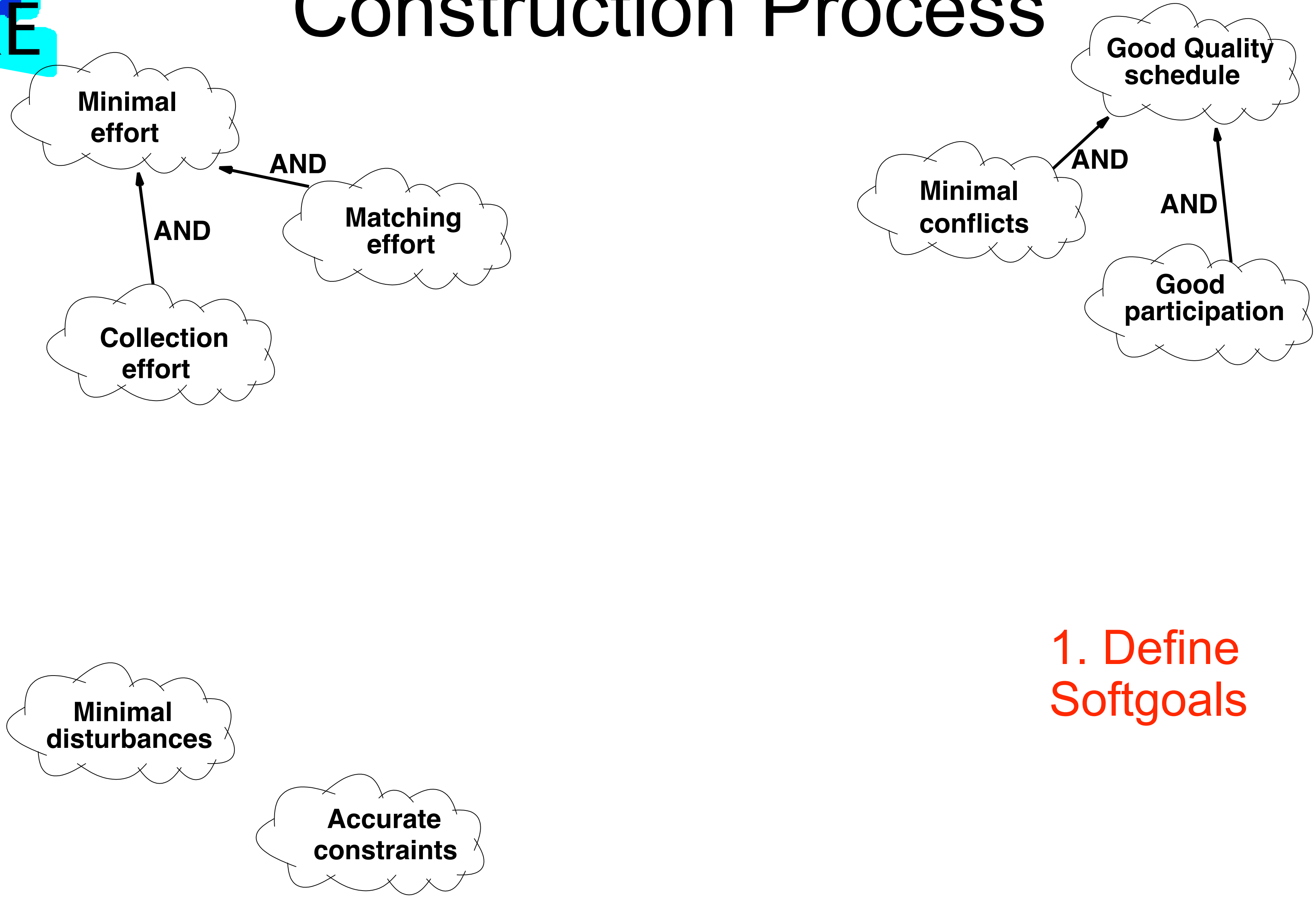
Example adapted from the RE06 keynote given by John Mylopoulos

Contributions to Softgoals



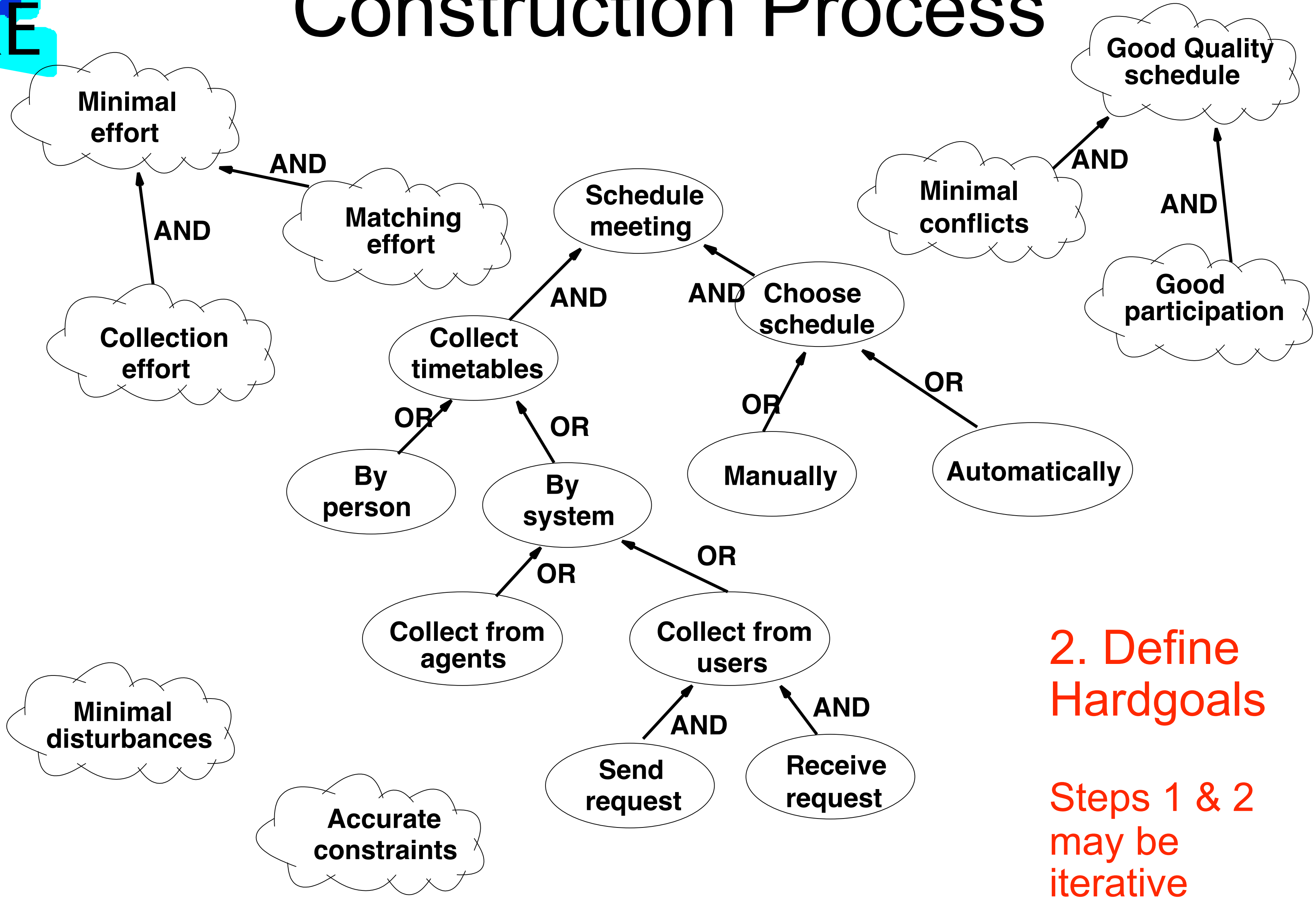
Example adapted from the RE06 keynote given by John Mylopoulos

Construction Process



1. Define Softgoals

Construction Process

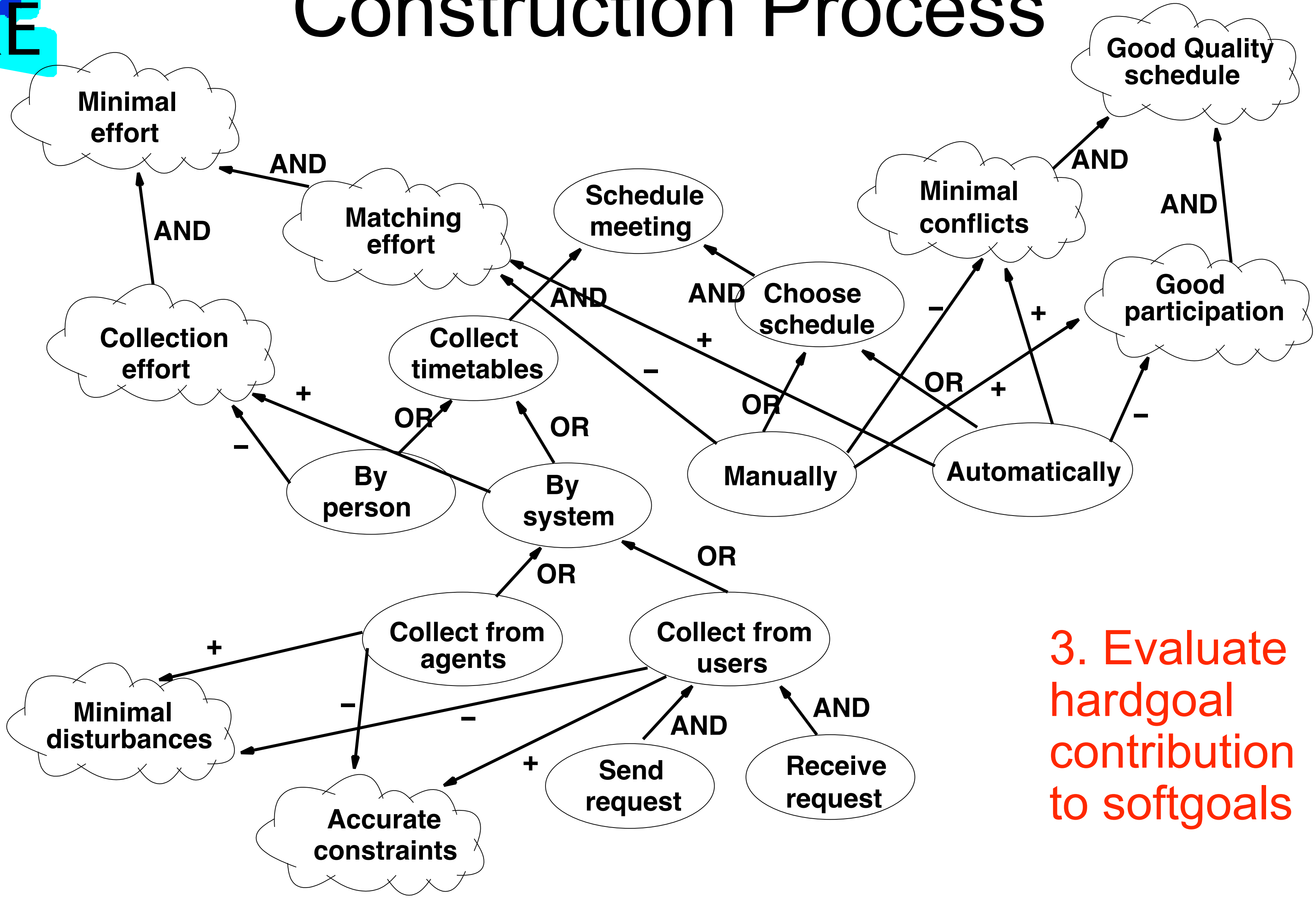


2. Define Hardgoals

Steps 1 & 2 may be iterative

Example adapted from the RE06 keynote given by John Mylopoulos

Construction Process



3. Evaluate hardgoal contribution to softgoals

Example adapted from the RE06 keynote given by John Mylopoulos

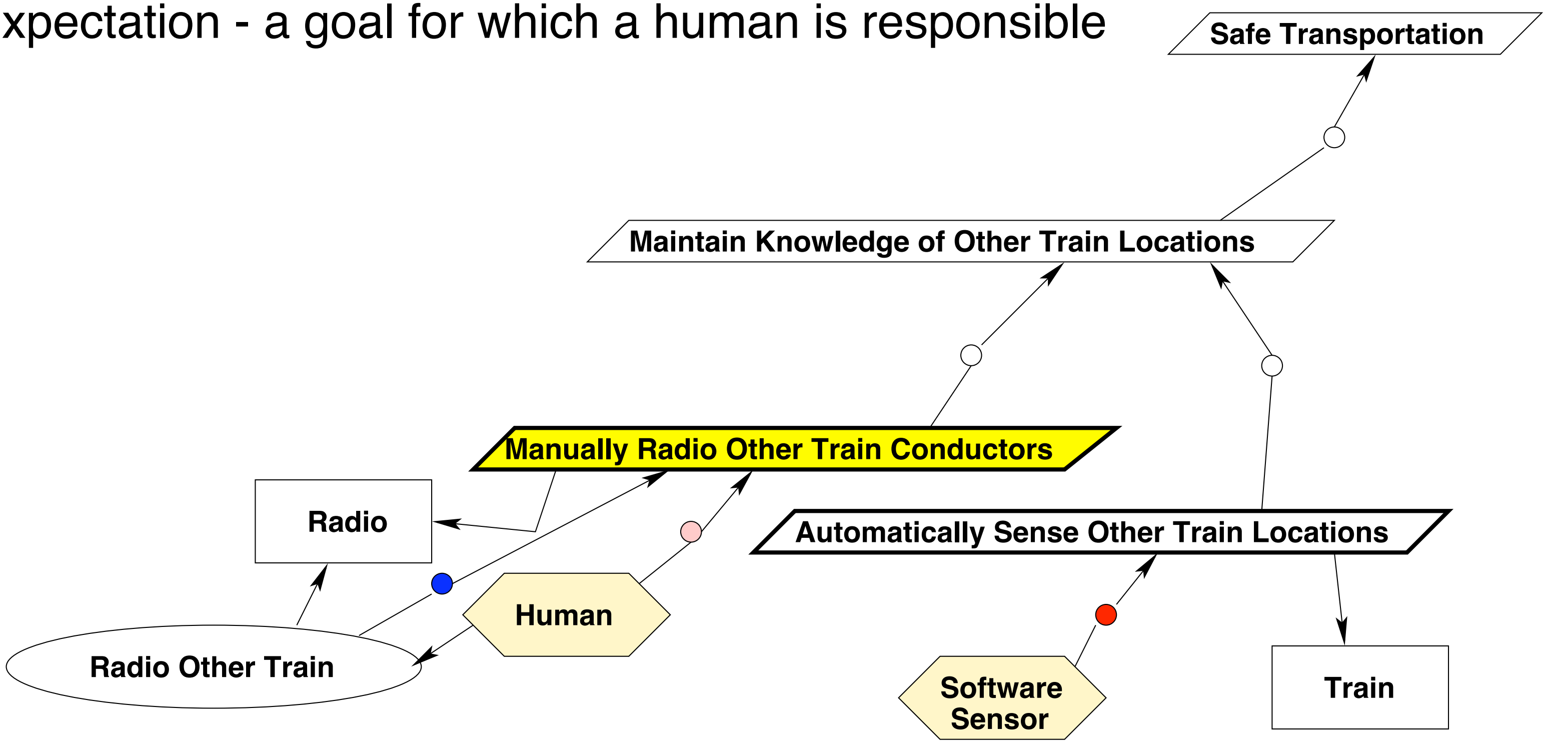
Integrating Goals with Other Models

Integrated Use of Goals

- KAOS
  - – Refining goals into requirements
  - – 4 models
    - **Goal**
    - Agent
    - Operationalization
    - Object

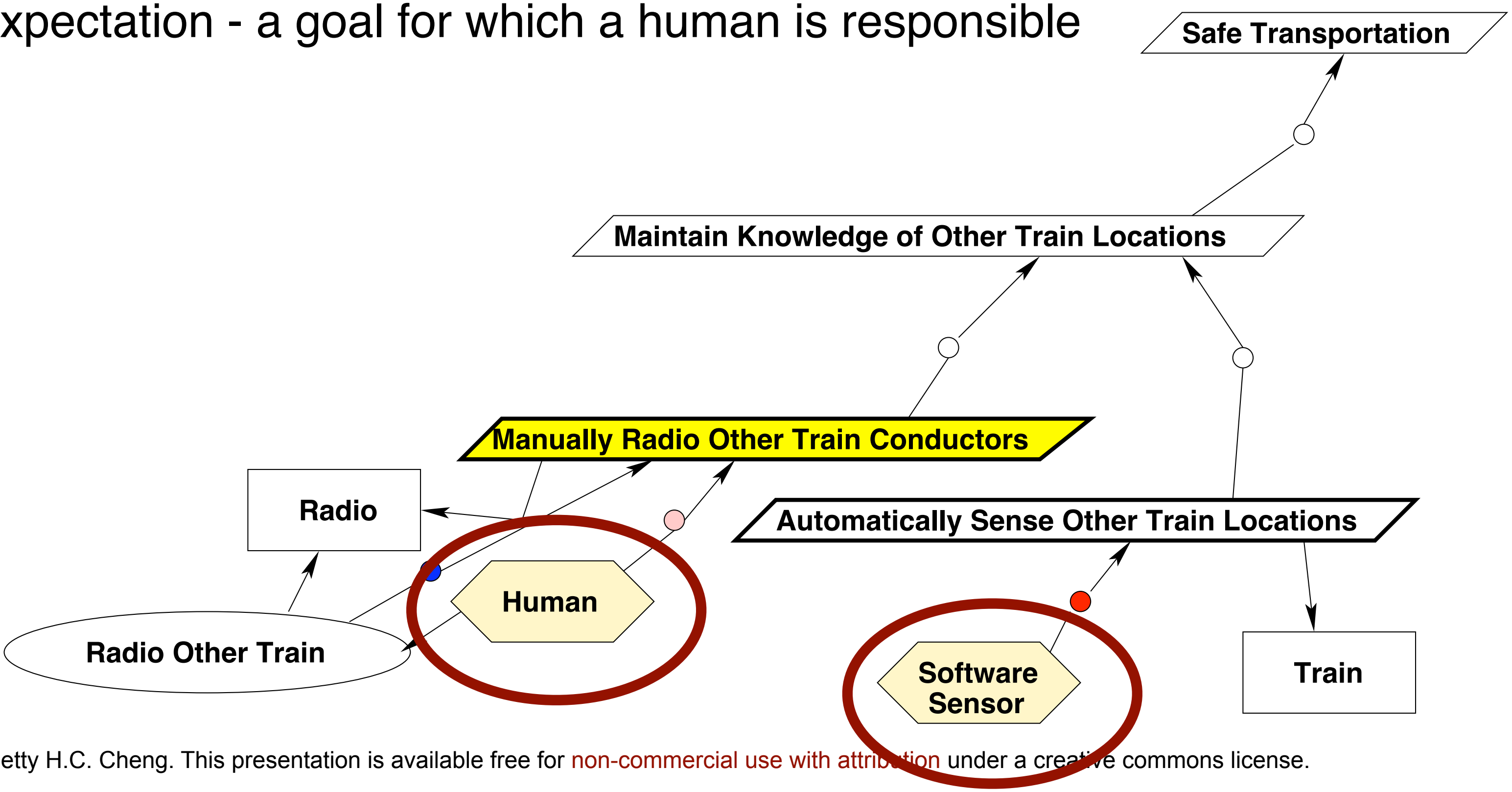
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Radio
Automatically Sense Other Train Locations
Human
Radio Other Train
Software Sensor
Train

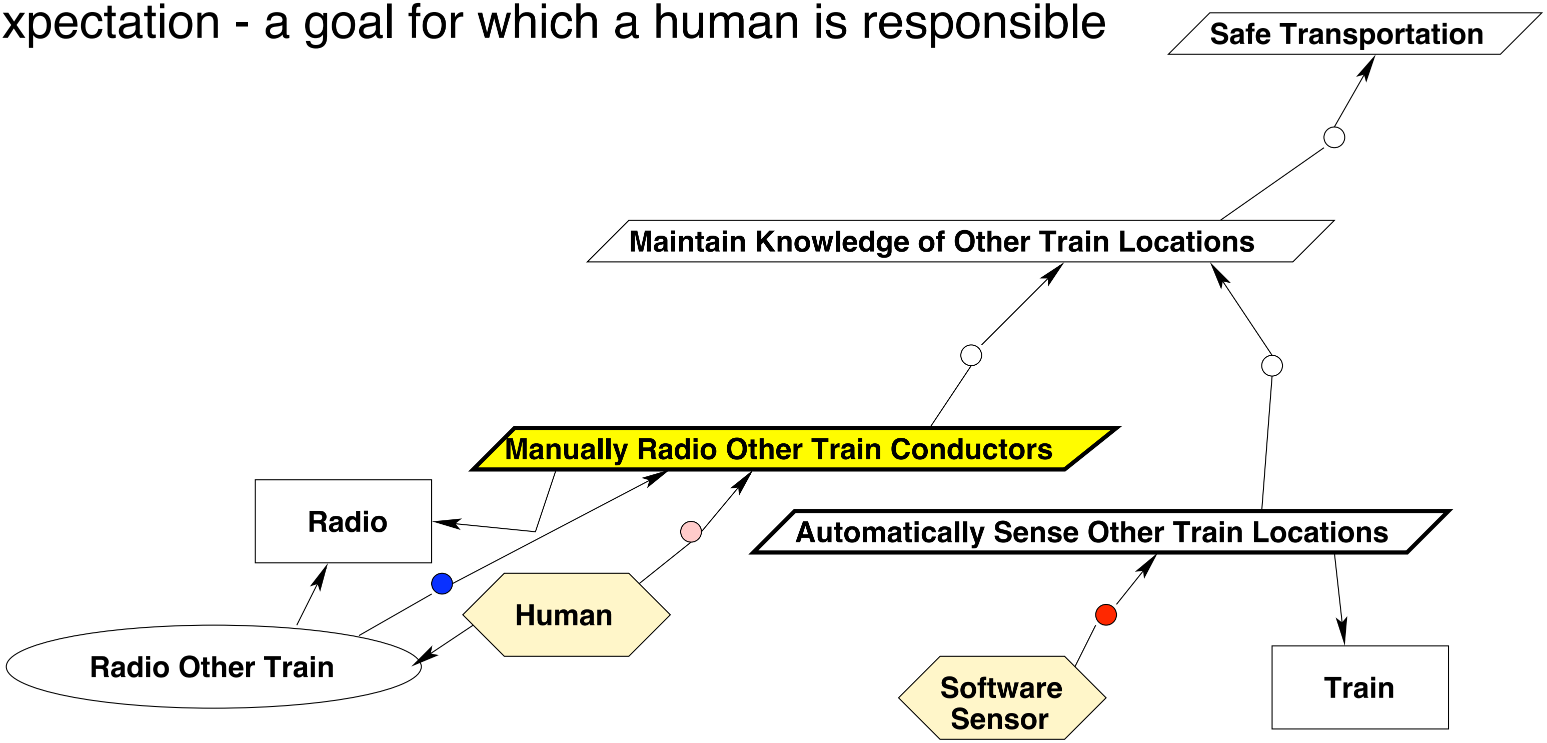
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

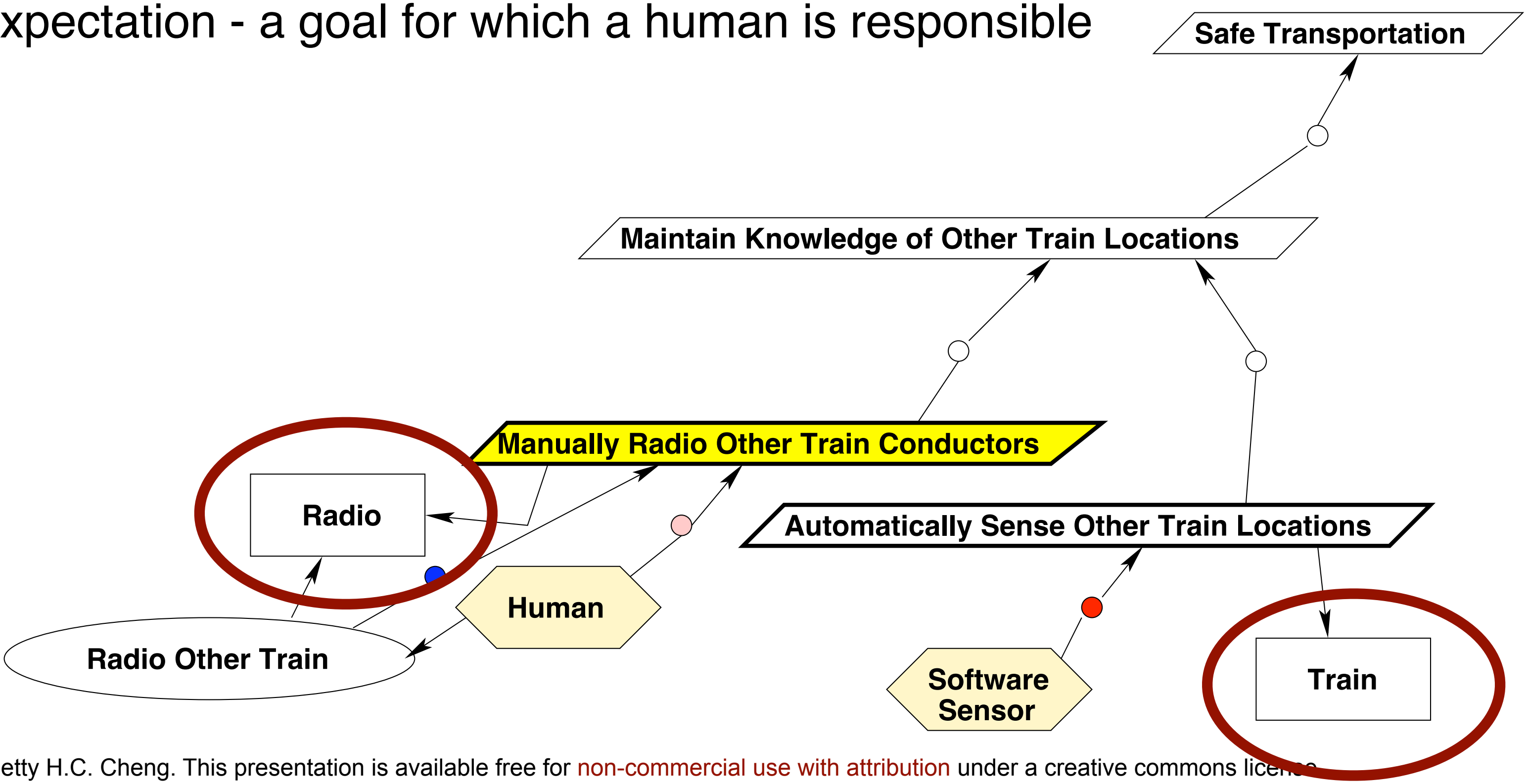
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Radio
Automatically Sense Other Train Locations
Human
Radio Other Train
Software Sensor
Train

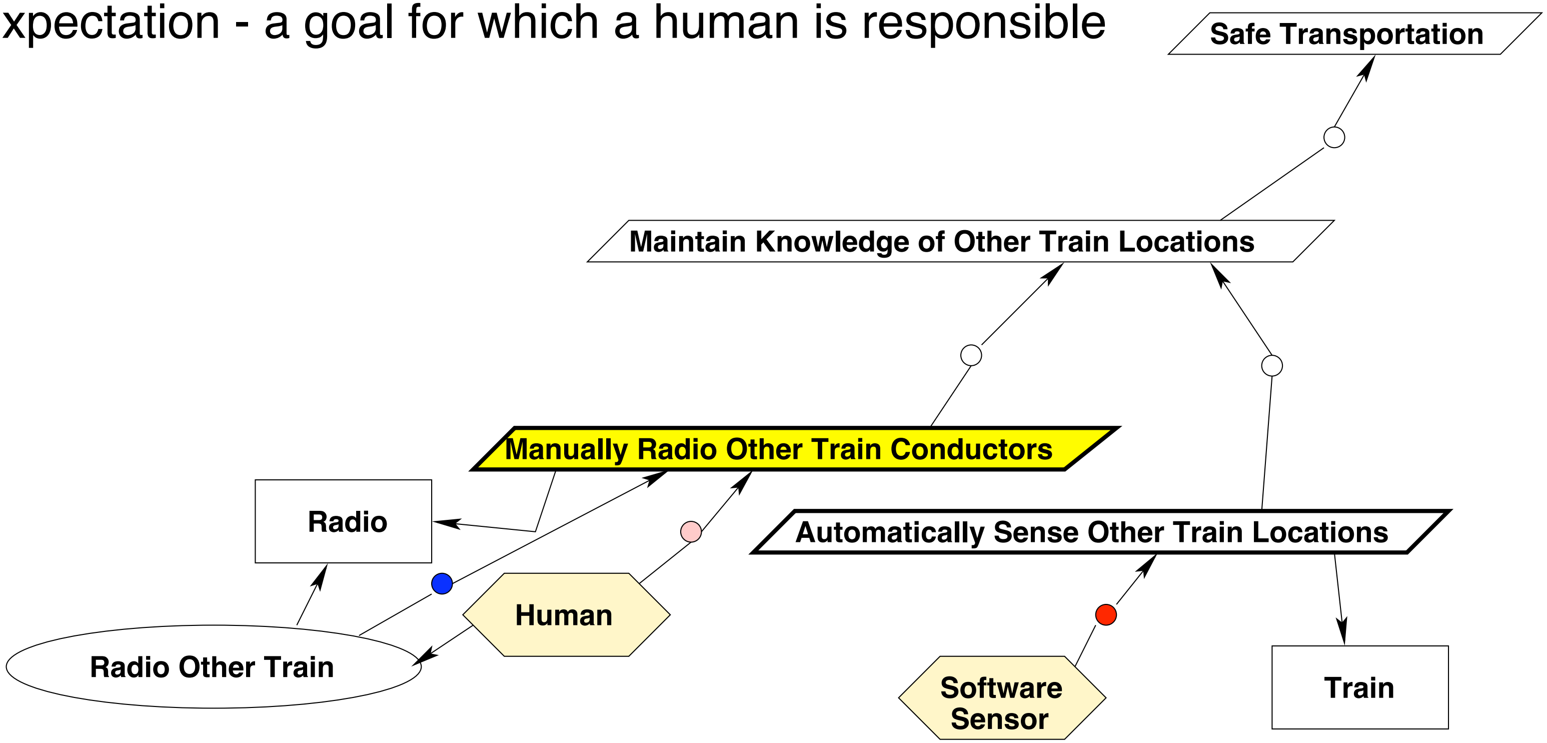
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

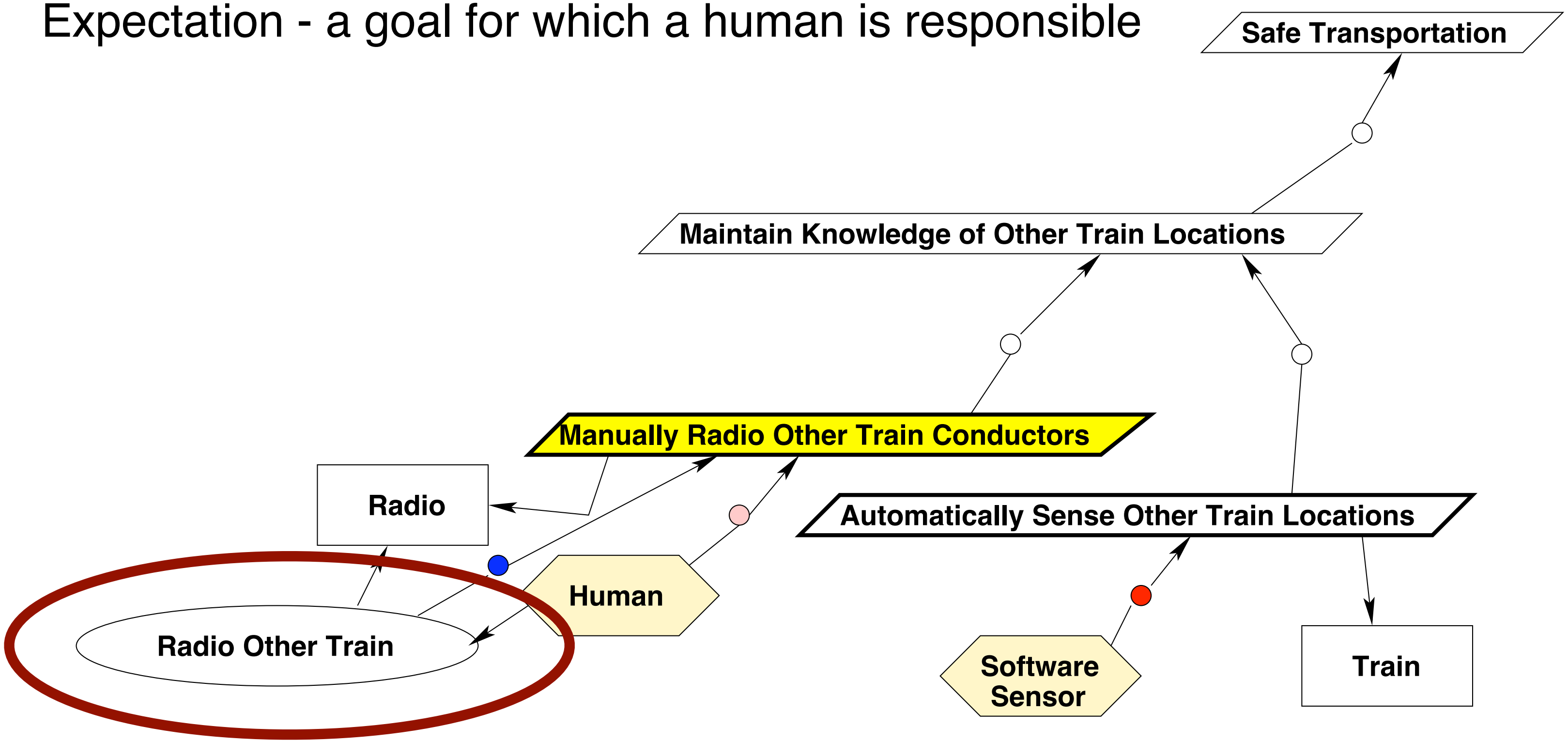
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Radio
Automatically Sense Other Train Locations
Human
Radio Other Train
Software Sensor
Train

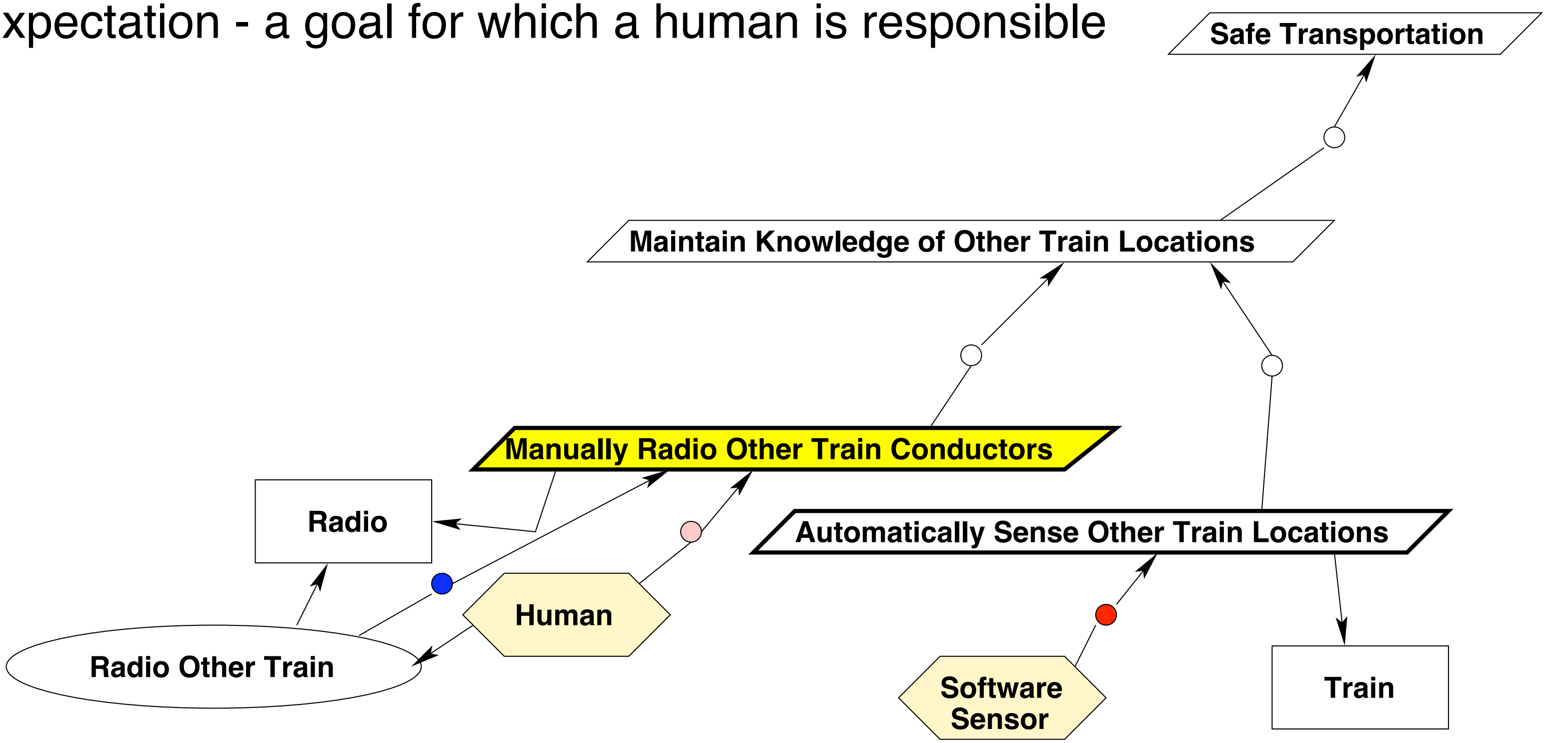
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

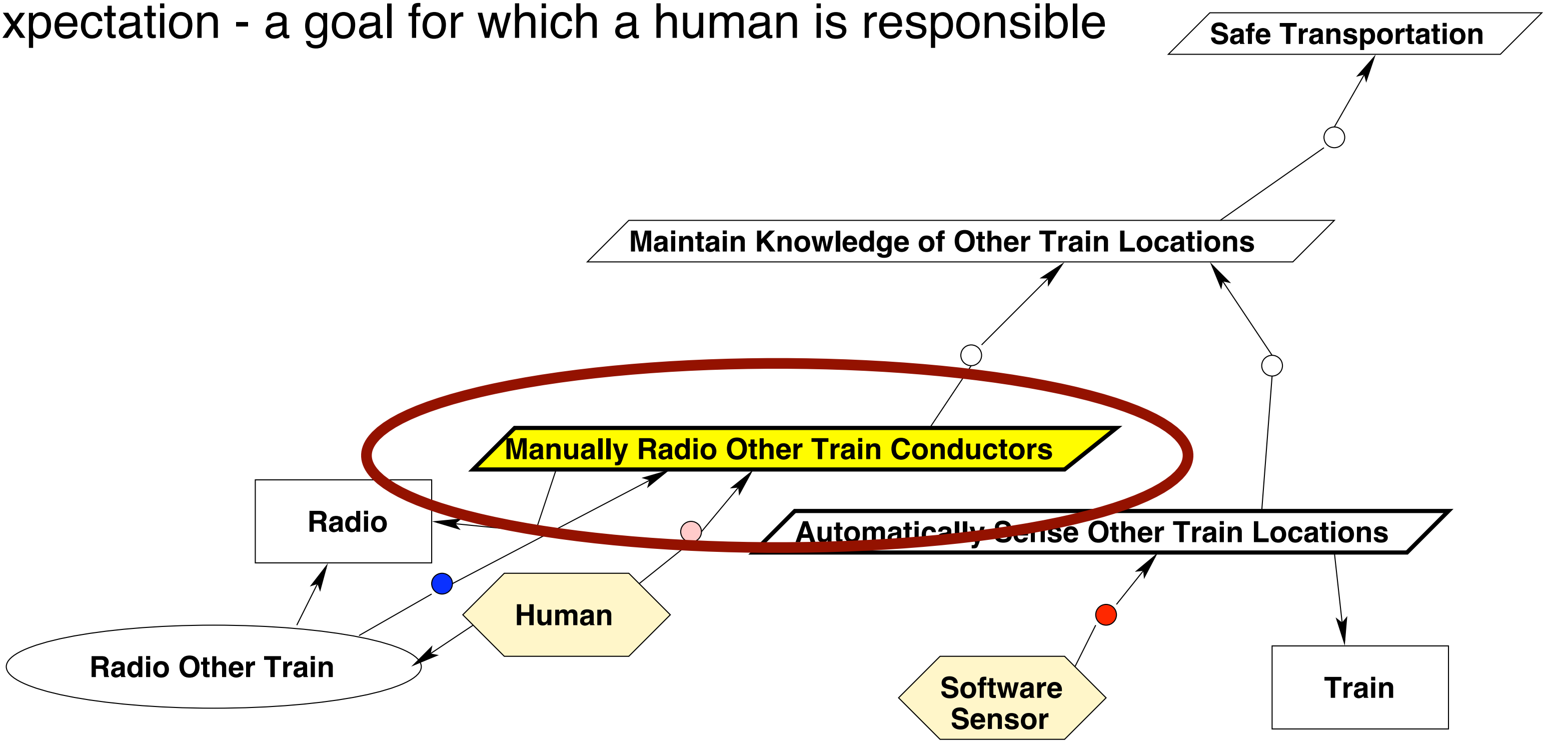
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

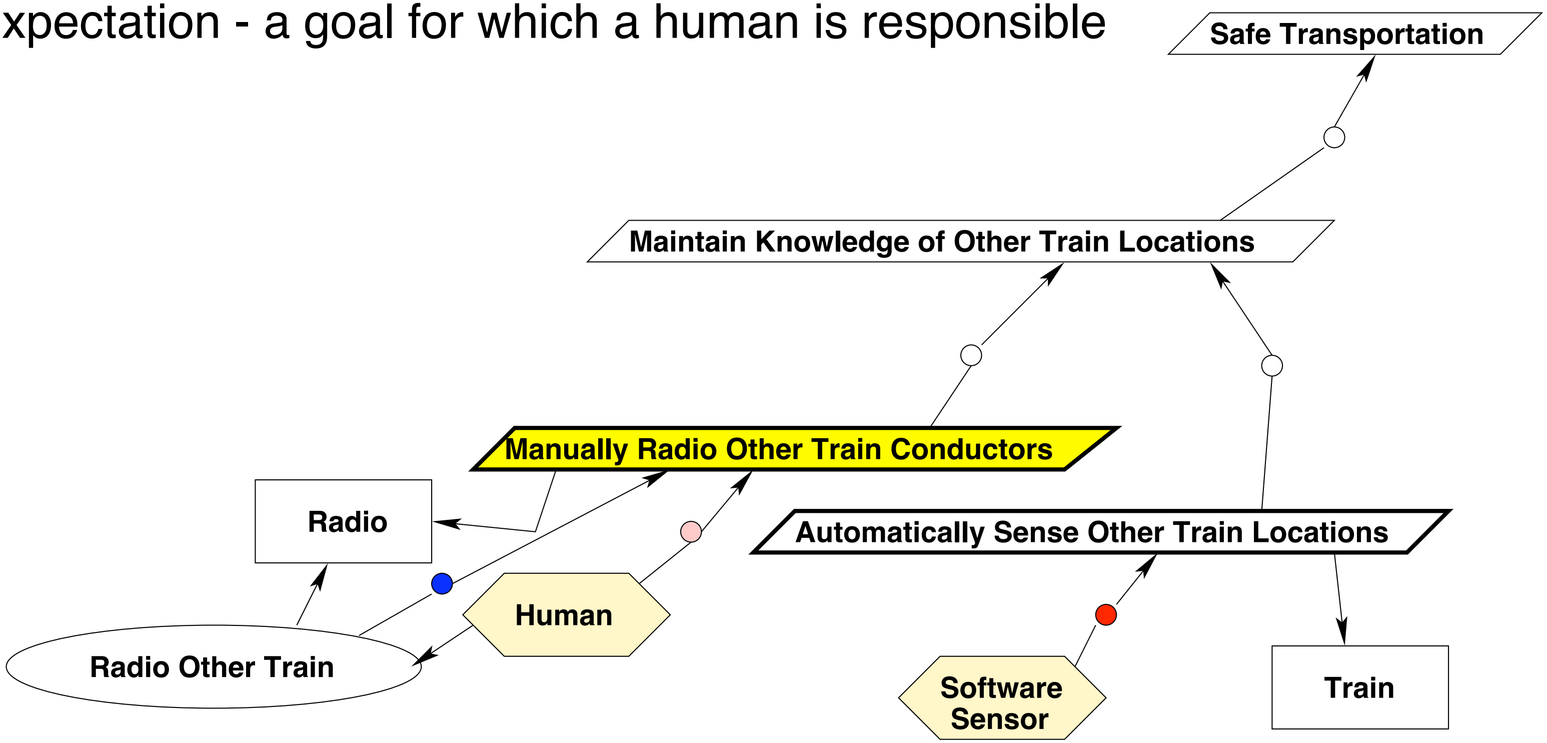
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Radio
Automatically Sense Other Train Locations
Human
Radio Other Train
Software Sensor
Train

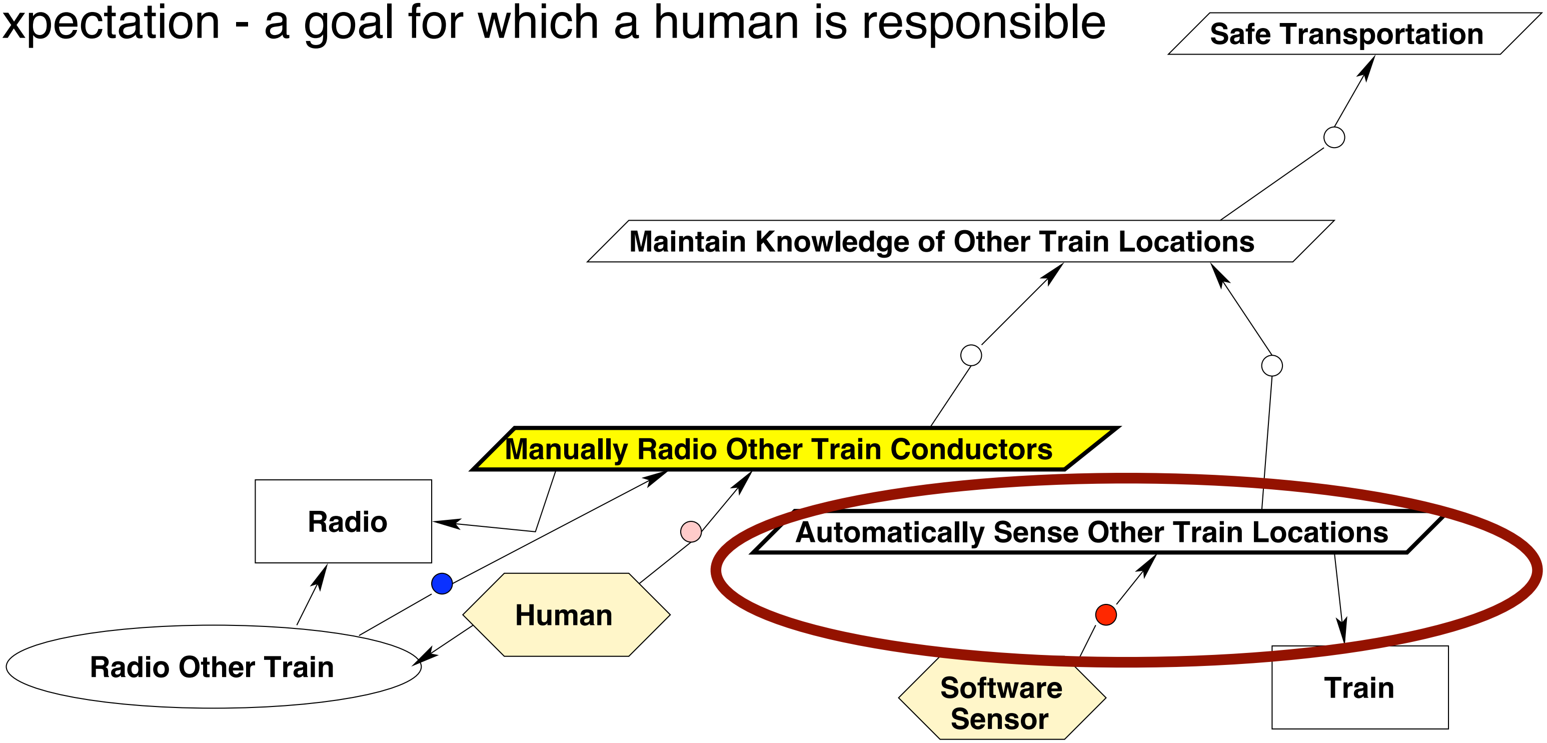
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Radio
Automatically Sense Other Train Locations
Human
Radio Other Train
Software Sensor
Train

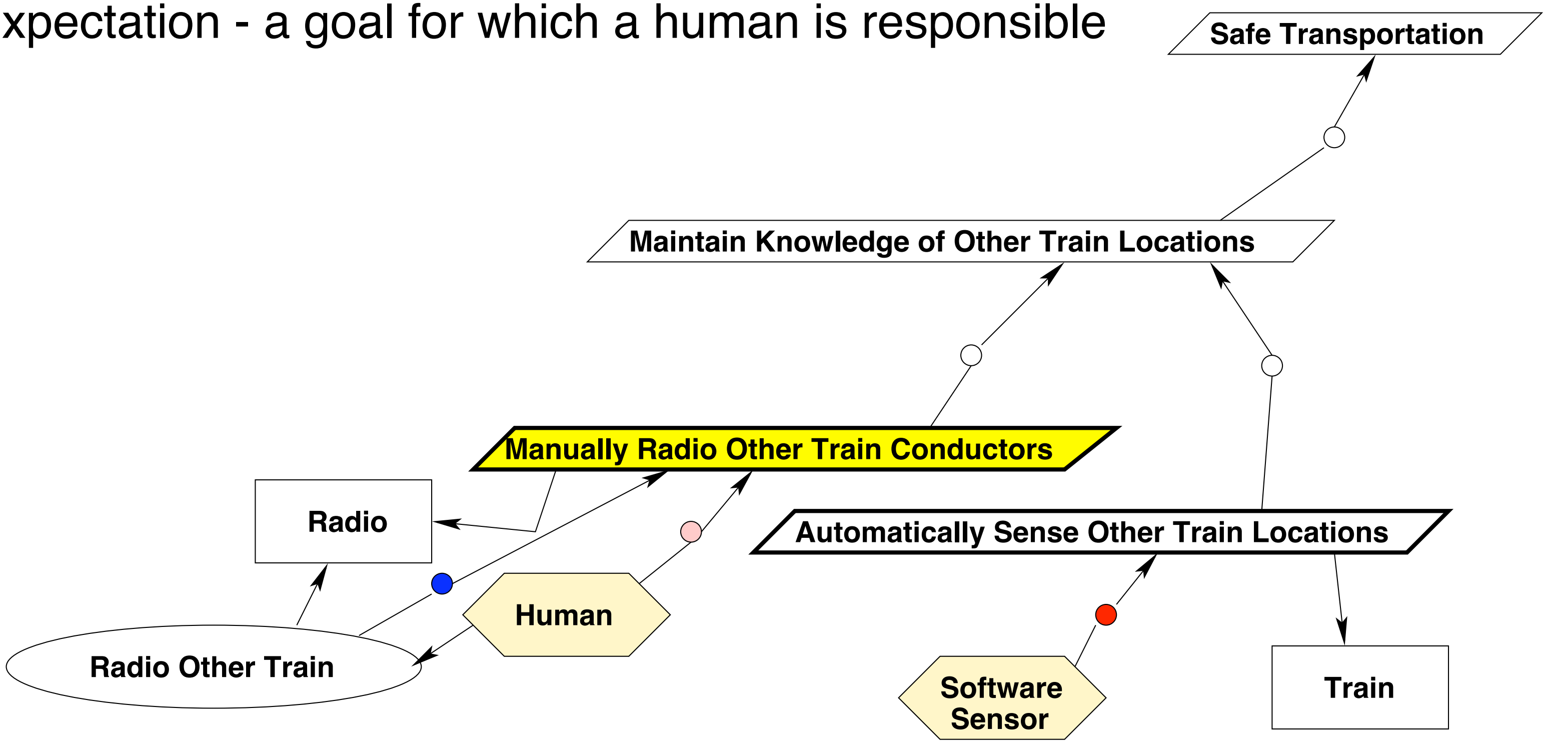
KAOS Goal Model

- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

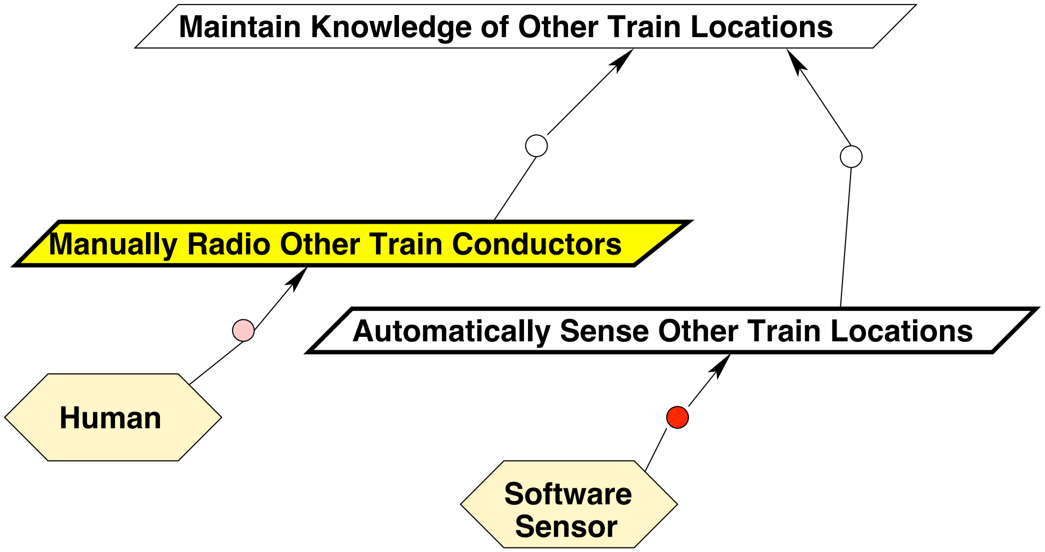
KAOS Goal Model

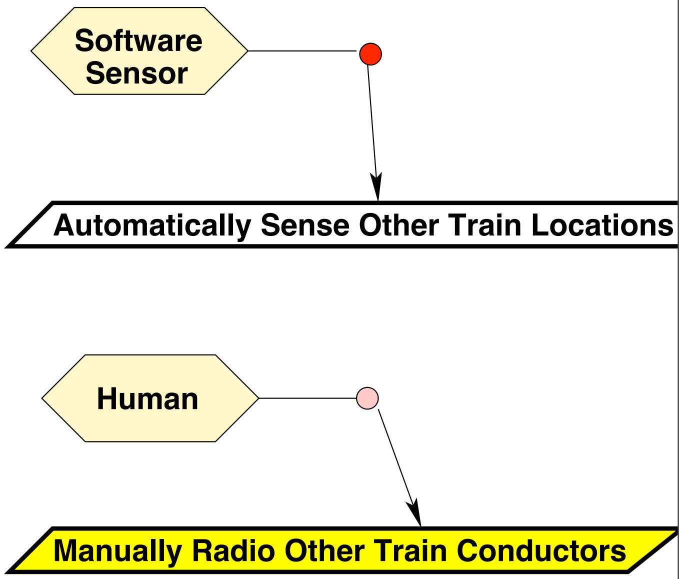
- Agent - active system component
- Object - inactive system component
- Operation - an action an agent takes to achieve a goal
- Requirement - a goal for which an automated component is responsible
- Expectation - a goal for which a human is responsible


Safe Transportation
Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Radio
Human
Radio Other Train
Software Sensor
Train

KAOS Agent Model

- Objective: depicts agent responsibilities
- Agent models can be inferred from goal models


Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Human
Software Sensor

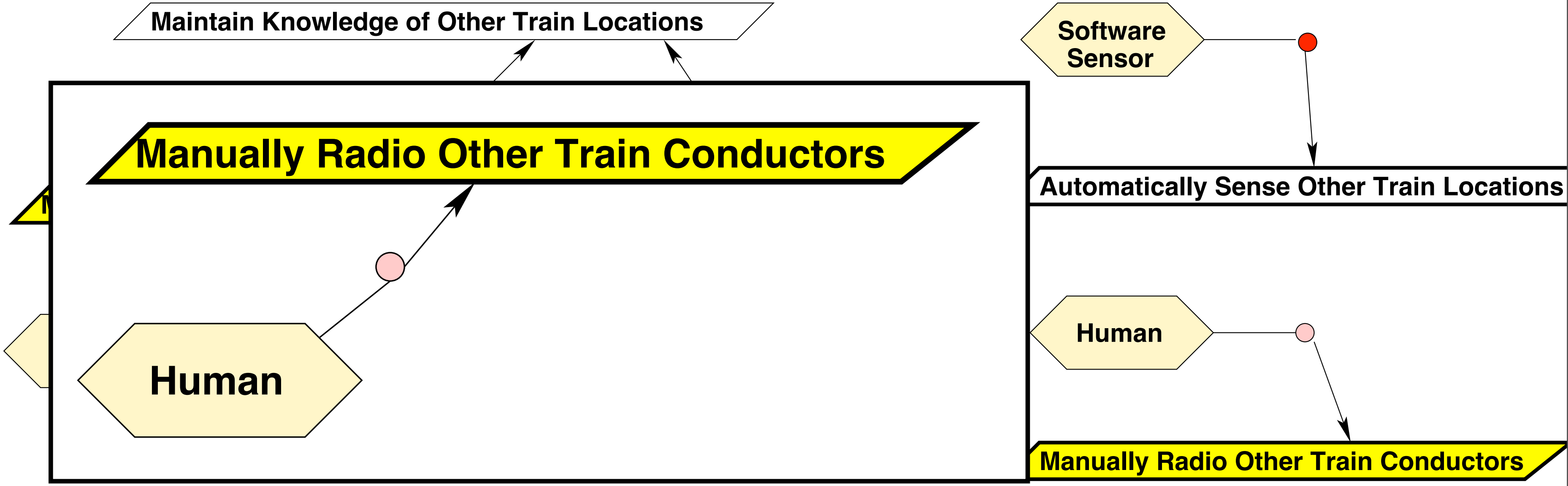

Software Sensor
Automatically Sense Other Train Locations
Human
Manually Radio Other Train Conductors

Goal Model

Agent Model

KAOS Agent Model

- Objective: depicts agent responsibilities
- Agent models can be inferred from goal models

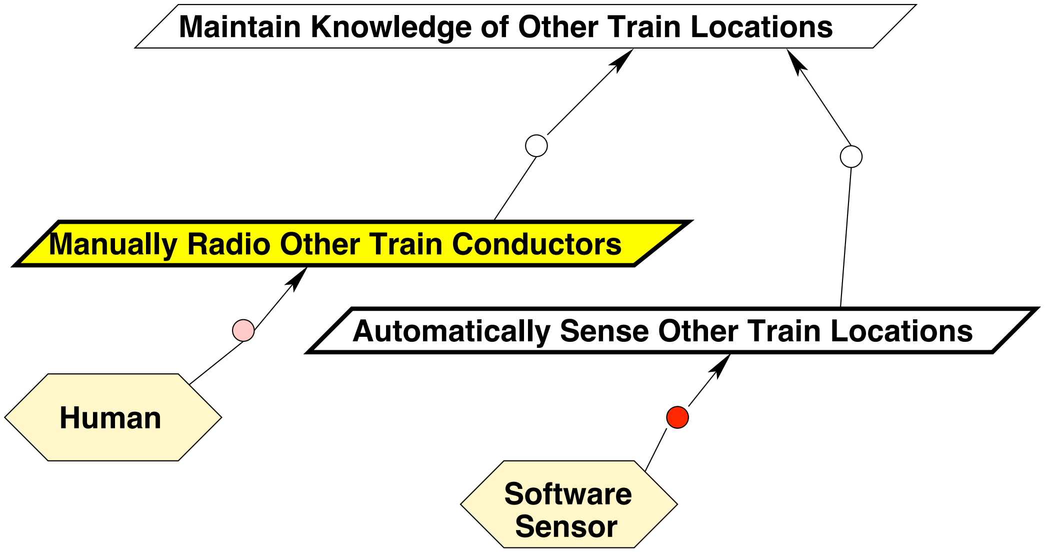

Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Human
Software Sensor
Automatically Sense Other Train Locations
Human
Manually Radio Other Train Conductors

Goal Model

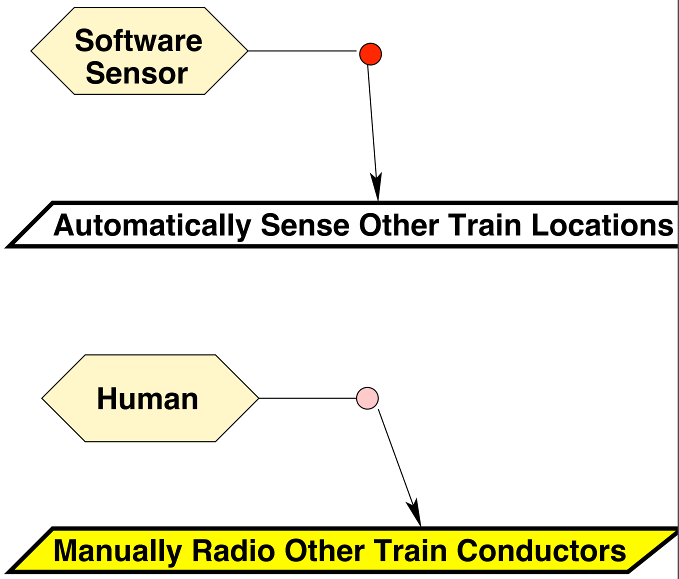
Agent Model

KAOS Agent Model

- Objective: depicts agent responsibilities
- Agent models can be inferred from goal models


Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Human
Software Sensor

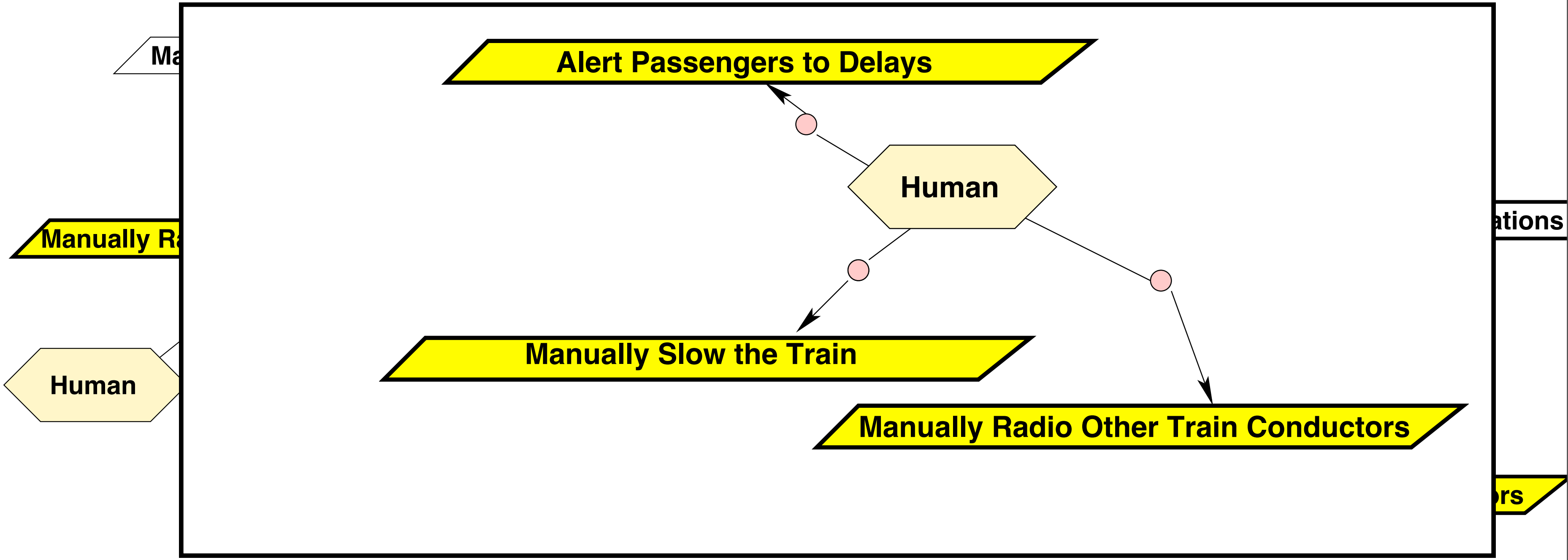
Goal Model


Software Sensor
Automatically Sense Other Train Locations
Human
Manually Radio Other Train Conductors

Agent Model

KAOS Agent Model

- Objective: depicts agent responsibilities
- Agent models can be inferred from goal models

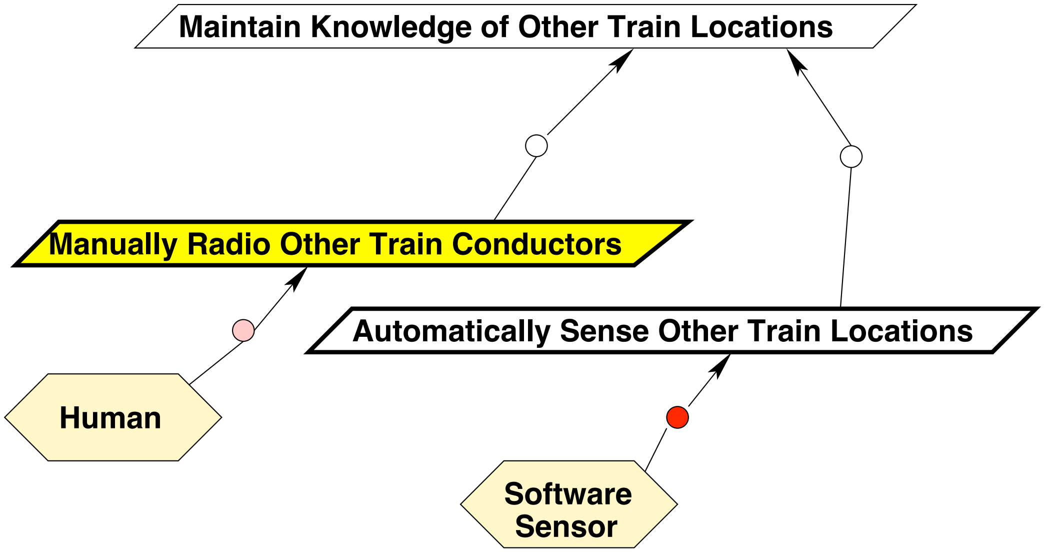

Ma
Manually Ra
Human
ations
ors
Alert Passengers to Delays
Human
Manually Slow the Train
Manually Radio Other Train Conductors

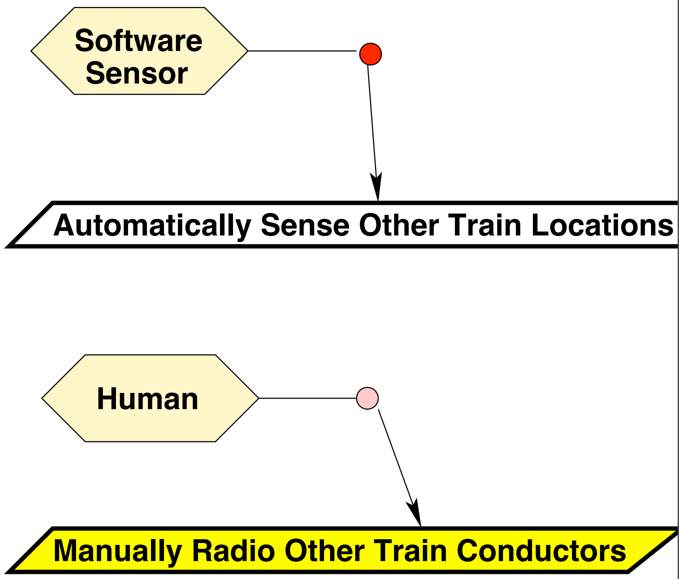
Goal Model

Agent Model

KAOS Agent Model

- Objective: depicts agent responsibilities
- Agent models can be inferred from goal models


Maintain Knowledge of Other Train Locations
Manually Radio Other Train Conductors
Automatically Sense Other Train Locations
Human
Software Sensor

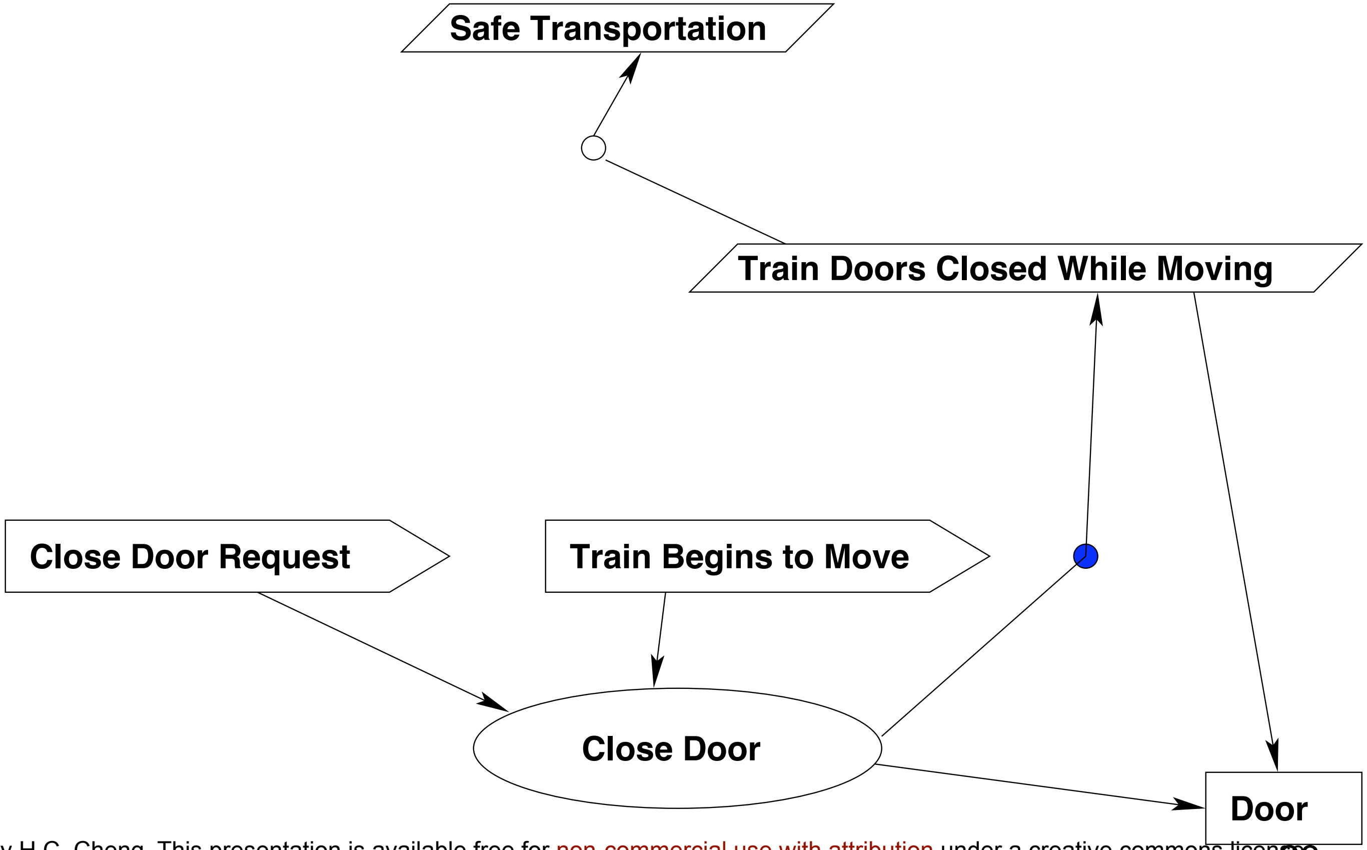

Software Sensor
Automatically Sense Other Train Locations
Human
Manually Radio Other Train Conductors

Goal Model

Agent Model

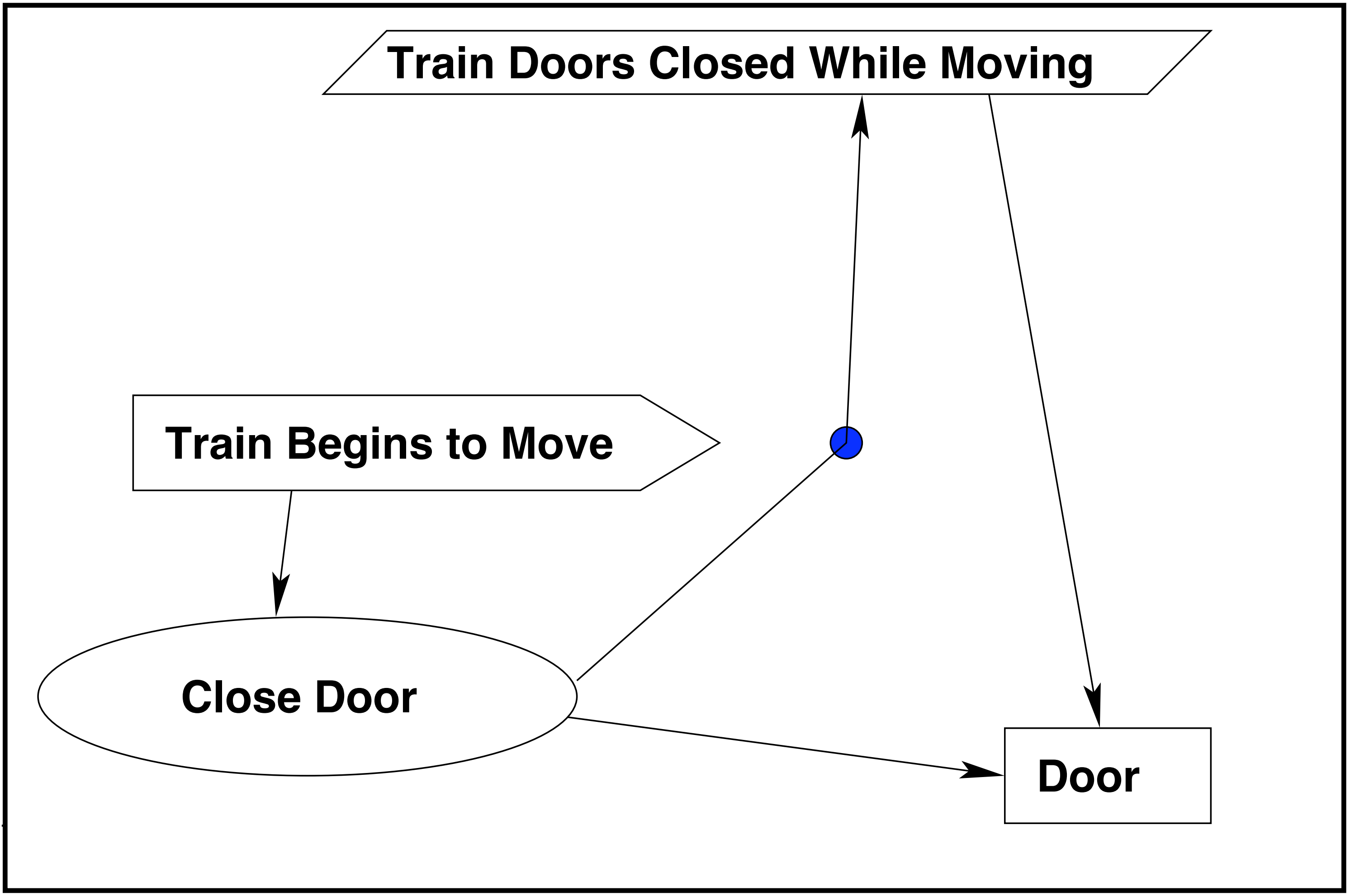
KAOS Operationalization Model

- Objective: specifies the operations that agents must perform to achieve the goals



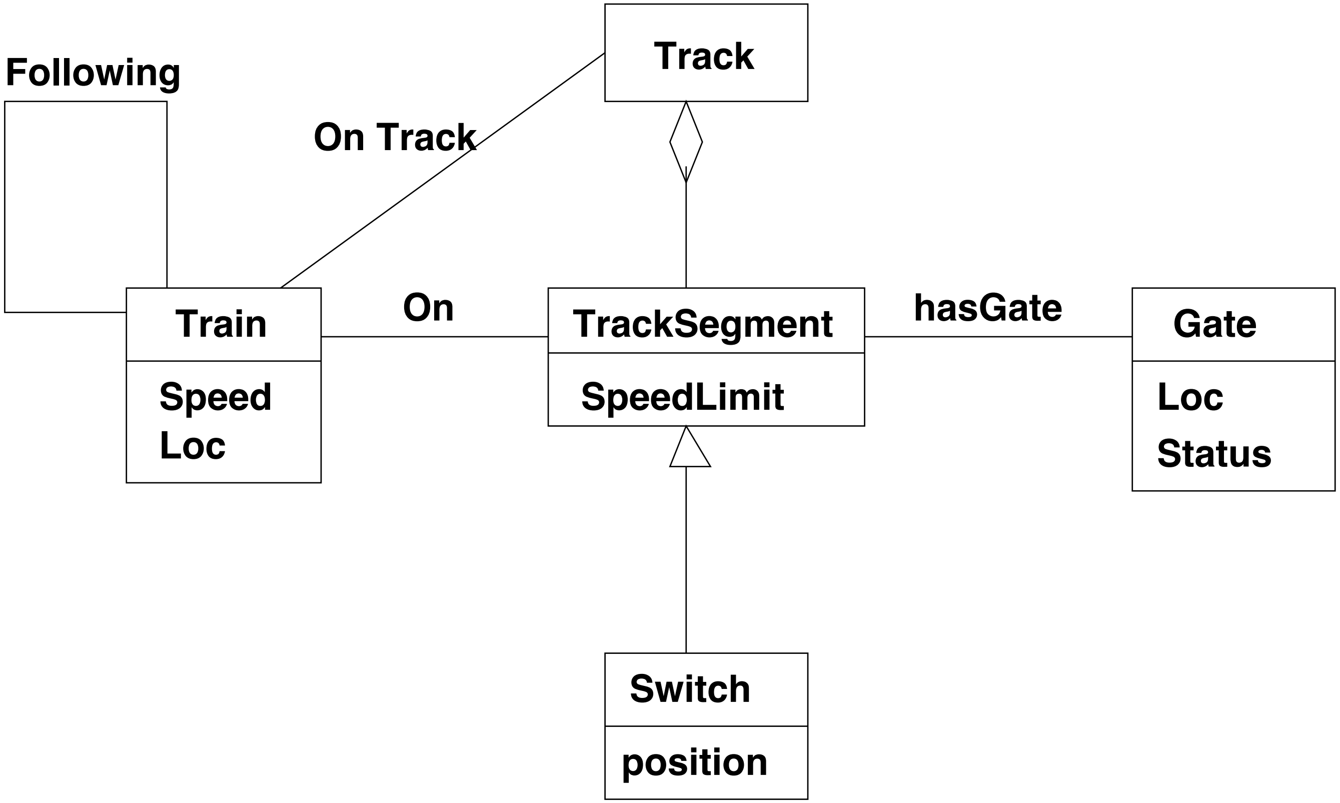
KAOS Operationalization Model

- Objective: specifies the operations that agents must perform to achieve the goals



KAOS Object Model

- Objective: further specifies objects used in the goal model
- The syntax is similar to a that of a UML class diagram



Integrated Use of Goals

- KAOS
  - Refining goals into requirements
  - 4 models
    - **Goal**
    - Agent
    - Operationalization
    - Object
- i*
  - Relates goals to the organization context
  - 2 models
    - Actor (Strategic) Dependency Model
    - Actor (Strategic) Rationale Model

i* Actor Dependency Model

- Dependencies between actors - An actor is a black box

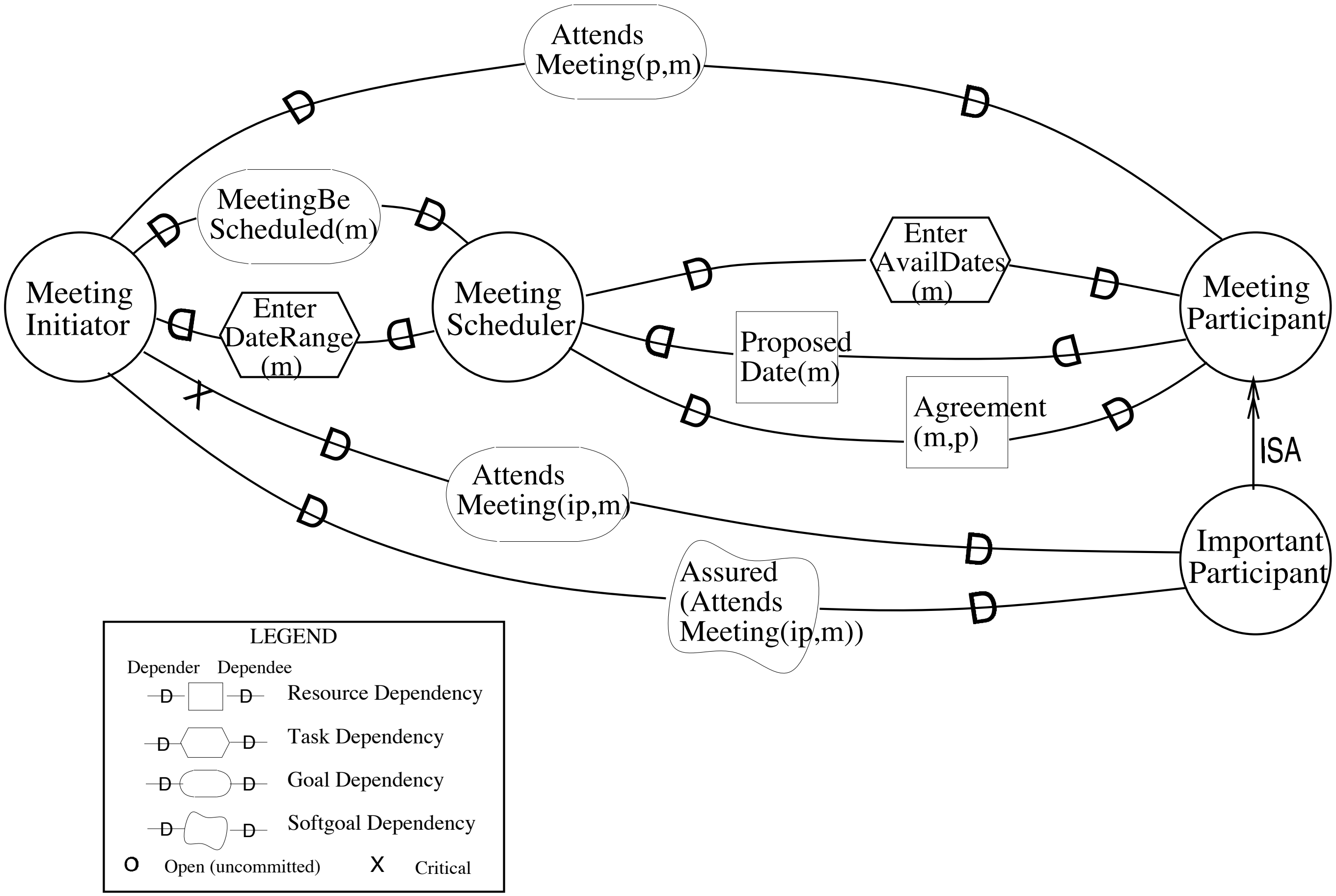


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu

i* Actor Dependency Model

- Dependencies between actors - An actor is a black box

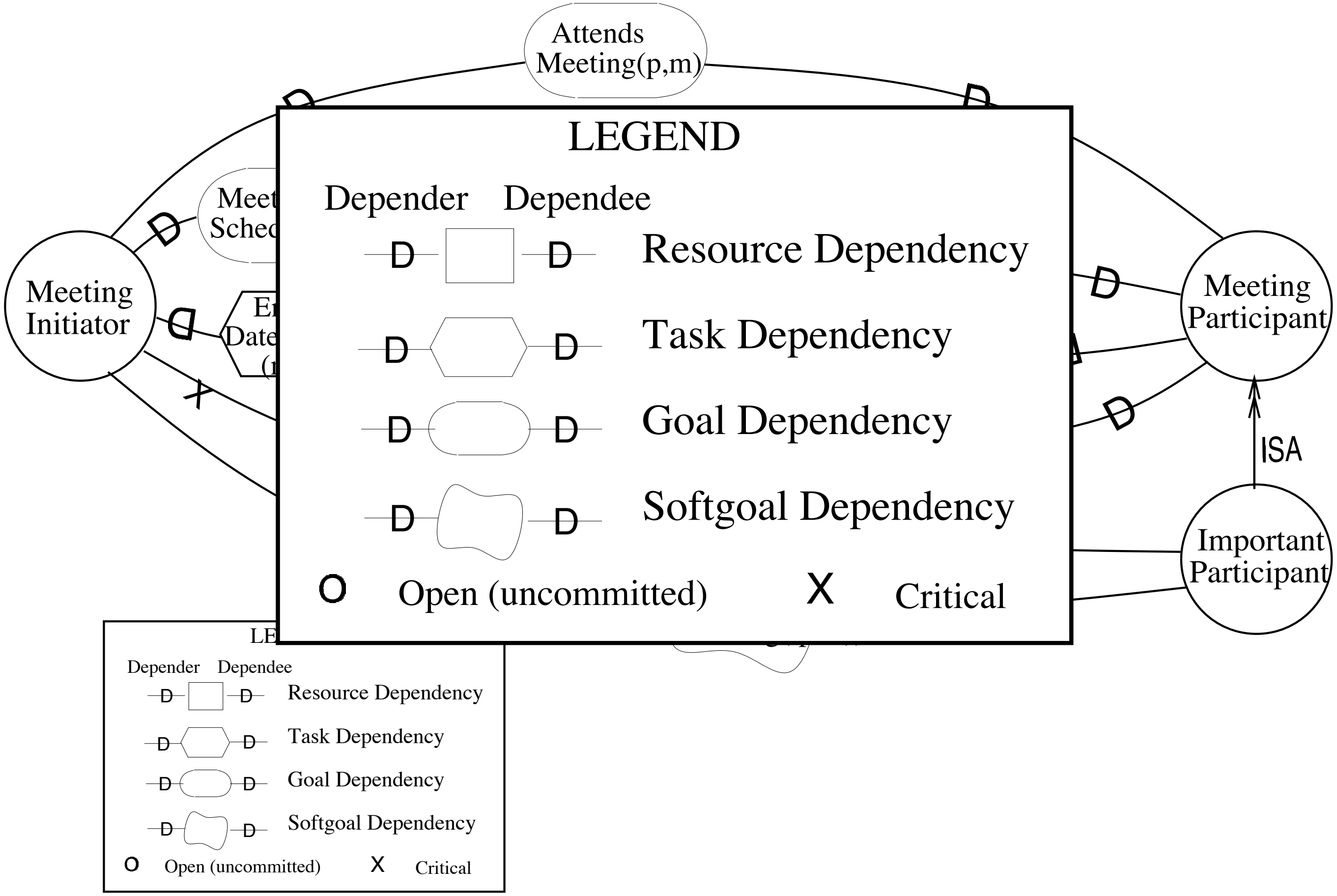


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu

i* Actor Dependency Model

- Dependencies between actors - An actor is a black box

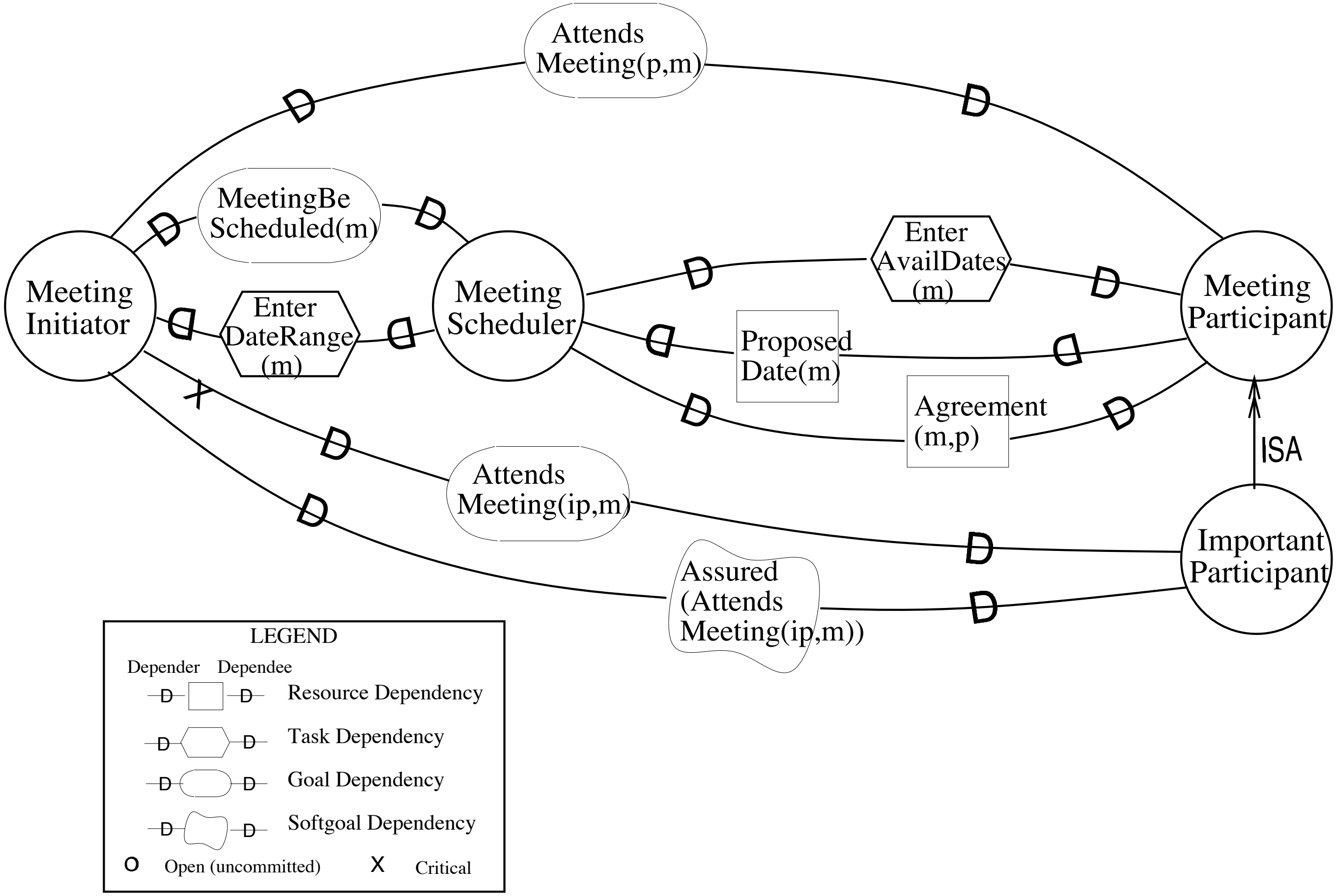


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu

i* Actor Rationale Model

- Internal actor relationships - An actor is a white box

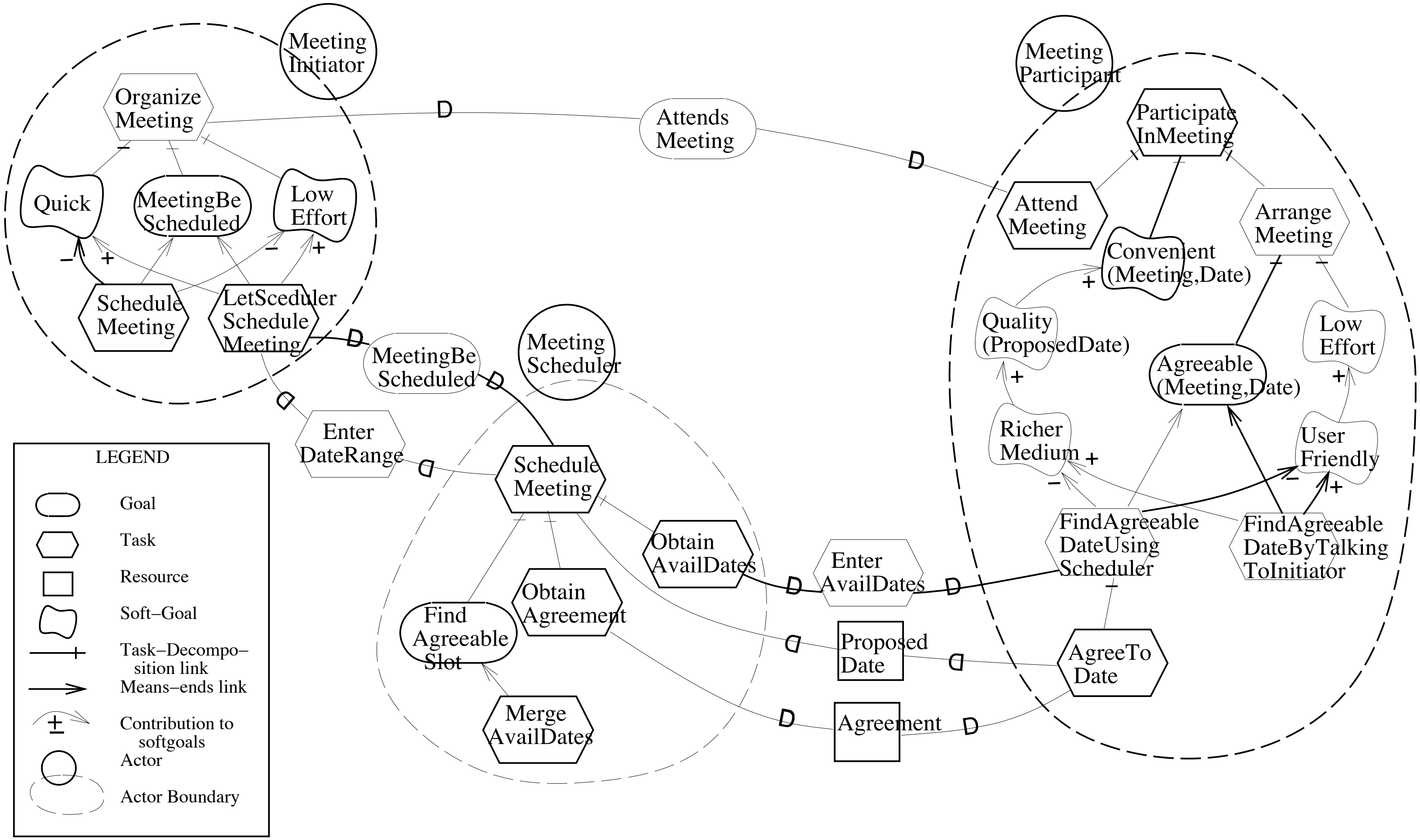
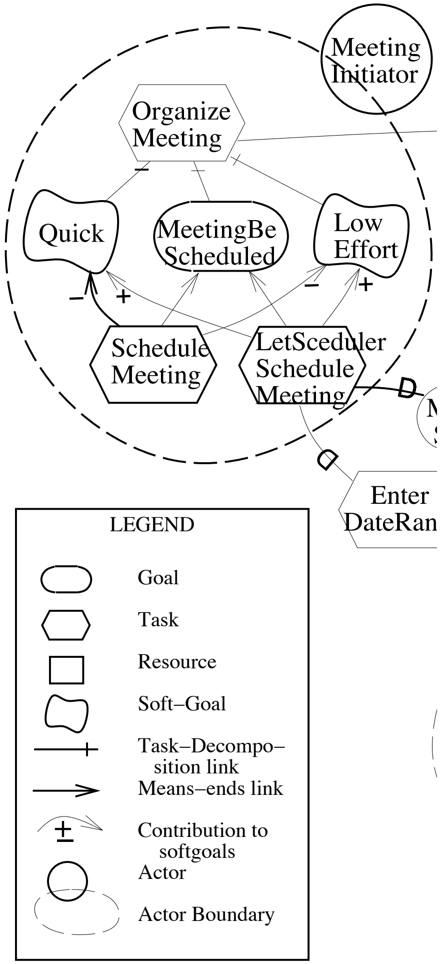
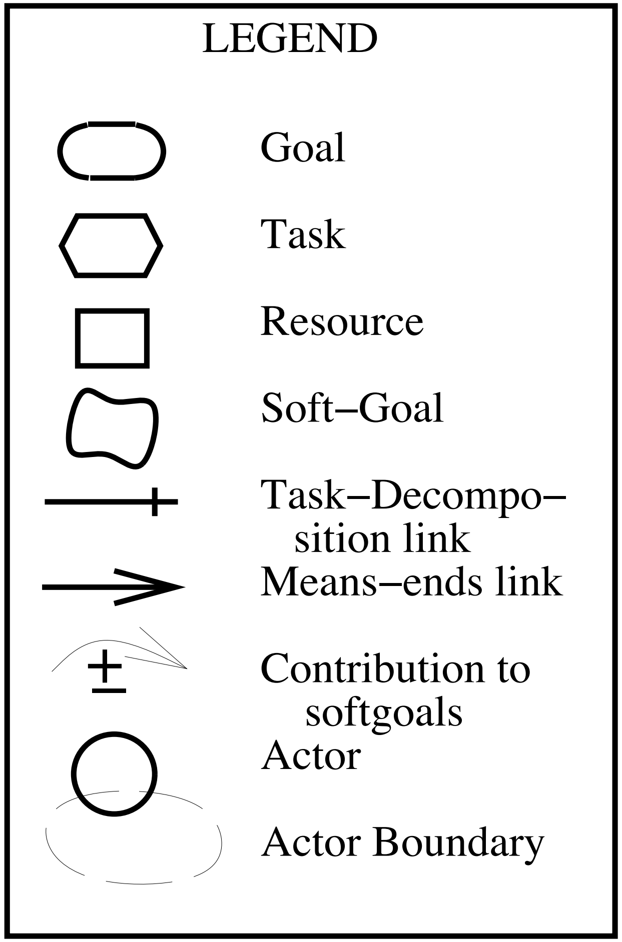


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu

i* Actor Rationale Model

- Internal actor relationships - An actor is a white box





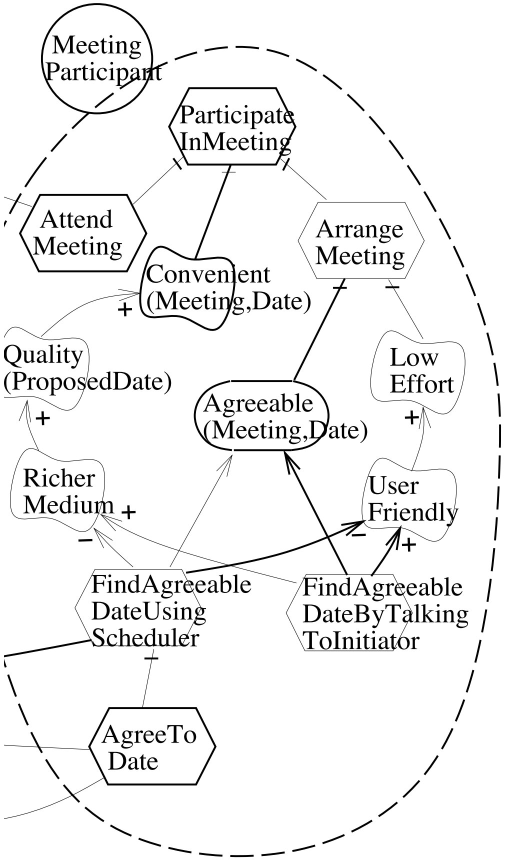


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu

i* Actor Rationale Model

- Internal actor relationships - An actor is a white box

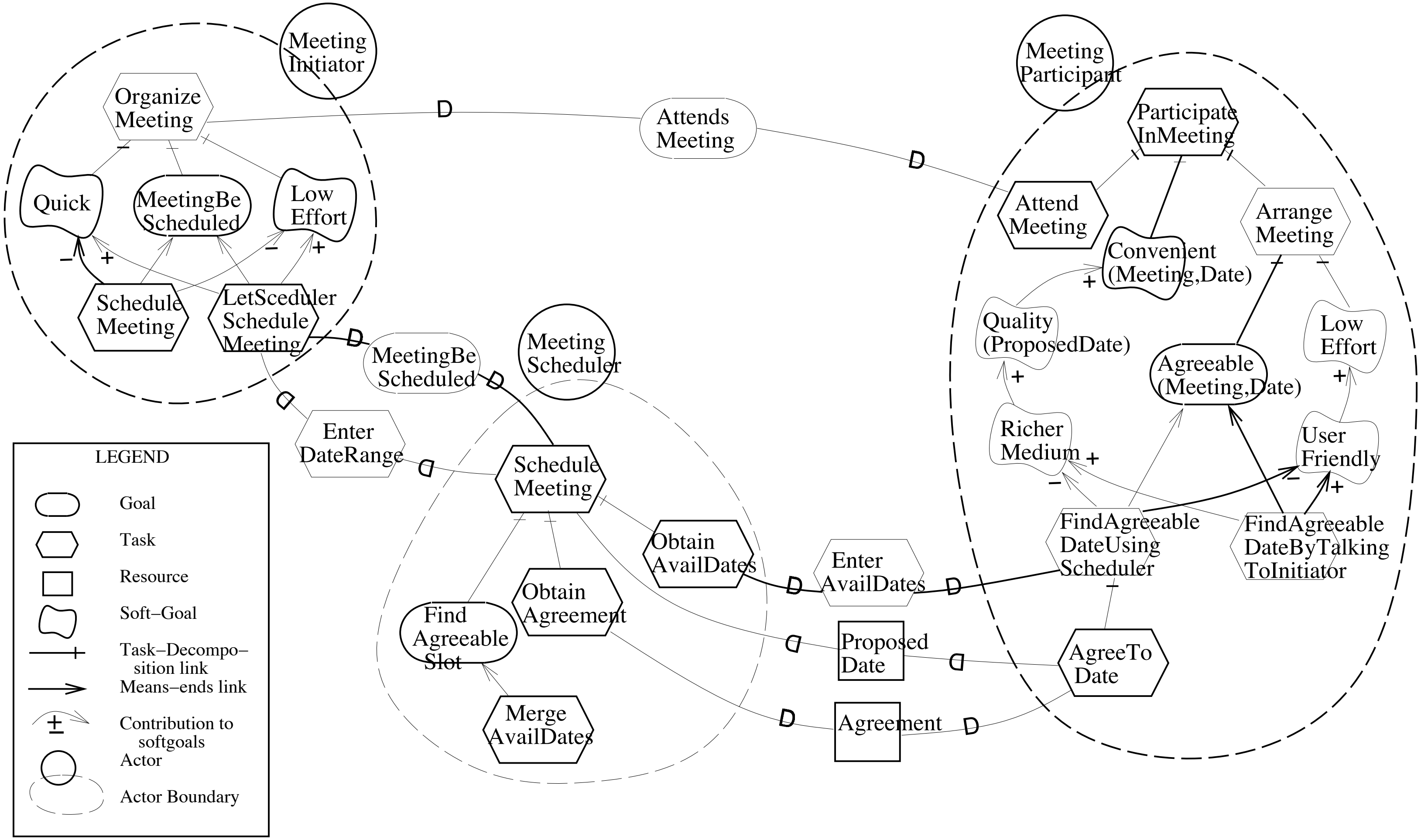


Diagram from *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering* by Eric Yu