The i* *conceptual model for requirements analysis*

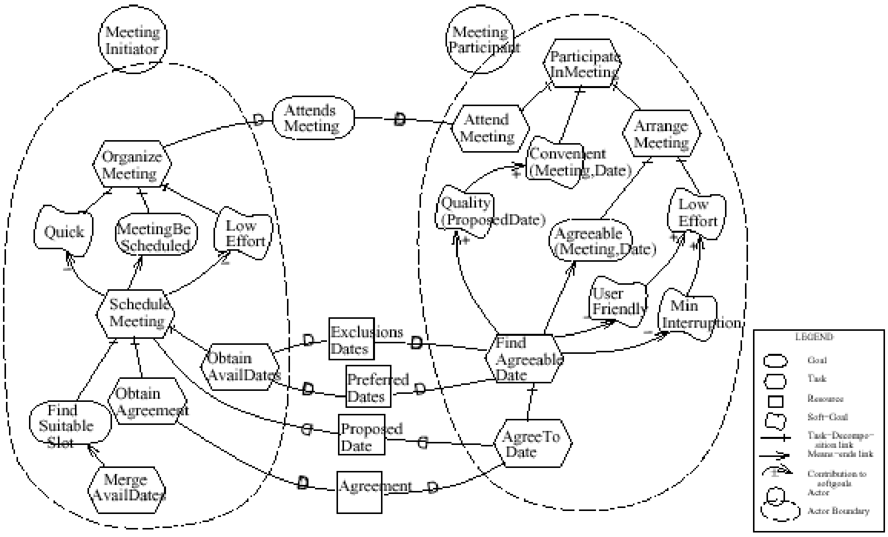
The Strategic Rationale Model

Examples + Exercise

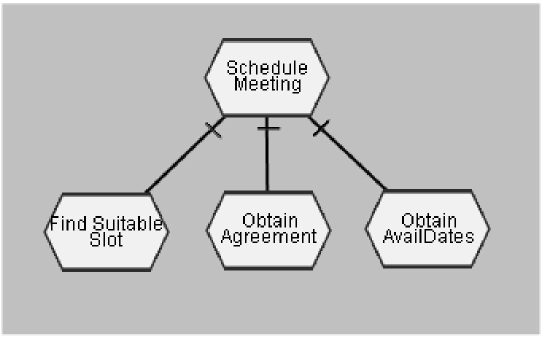
The Strategic Rationale (SR) model

- SD shows external relationships among actors, while hides the intentional constructs within each actor
- SR models internal intentional relationships inside each actor
- Intentional elements (goals, tasks, resources, and softgoals) appear in the SR model not only as external dependencies, but also as internal elements linked by **means-ends relationships** and **task-decompositions**

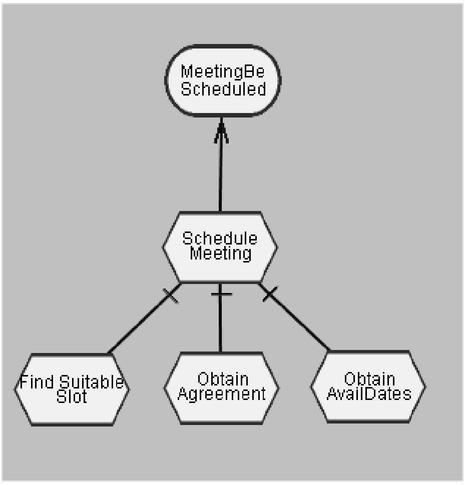
SR without the Meet. Sched.



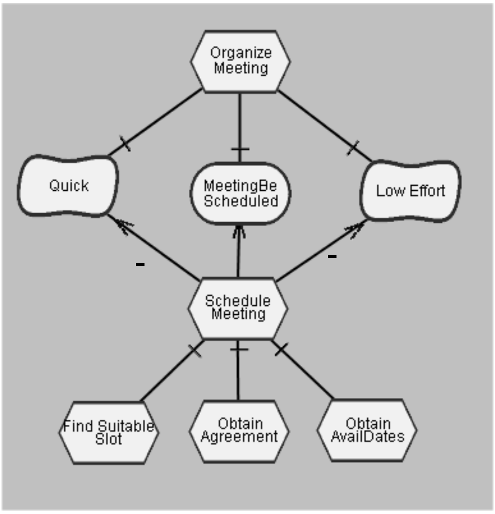
Task-decomposition



Means-end links

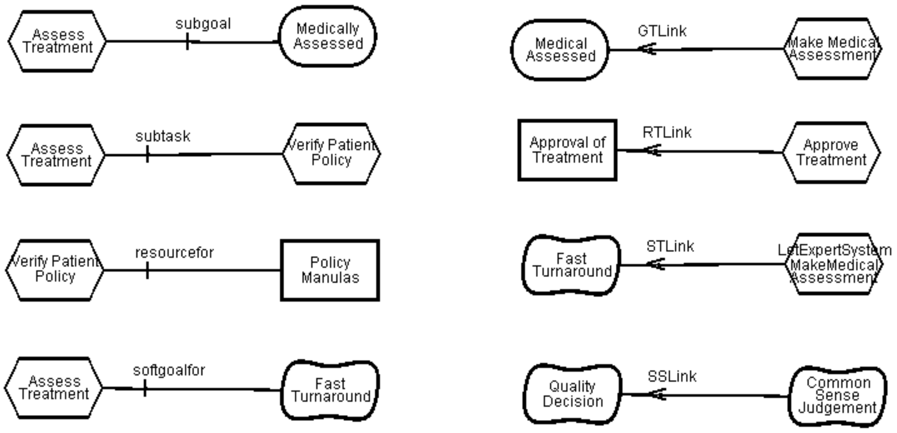


Contribution to GostGoals



(- -) (-) (+) (++)

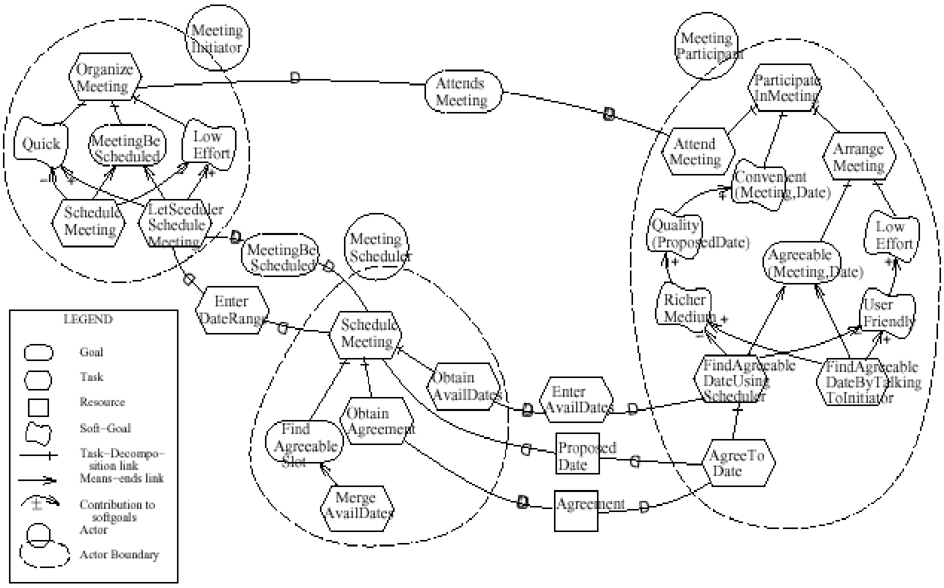
Link types



Decomposition links

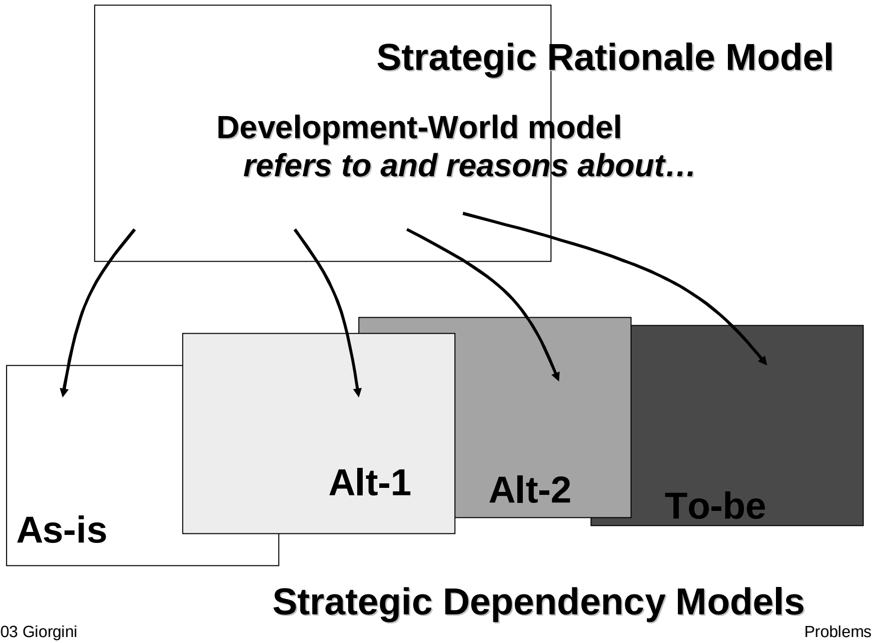
Means-ends links

SR with the Meet. Sched.



The SR model

- The SR model provides a way of modelling:
  - stakeholder interests, and how they might be met
  - the stakeholders evaluation of various alternatives with respect to their interests.
- Task-decomposition links provide a hierarchical description of intentional elements that make up a routine.
- The means-ends links in the SR provides understanding about why an actor would engage in some tasks, pursue a goal, need a resource, or want a softgoal.
- From the softgoals, one can tell why one alternative may be chosen over others



Strategic Rationale Model

Development-World model
refers to and reasons about…

As-is

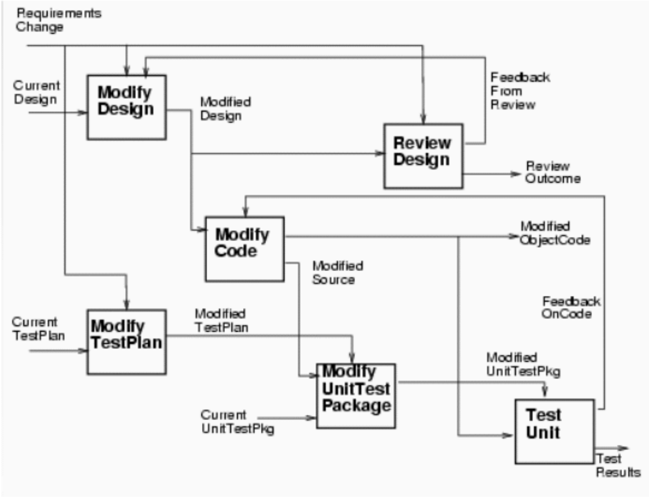
Alt-1

Alt-2

To-be

Strategic Dependency Models

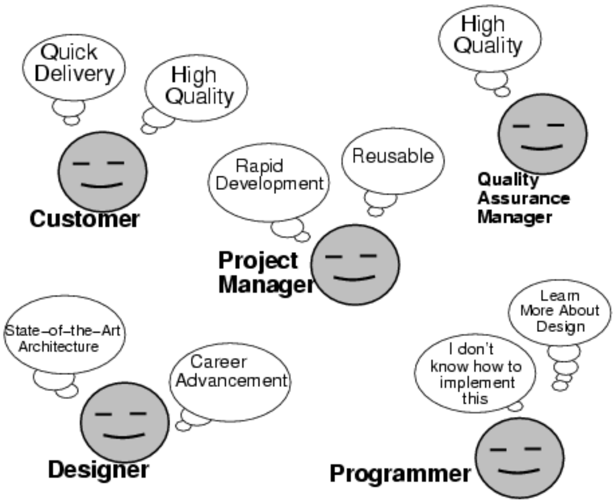
Software Process



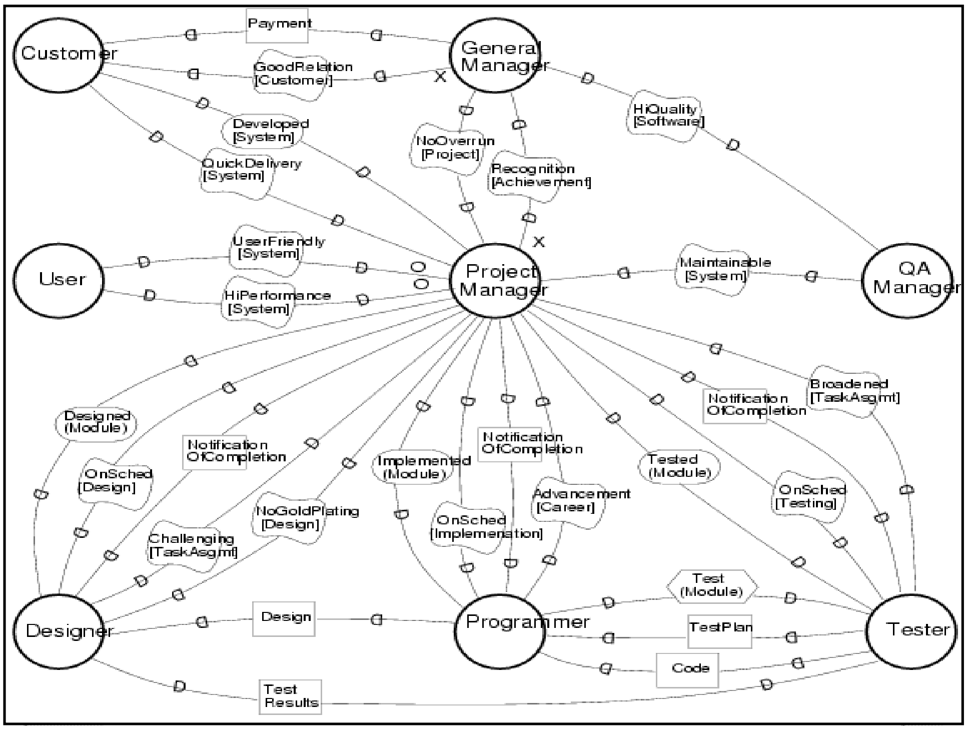
We need to understand **Why**
in order to improve a process

The intentional structure of a software process

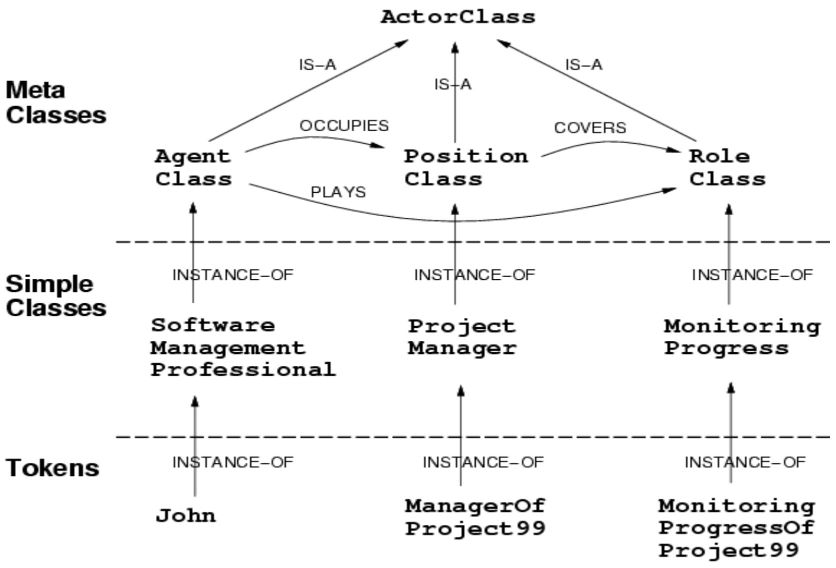
- What goals does an actor want others to achieve?
- What tasks does an actor want others to perform?
- What resources does an actor want others to furnish?
- The intentional structure of a software process is modelled as a network of intentional dependencies among actors – the actor dependency model.



The “whys” can be traced to motivations, goals, and interests of different participants/ stakeholders in the software process.



Roles, Agents, and Positions



Meta Classes

ActorClass

IS–A

IS–A

IS–A

Agent Class

OCCUPIES

Position Class

COVERS

Role Class

PLAYS

Simple Classes

INSTANCE–OF

INSTANCE–OF

INSTANCE–OF

Software Management Professional

Project Manager

Monitoring Progress

Tokens

INSTANCE–OF

INSTANCE–OF

INSTANCE–OF

John

ManagerOf Project99

Monitoring ProgressOf Project99

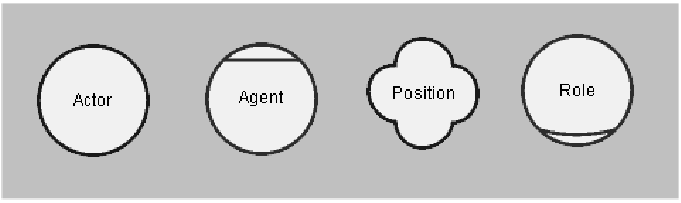
Agent, Role, Position

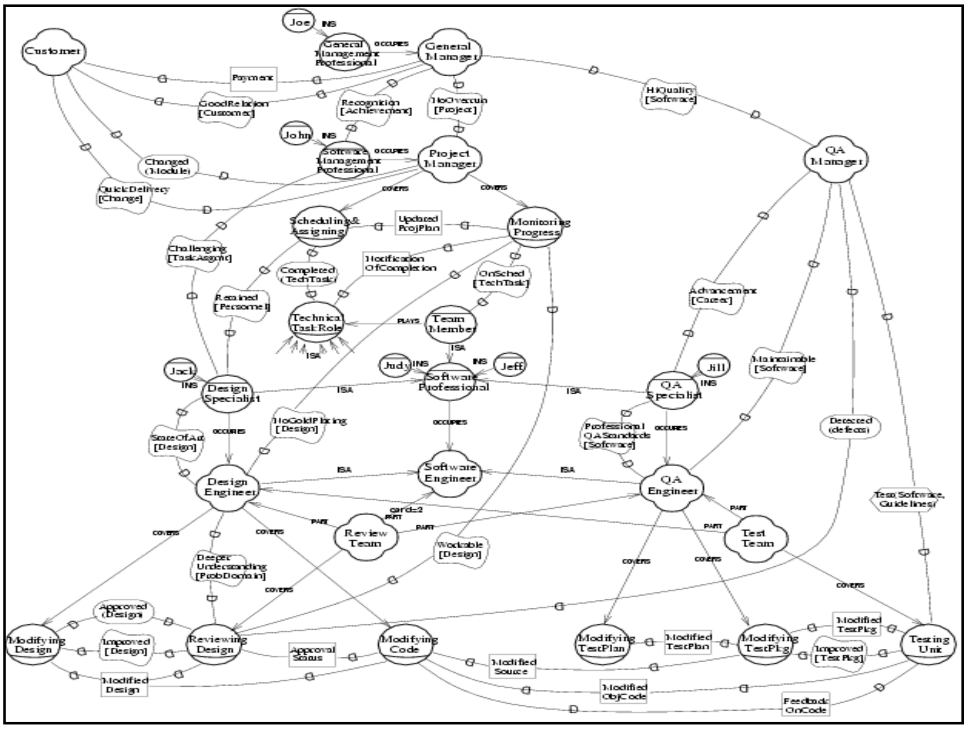
- We use the term actor to refer generally to any unit to which intentional dependencies can be ascribed
- To model the sub-units of a complex social actor, we define three types of sub-units – agent, roles, and positions – each of which is an actor in a more specialized sense
- Role is an abstract characterization of the behavior of a social actor within some specialized context or domain Its characteristics are easily transferable to other social actors. Dependencies are associated with a role when these dependencies apply regardless of who play the role

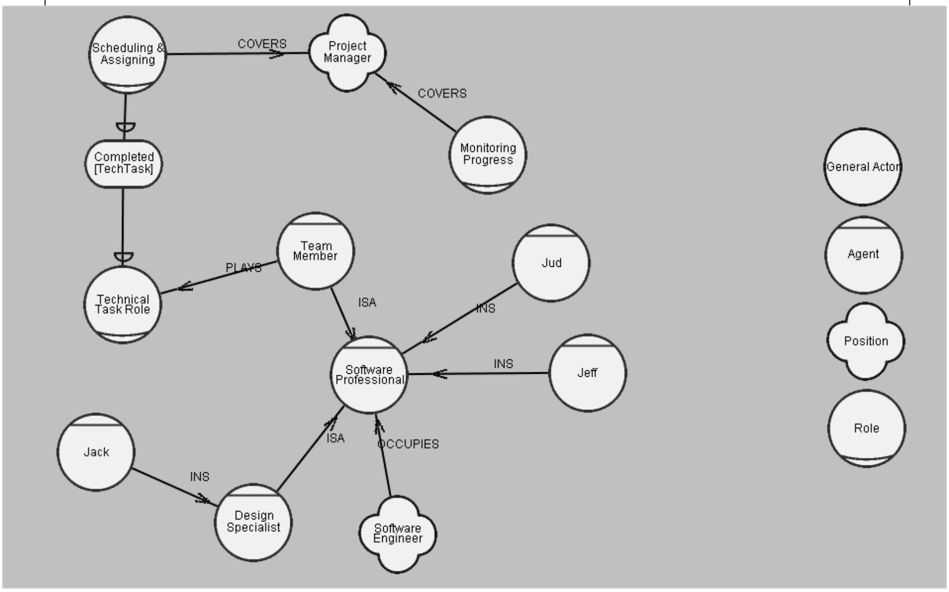
Agent, Role, Position

- An agent is an actor with concrete, physical manifestations, such as a human individual. We use the term agent instead of person for generality, so that it can be used to refer to human as well as artificial (hw/sw) agents. An agent has dependencies that apply regardless of what roles he/she/it happens to be playing. These characteristics are typically not easily transferable to other individuals, e.g., its skills and experiences, and its physically limitations.
- A position is intermediate in abstraction between a role and an agent. It is a set of roles typically played by one agent (e.g., assigned jointly to that one agent). We say that an agent occupies a position. A position is said to cover a role.

Graphical representation

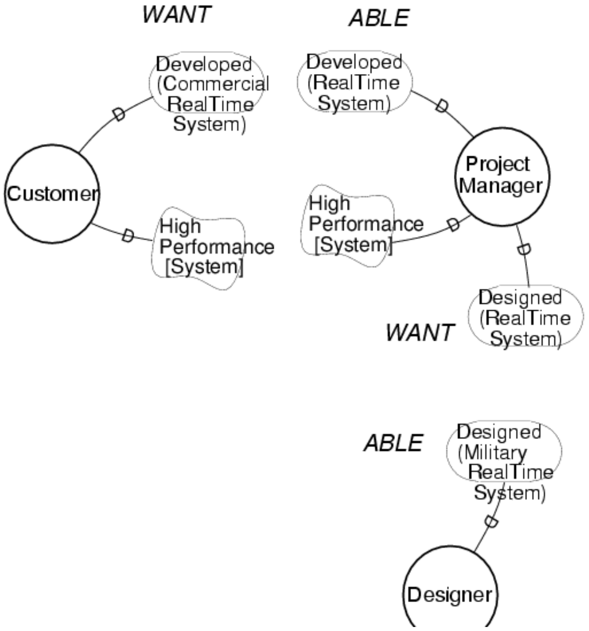



Joe
INS
General Management Professional
OCCUPIES
General Manager
Customer
Payment
GoodRelation [Customer]
Recognition [Achievement]
NoOverrun [Project]
HiQuality [Software]
John
INS
Software Management Professional
OCCUPIES
Project Manager
QA Manager
Changed (Module)
QuickDelivery [Change]
COVERS
COVERS
Scheduling& Assigning
Updated ProjPlan
Monitoring Progress
Challenging [TaskAssgmt]
Completed (TechTask)
Notification OfCompletion
OnSched [TechTask]
Advancement [Career]
Retained [Personnel]
Technical TaskRole
PLAYS
Team Member
ISA
ISA
Judy
INS
INS
Jeff
Jill
Maintainable [Software]
Jack
INS
Design Specialist
ISA
Software Professional
ISA
QA Specialist
INS
OCCUPIES
NoGoldPlating [Design]
OCCUPIES
Professional QA Standards [Software]
OCCUPIES
Detected (defects)
StateOfArt [Design]
ISA
Software Engineer
ISA
Design Engineer
QA Engineer
Test(Software, Guidelines)
card=2
PART
PART
PART
PART
PART
COVERS
COVERS
Review Team
Test Team
Workable [Design]
Deeper Understanding [ProbDomain]
COVERS
COVERS
COVERS
COVERS
Approved (Design)
Modifying Design
Improved [Design]
Reviewing Design
Approval Status
Modifying Code
Modifying TestPlan
Modified TestPlan
Modifying TestPkg
Modified TestPkg
Improved [TestPkg]
Testing Unit
Modified Design
Modified Source
Modified ObjCode
Feedback OnCode

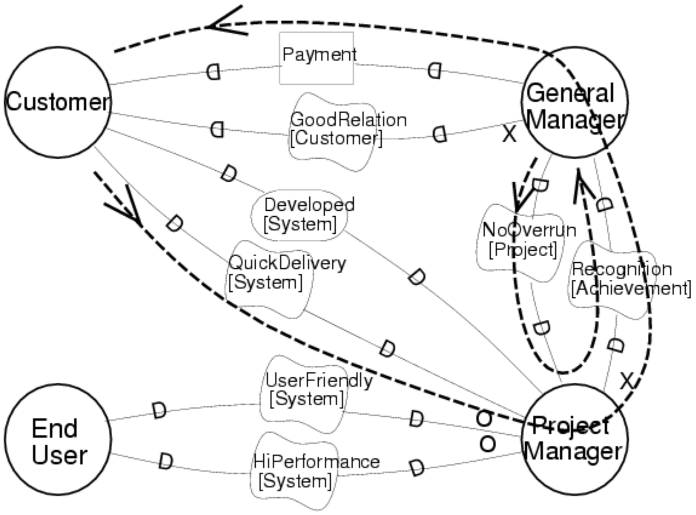

Scheduling & Assigning
COVERS
Project Manager
COVERS
Monitoring Progress
Completed [TechTask]
General Actor
Agent
Team Member
PLAYS
Technical Task Role
ISA
Jud
INS
Position
Software Professional
INS
Jeff
Role
Jack
INS
ISA
OCCUPIES
Design Specialist
Software Engineer

Analyzing opportunities

Matching Wants and Abilities


WANT
ABLE
Developed (Commercial RealTime System)
Developed (RealTime System)
Customer
Project Manager
High Performance [System]
High Performance [System]
Designed (RealTime System)
WANT
ABLE
Designed (Military RealTime System)
Designer

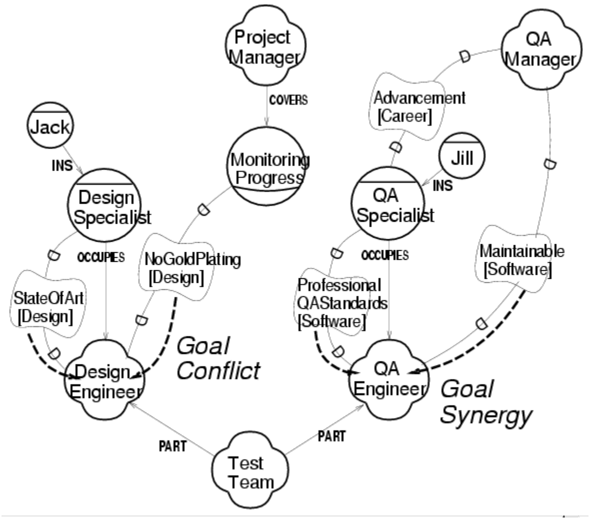
Analyzing vulnerabilities



Reciprocal dependency

Loop analysis

Analyzing vulnerabilities



Goal synergy or conflict

Exercise

- Let’s try to model the 3 actors Customer, Bank and House-vendor when the customer want to buy a new house from the House-vendor and has to ask money to the Bank.