NOVALINCS
LABORATORY FOR COMPUTER SCIENCE AND INFORMATICS

LETS Laboratório de especificação e teste de software

FCT FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Exploring Views for Goal-oriented Requirements Comprehension

Lyrene Silva[1], Ana Moreira[2], João Araújo[2], **Catarina Gralha[2]**, Miguel Goulão[2] and Vasco Amaral[2]

[1]Universidade Federal do Rio Grande do Norte (Brazil),
[2]Universidade NOVA de Lisboa (Portugal)

November 14, 2016

There is a diversity of options in the software development process

need to use

based on

Artifacts

- Diagrams
- Code
- Text

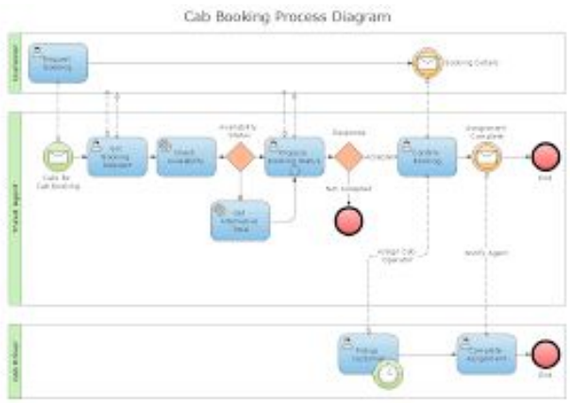
Activities

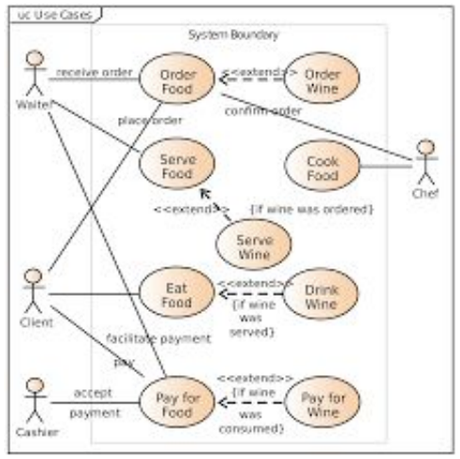
- Management
- Modelling
- Analysis
- Tests
- Maintenance

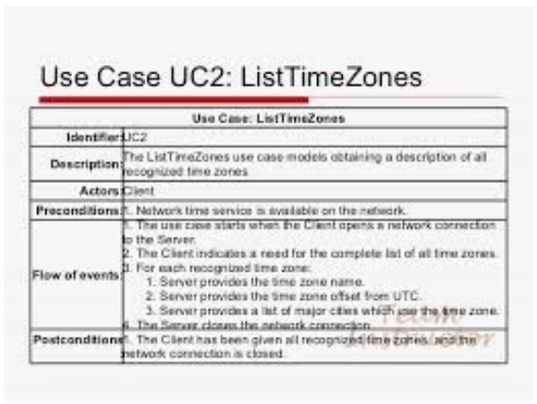
People profile

- Programmer
- Architect
- Analyst
- Tester

SAMPLE Project Requirements Document – Library Blog






Use Case UC2: ListTimeZones

Use Case: ListTimeZones	
Identifier	UC2
Description	The ListTimeZones use case models obtaining a description of all recognized time zones
Actors	Client
Preconditions	1. Network time service is available on the network.
Flow of events	1. The use case starts when the Client opens a network connection to the Server. 2. The Client indicates a need for the complete list of all time zones. 3. For each recognized time zone: 1. Server provides the time zone name. 2. Server provides the time zone offset from UTC. 3. Server provides a list of major cities which use the time zone. 4. The Server closes the network connection.
Postconditions	1. The Client has been given all recognized time zones and the network connection is closed.

But what is the problem?

Requirements artifacts are often too **large** or too **complex** to be quickly understood or queried by different stakeholders

Multiple views are broadly employed in requirements modeling for very specific purposes

These views do not offer interactive features to allow stakeholders browsing the information according to their needs

Views of the requirements should be configured to their particular needs

Requirements exploration

It is a process to navigate through requirements artifacts, aiming at comprehending their structure and content

Reasons to provide mechanisms for requirements exploration:

1) Requirements artifacts are often used by people who have not created them
2) Stakeholders need to search information on these artifacts
3) Exploration mechanisms can help navigating through the entire documentation to find the elements associated with a specific point of interest

Viewpoint, point-of-view, view and visualization

Viewpoint and point-of-view
opinion that a person has about a given subject

View
includes a representation language

Visualization
includes a representation mechanism such as zoom, filter, details on demand, history log, overview, …

Proposing 3 views for requirements exploration

1

Big Picture view

Generates an overview for a source model

Offers the ability to expand and reduce the details on demand

Organizes the model information on levels of importance or by aggregation

Proposing 3 views for requirements exploration

1 **Big Picture view**

Generates an overview for a source model

Offers the ability to expand and reduce the details on demand

Organizes the model information on levels of importance or by aggregation

2 **Syntax-based view**

Filters the types of language elements that will be visualized

Proposing 3 views for requirements exploration

1 **Big Picture view**

Generates an overview for a source model

Offers the ability to expand and reduce the details on demand

Organizes the model information on levels of importance or by aggregation

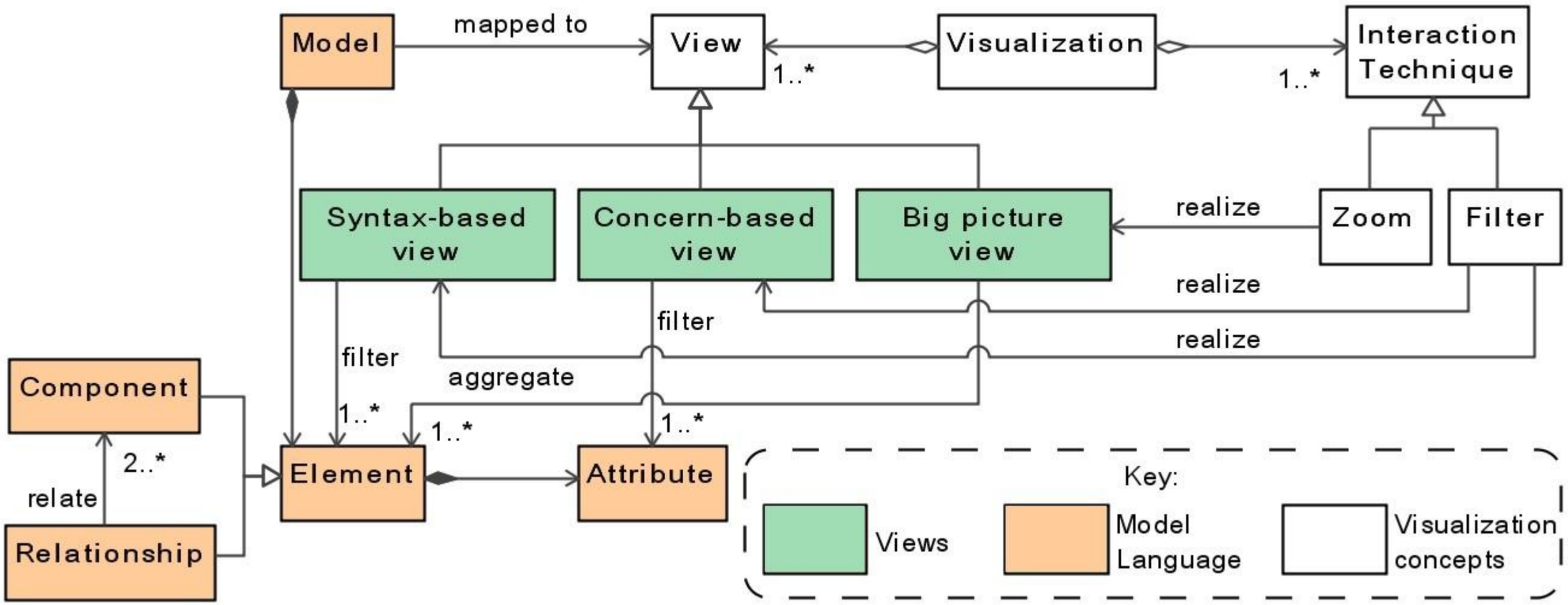
2 **Syntax-based view**

Filters the types of language elements that will be visualized

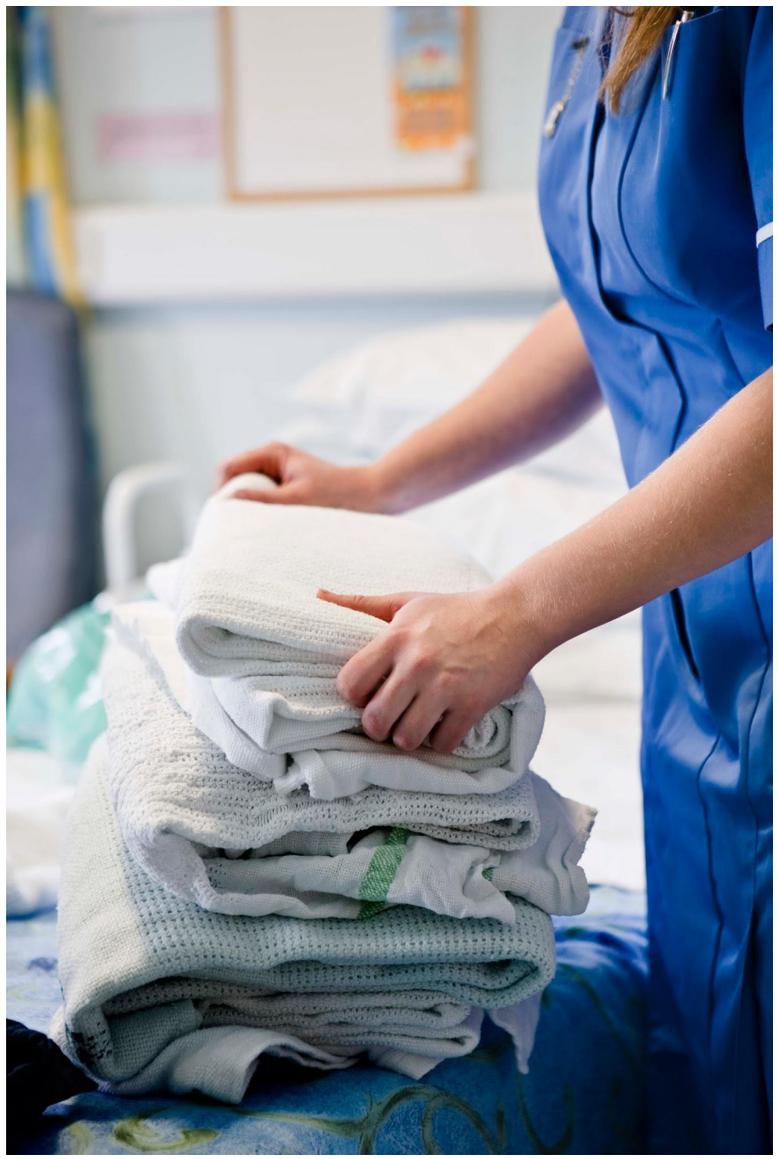
3 **Concern-based view**

Filters concerns through meta-data, system lexicon (key words) or semantic similarity

Conceptual model: exploratory views of requirements



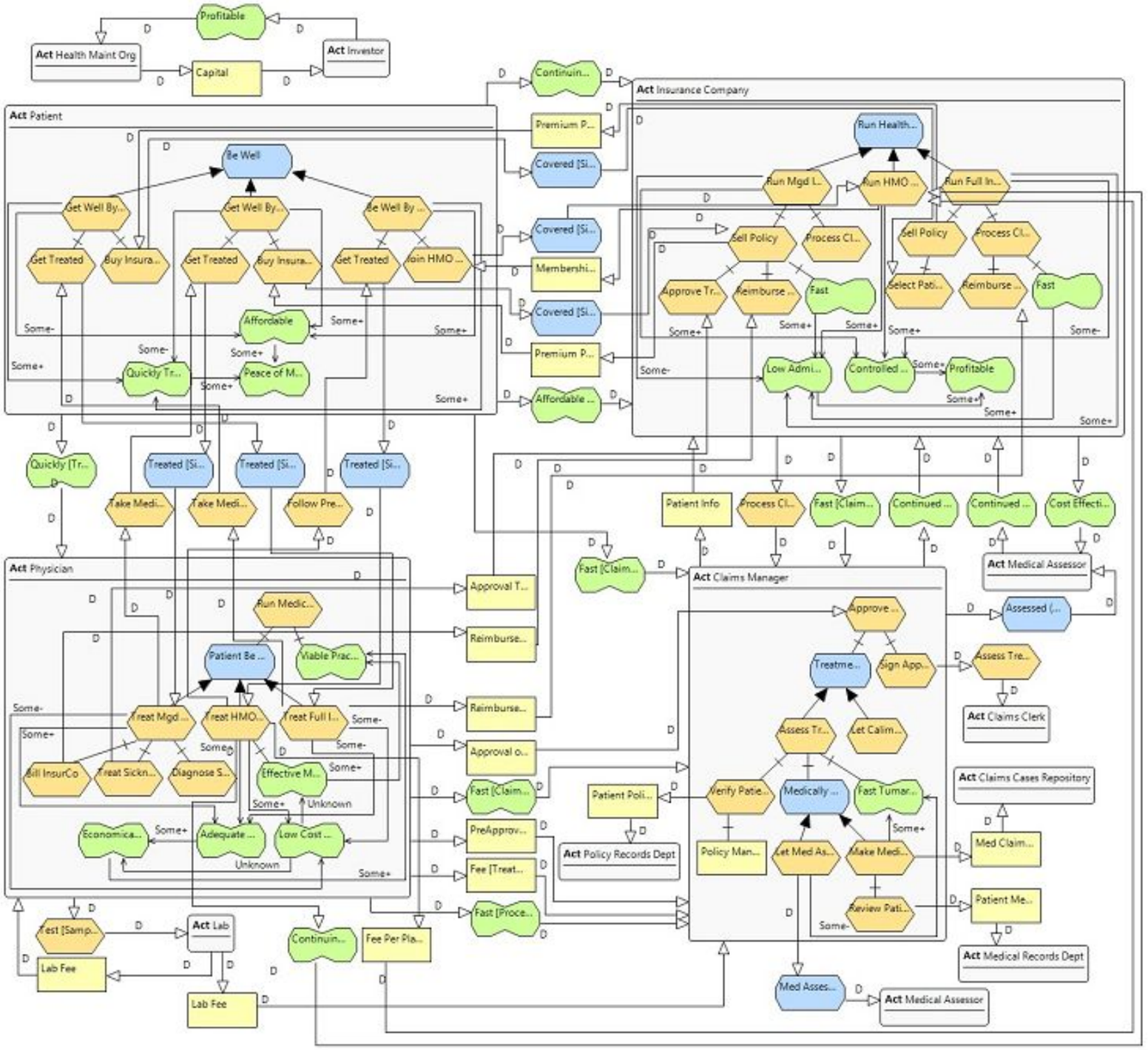
Demonstration: Health Care System



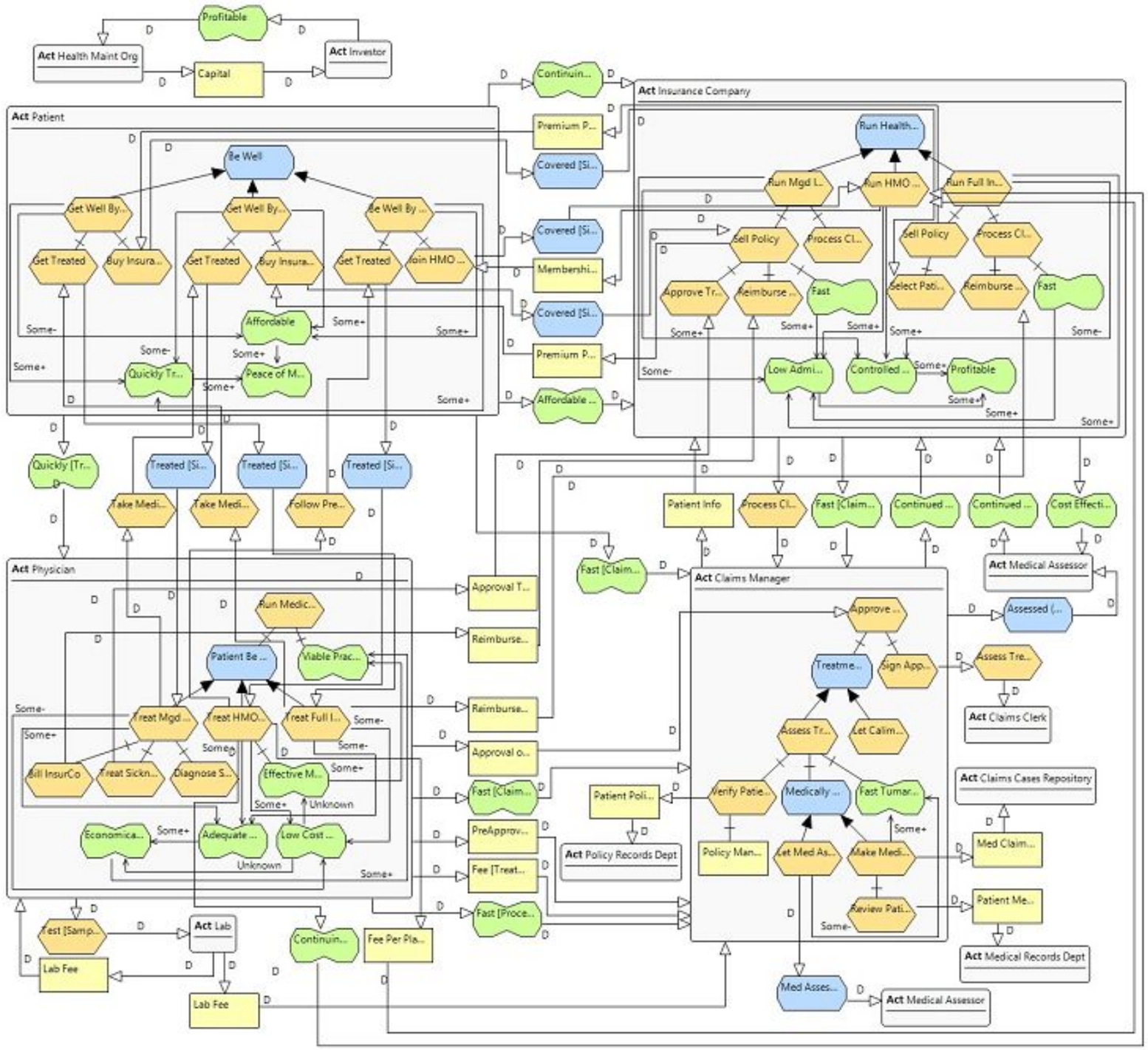
Demonstration: *i** model for Health Care

Numbers for Health Care System (HCS):

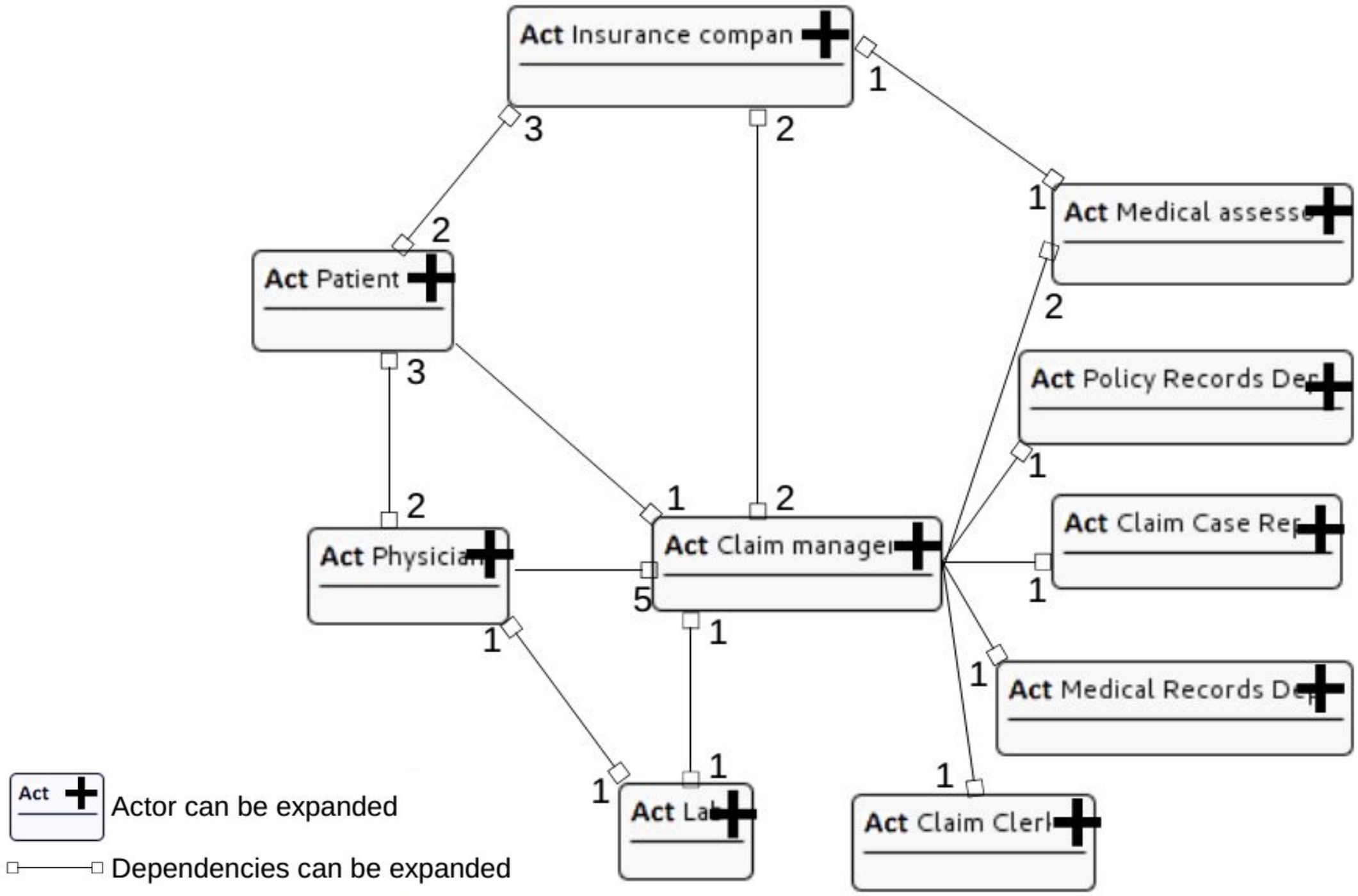
- **13 actors**
- **13 goals**
- **41 tasks**
- **26 softgoals**
- **165 links**



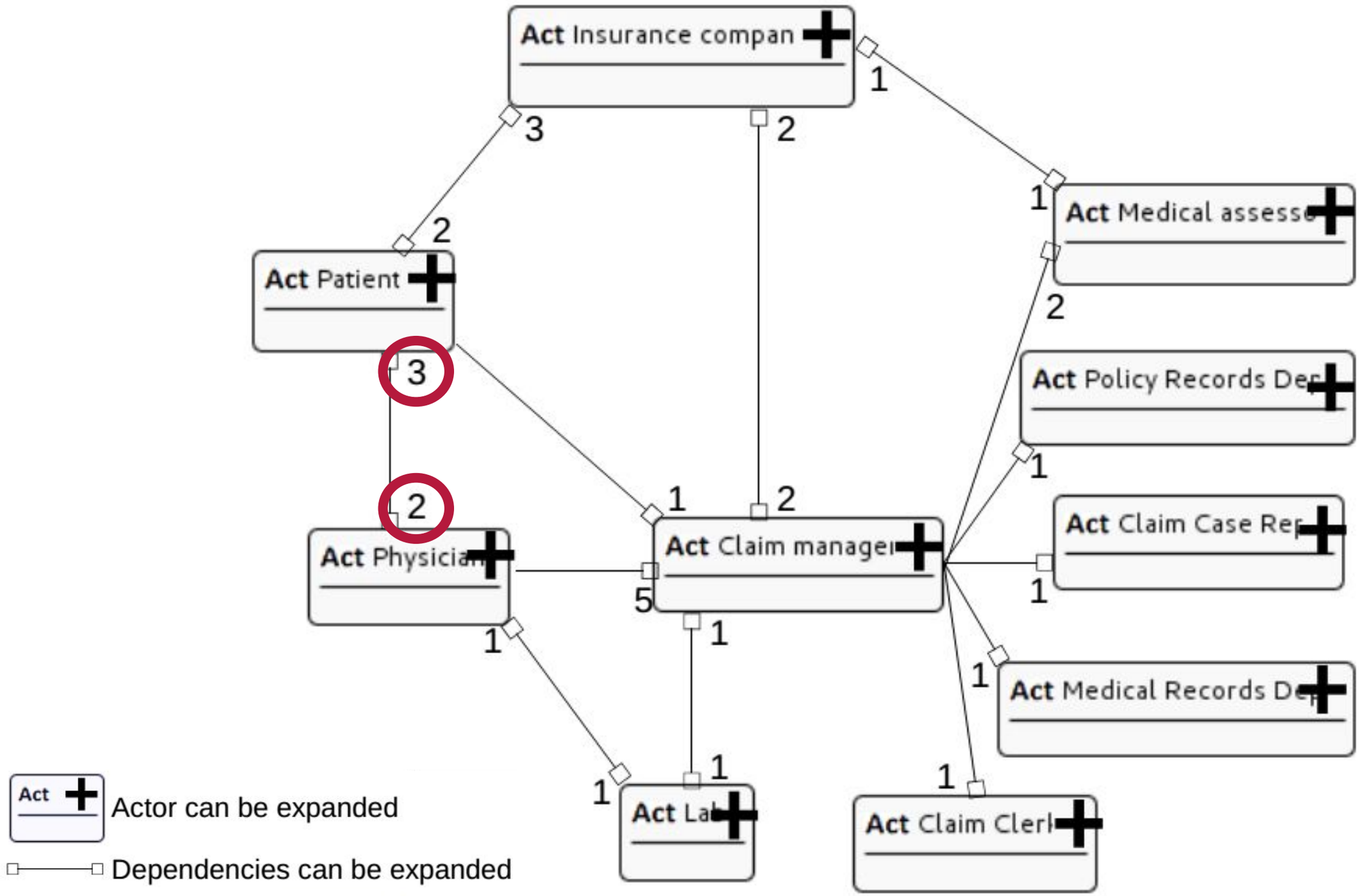
Big Picture view expanded



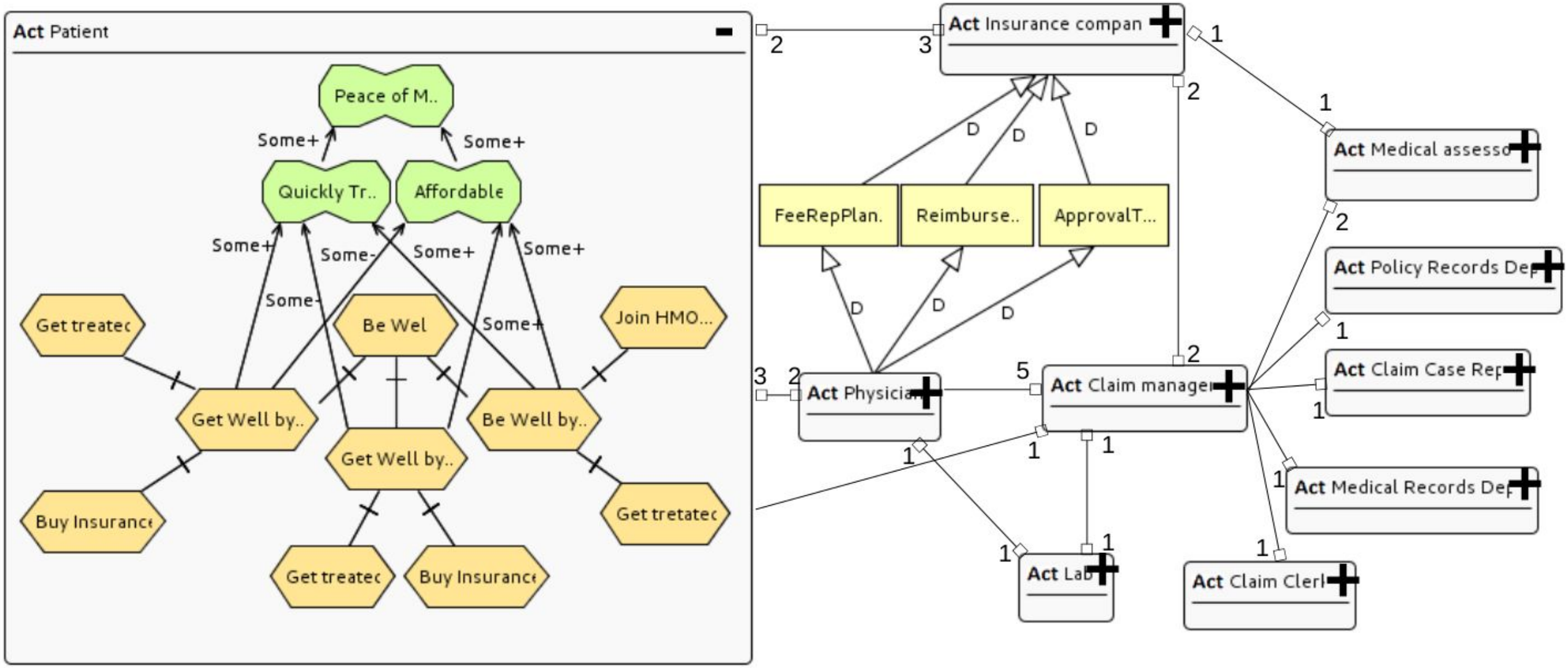
Big Picture view with both actors and dependencies collapsed



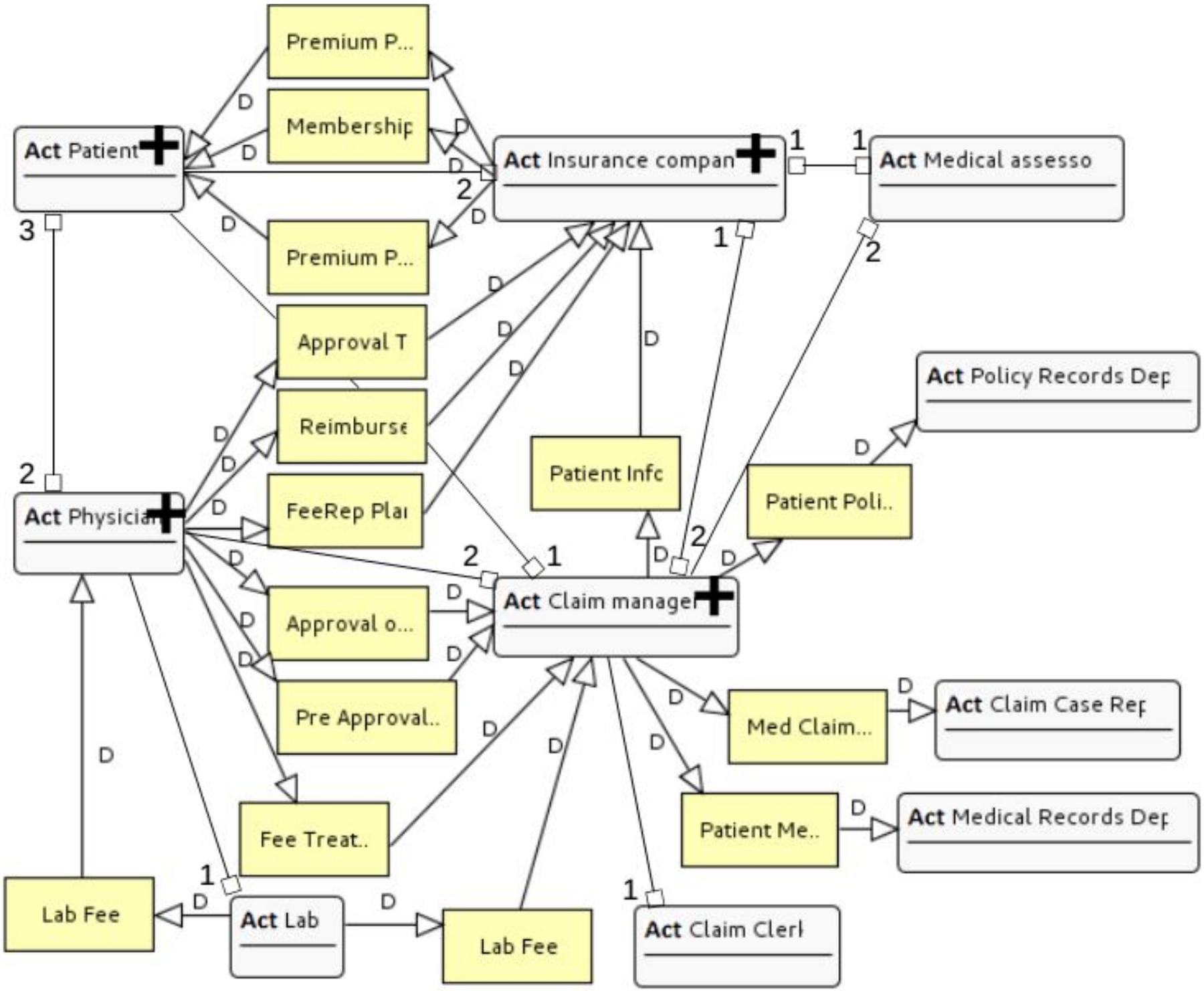
Big Picture view with both actors and dependencies collapsed



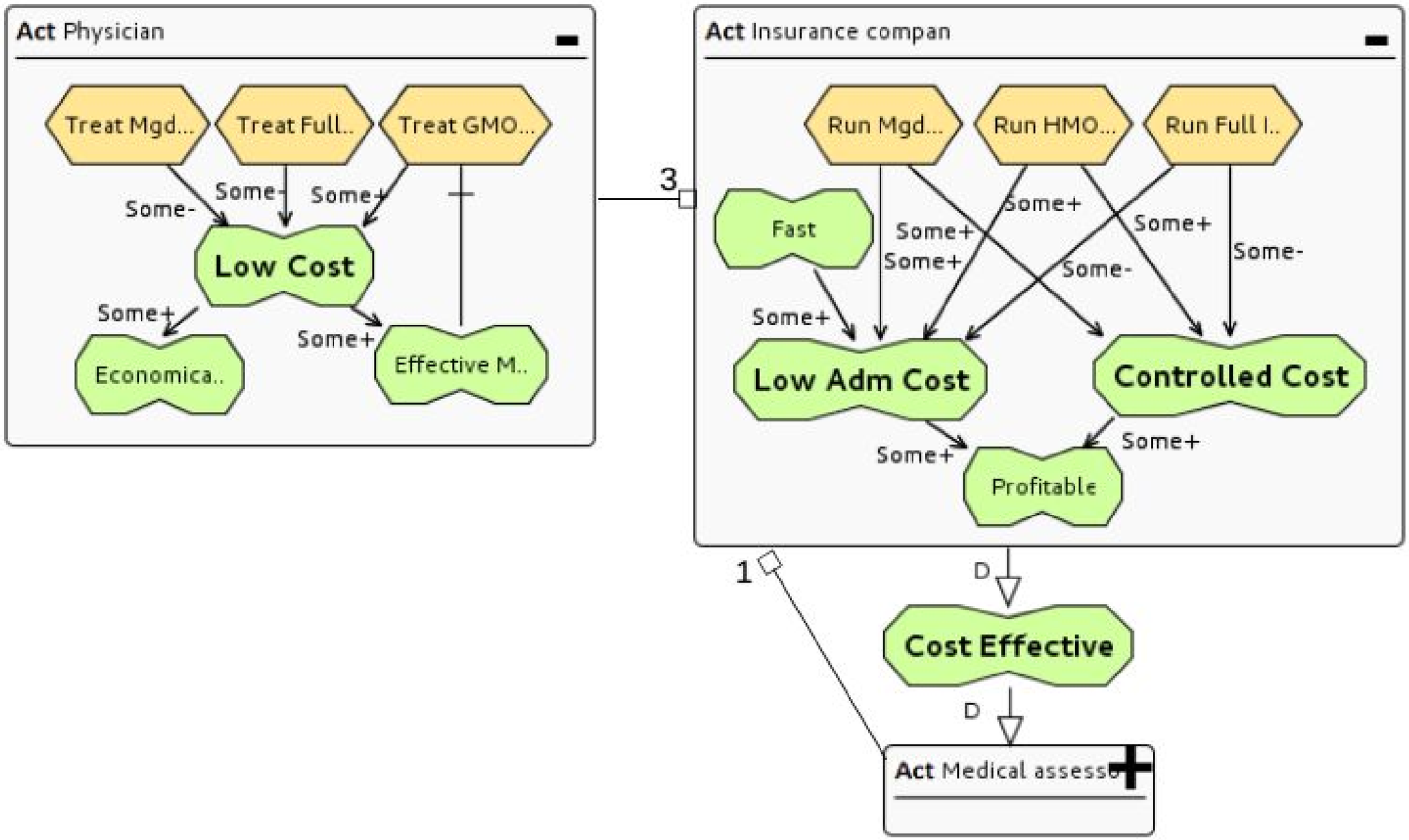
Big Picture view with an expanded actor and link



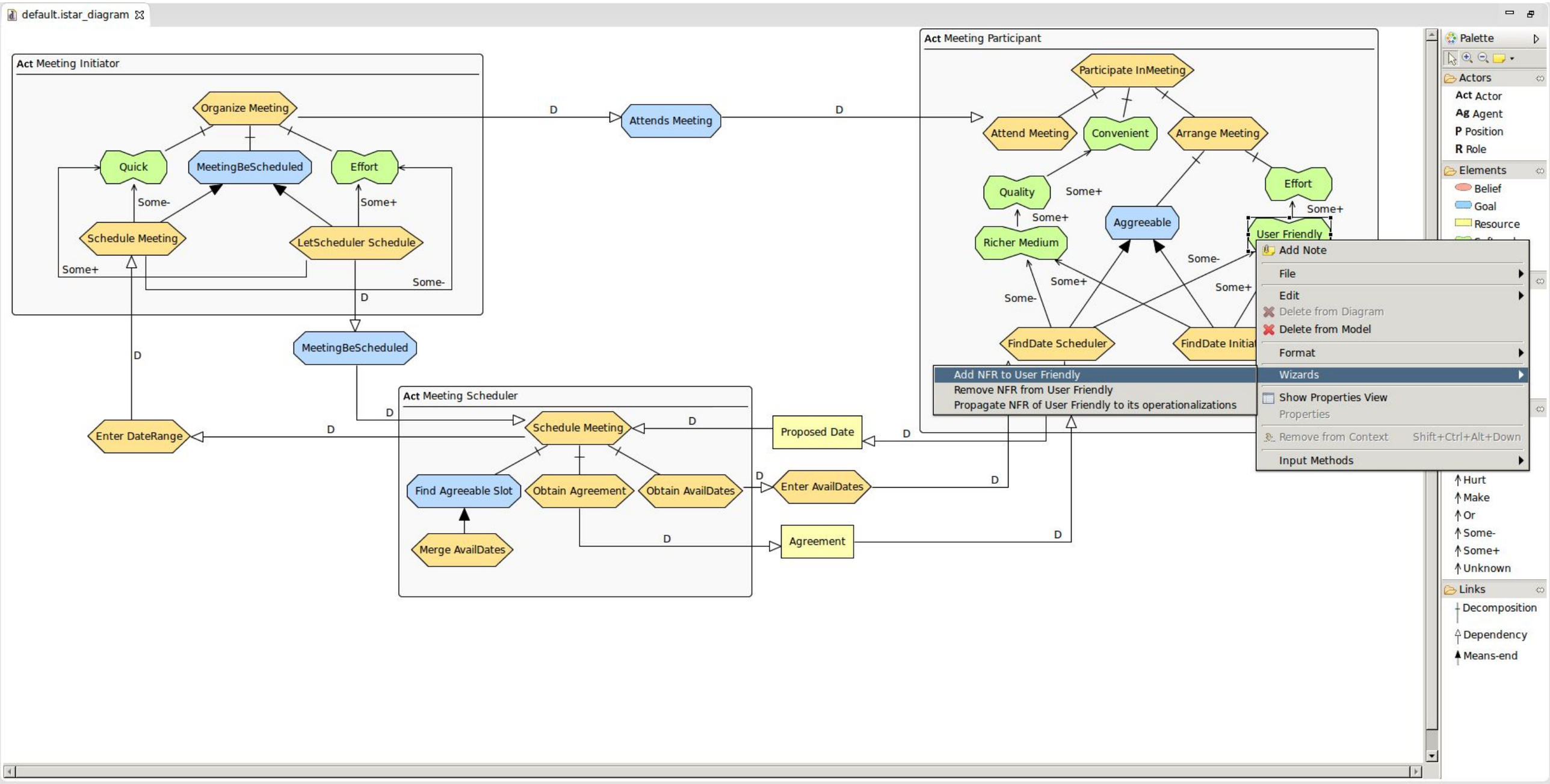
Syntax-based view: filtering resource dependencies



Concern-based view: filtering the string *Cost*, considering a distance of one



Tool support for the syntax-based view



Discussion and current challenges for using multiple view for requirements exploration

In the requirements engineering process, many models are generated.
It is necessary to provide **exploration mechanisms** to navigate **through them**, instead of only navigating an isolated model

The tools implementing this **complementary views** should provide mechanisms to allow users to interact directly with the visual elements

Users may interact with the source model as well as with the views.
It is necessary to generate a view about **the path followed** to the achieved result

Interaction mechanisms include aspects from human-computer interaction have not been taken into consideration yet

Summary and conclusions

We proposed 3 views for requirements exploration:
big picture, syntax-based and concern-based

They are based on the interaction tasks zoom and filter

They capture 3 manners of abstracting a model, by:

1) decreasing it's amount of elements
2) allowing stakeholders to search
3) focusing on information of interest

Can be applied to other kinds of models: we have done it for use cases

Our views deal with the complexity of requirements models

Without this kind of mechanisms, more stakeholders effort is demanded to find and analise relevant information in the system model

Future work

Investigate how tools can be prepared for supporting our views

Define a process to instantiate our views to other requirement models

Conduct experimental evaluations of the impact of introducing the proposed views in requirements tools

Evaluate how the proposed views impact on the efficiency and effectiveness of different stakeholders while performing requirements exploration

Thank you

Questions?

NOVALINCS
LABORATORY FOR COMPUTER
SCIENCE AND INFORMATICS