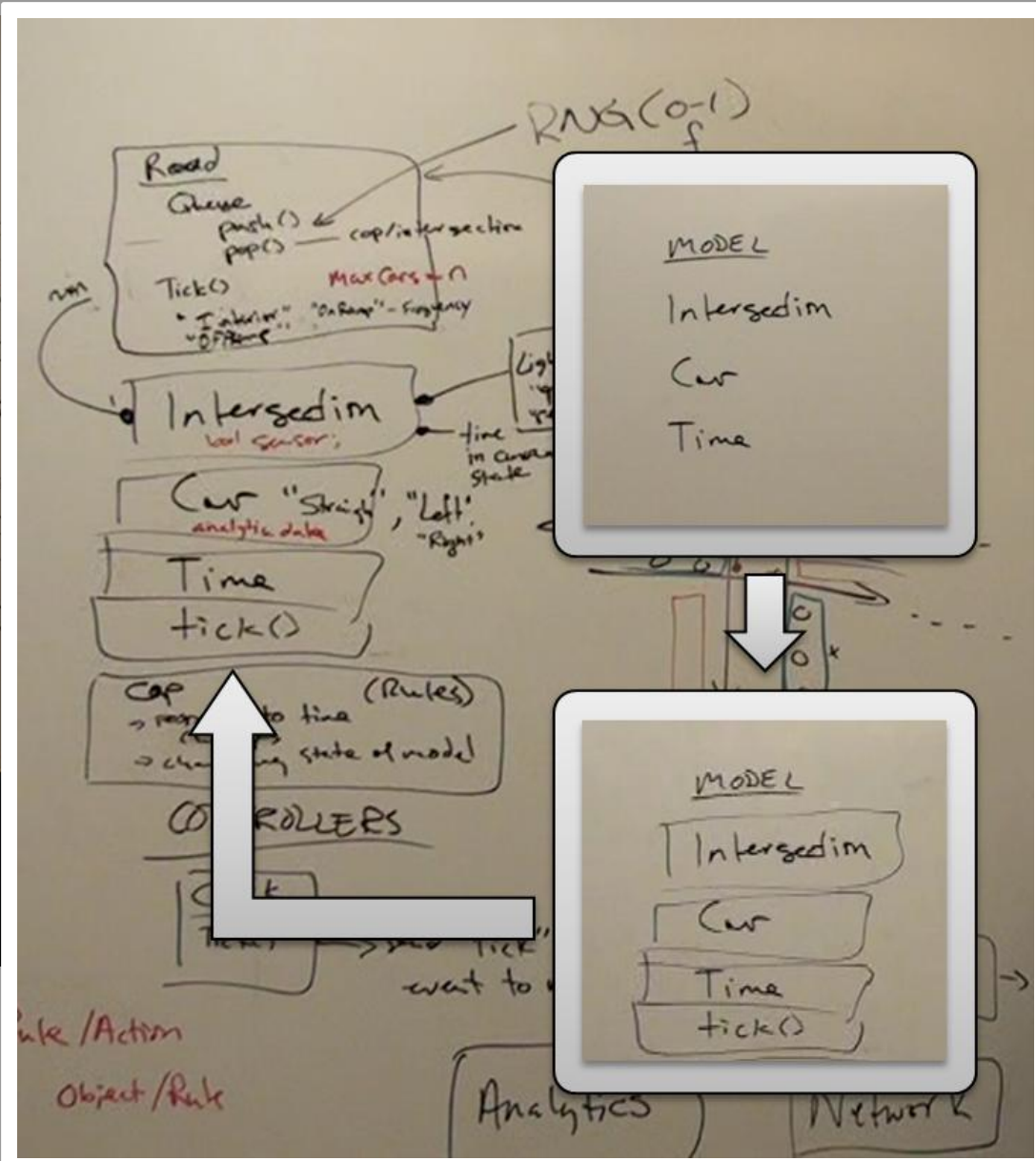
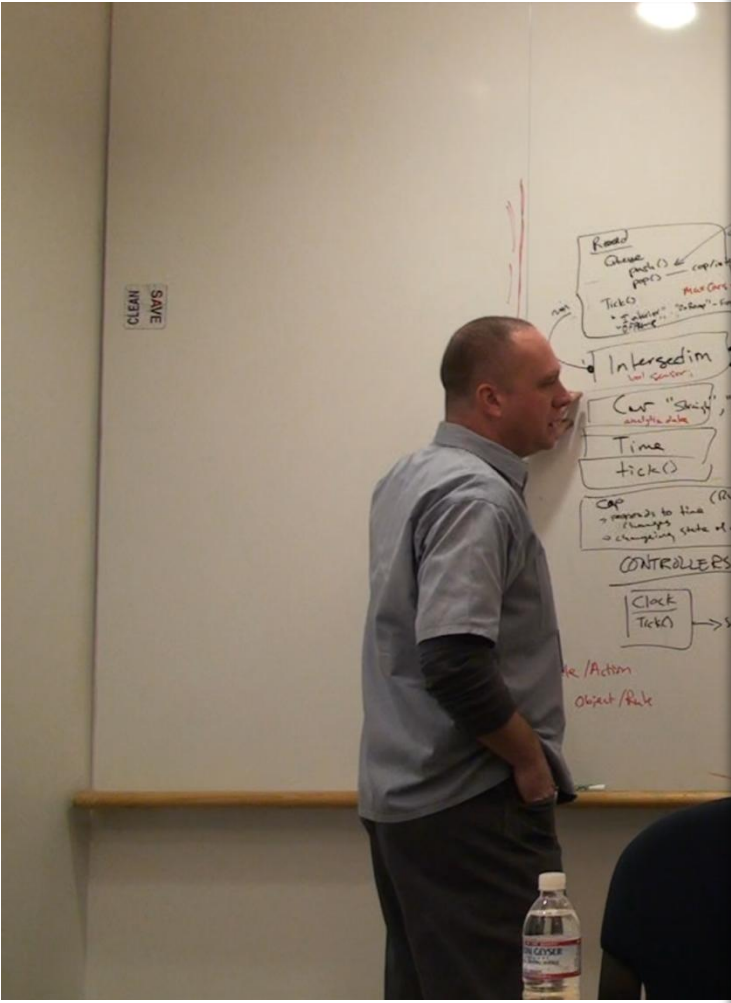


Lightweight Analysis of Software Design Models at the Whiteboard

Alfredo Motta, Nicolas Mangano, André van der Hoek

Software Design at the Whiteboard



Software Design at the Whiteboard



- Design sketches at the whiteboard
 - start as rough, informal sketches
 - refined into formal design notations
- In each incremental step, the engineers
 - inspect the design
 - identify and solve potential issues

Research Challenge



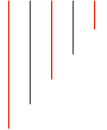
- Help software developers at the whiteboard by providing
 - early automated feedback about their design
 - without interfering with their design session
- Fill the gap between
 - a whiteboard
 - and a formal analysis tool

Questions That Must Be Addressed

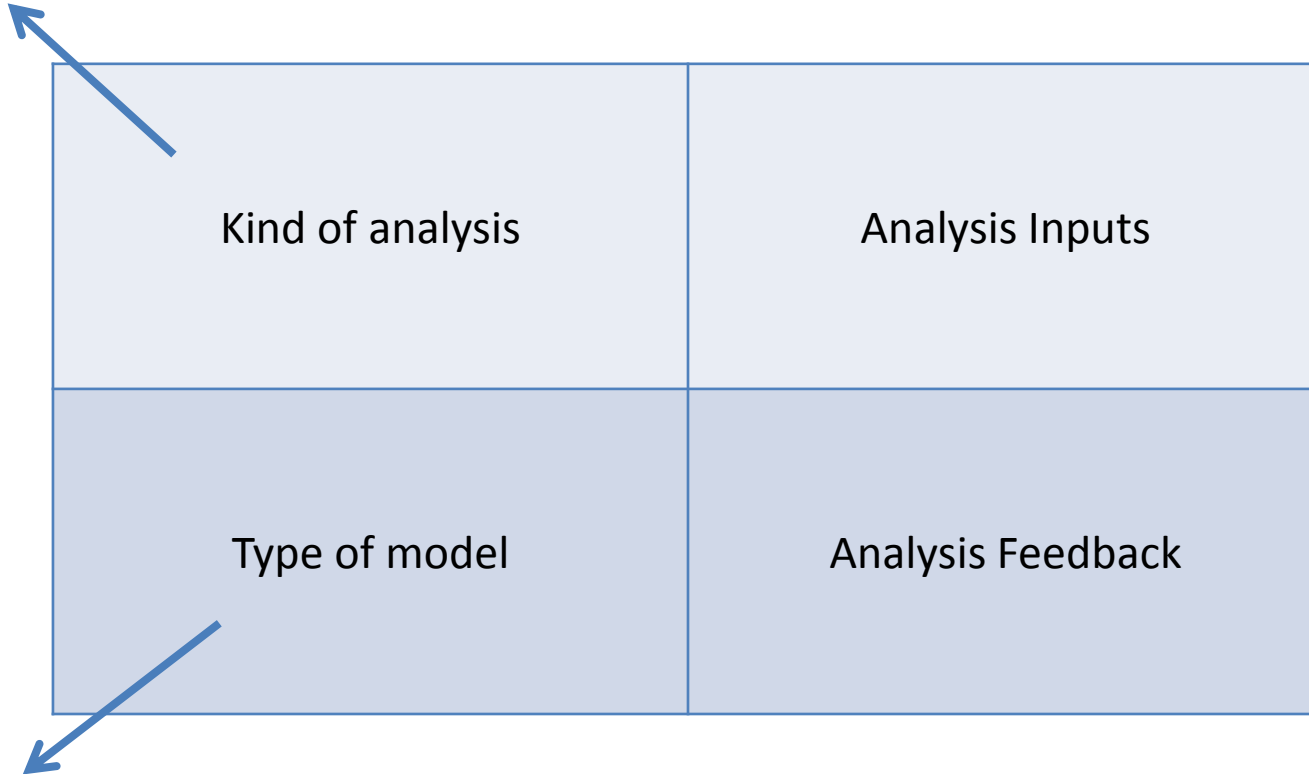


- How to build analyzable models with minimal input from the developer?
- How to present feedback to the developer in a non-intrusive fashion?

Approach



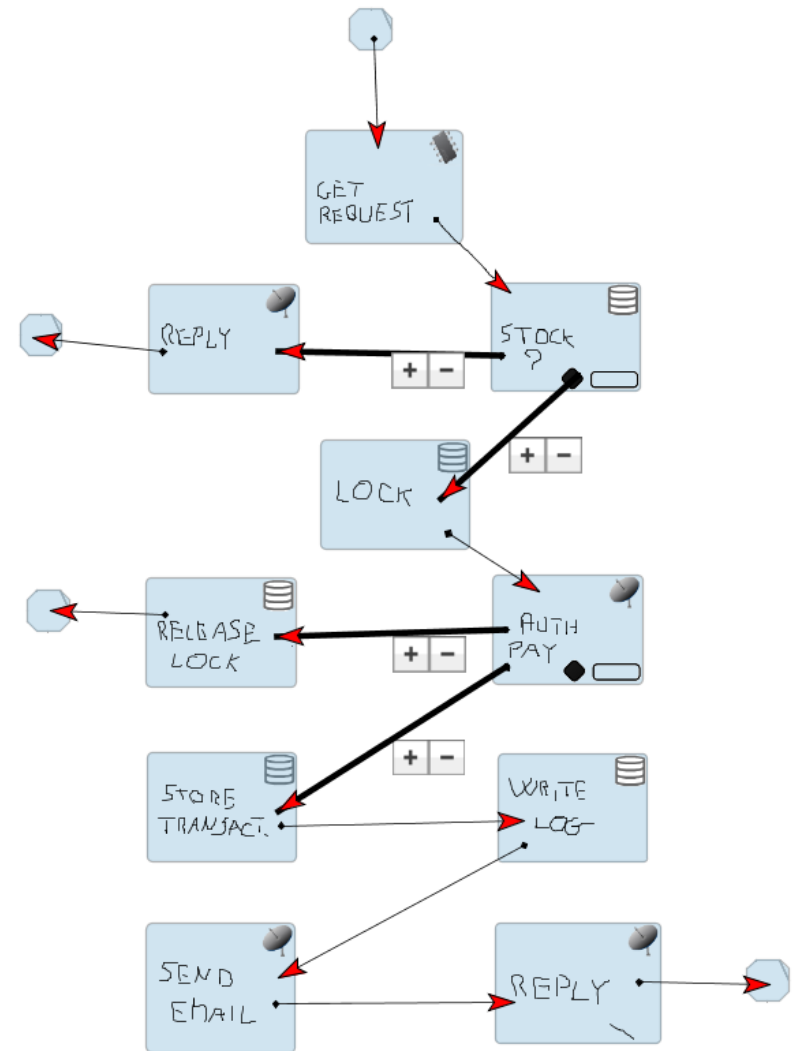
Performance



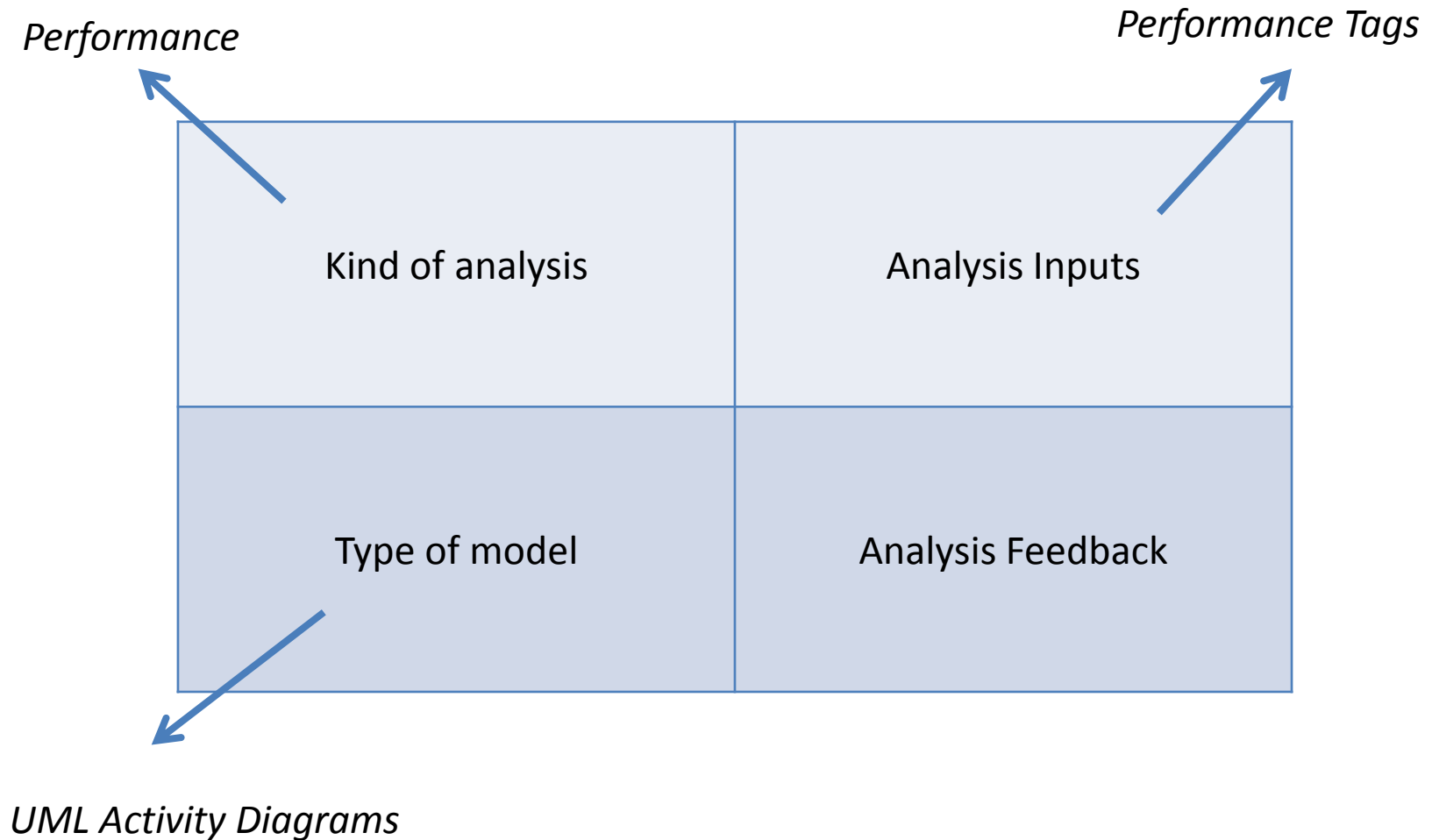
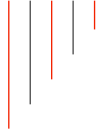
UML Activity Diagrams

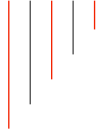
E-Commerce Example

- Typical operations to buy a product
 - a buy request is received by the server
 - the server checks if the item is available
 - the payment is processed
 - a reply is sent to the user



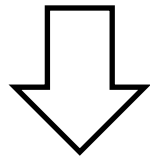
Approach





Performance Tags

- Tags are used to specify the expected execution time of the activity node
- A pre-built taxonomy is available



CPU (0x)



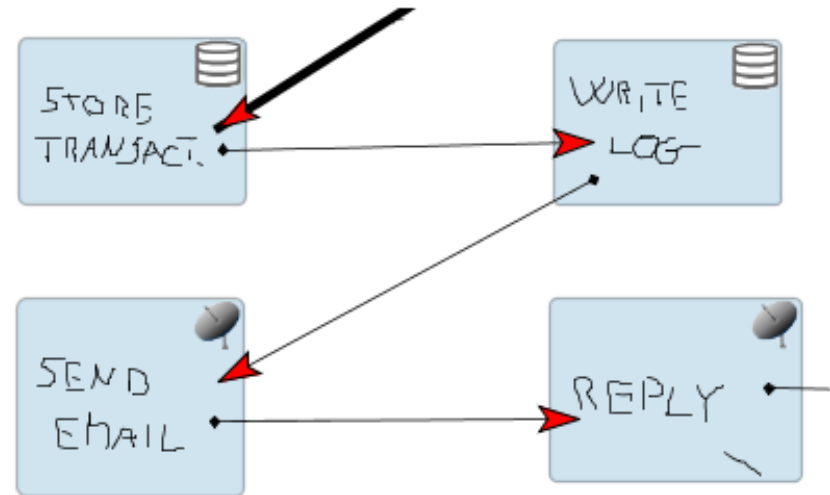
DB (5x)



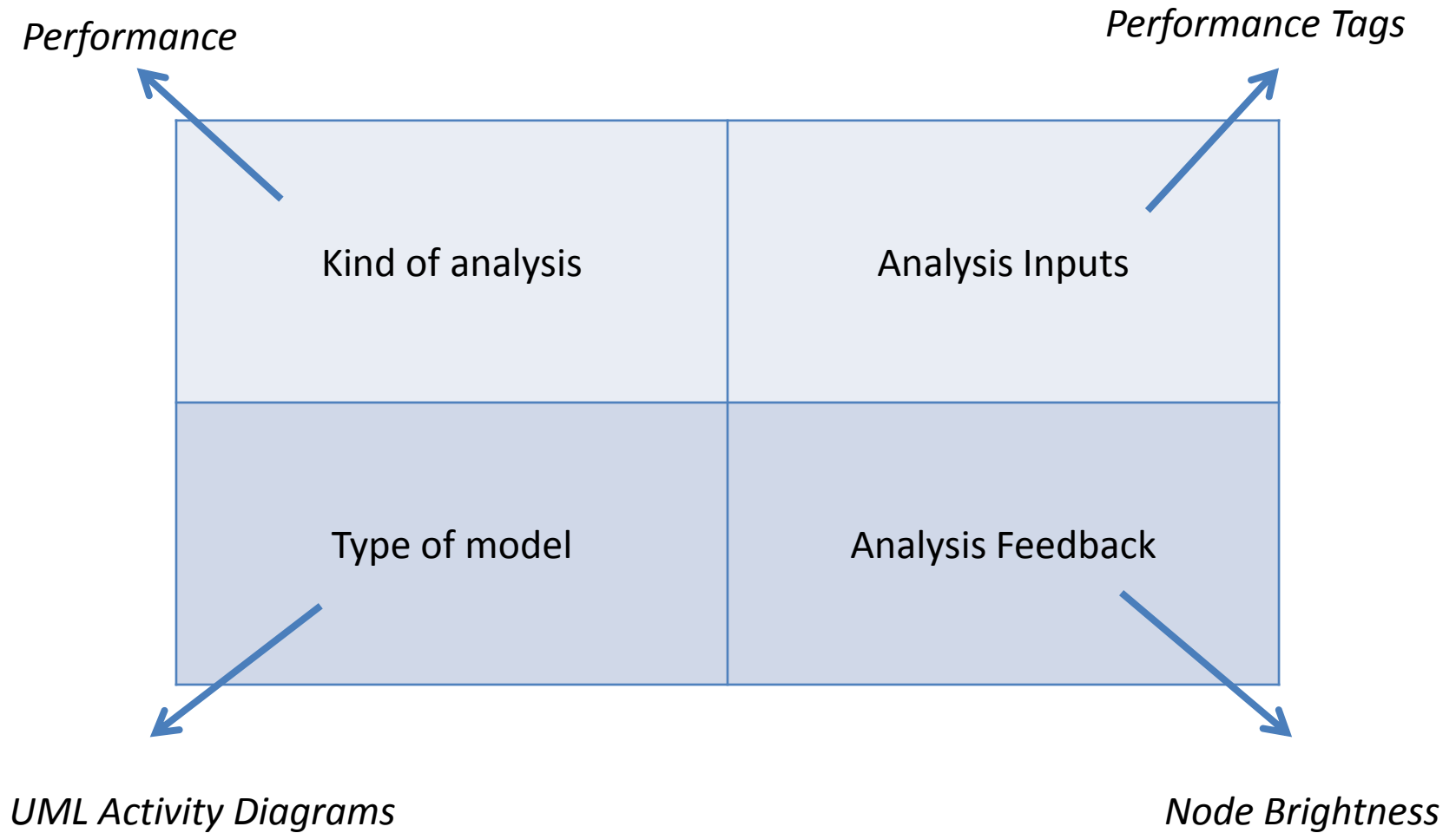
RAM (2x)



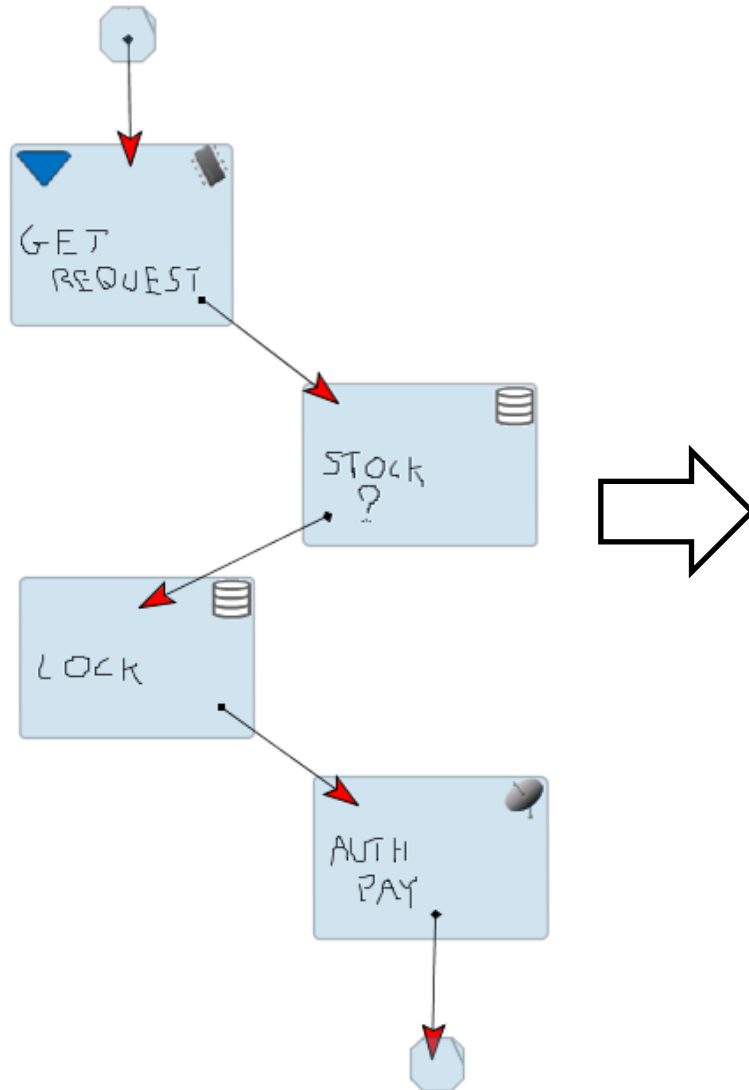
NET (8x)



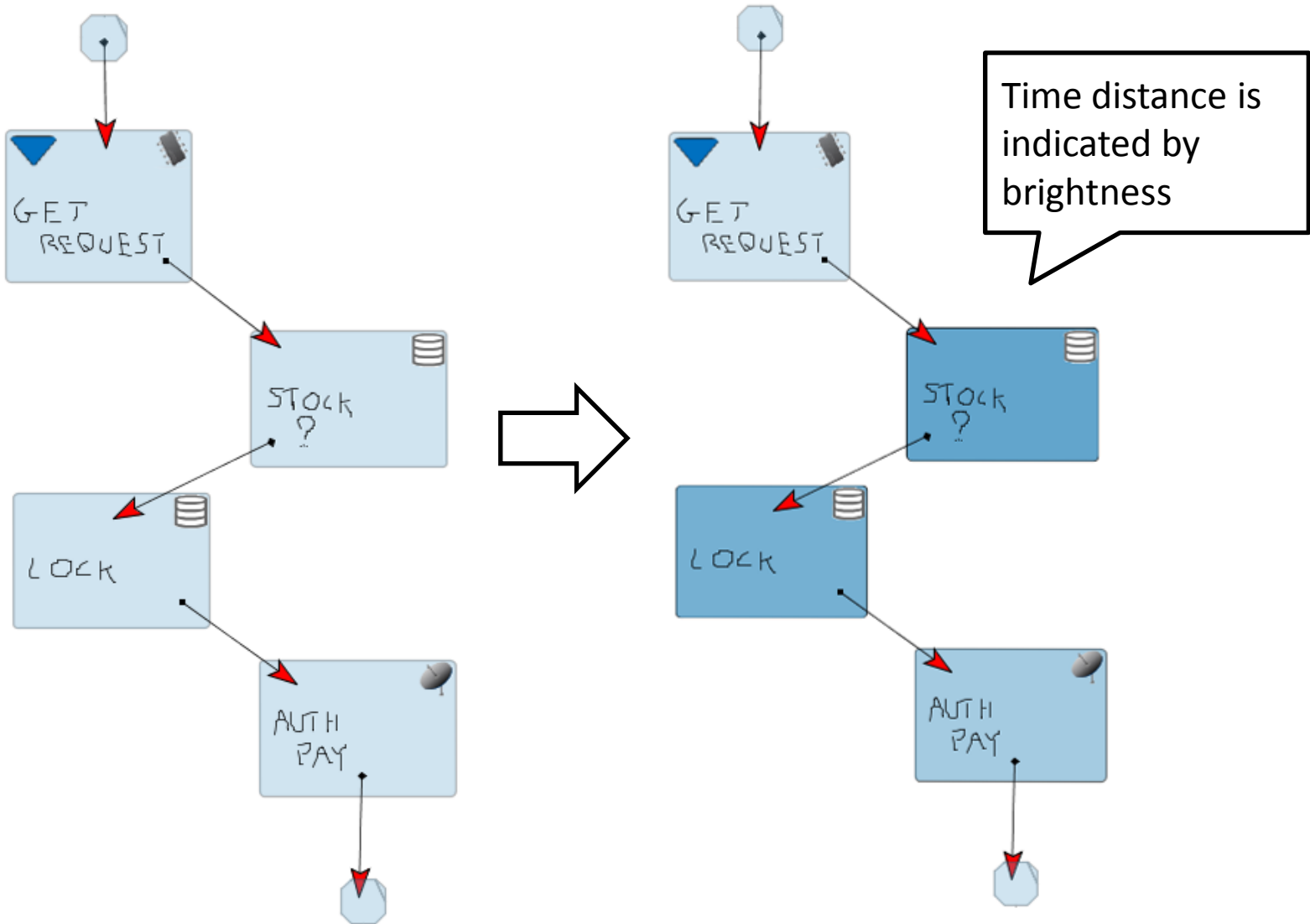
Approach



Feedback Example



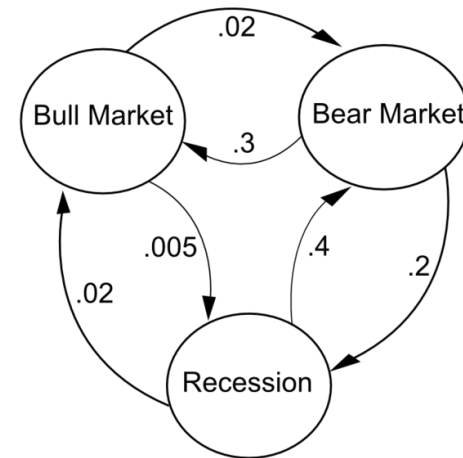
Feedback Example

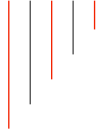


Time distance is indicated by brightness

Prism Model

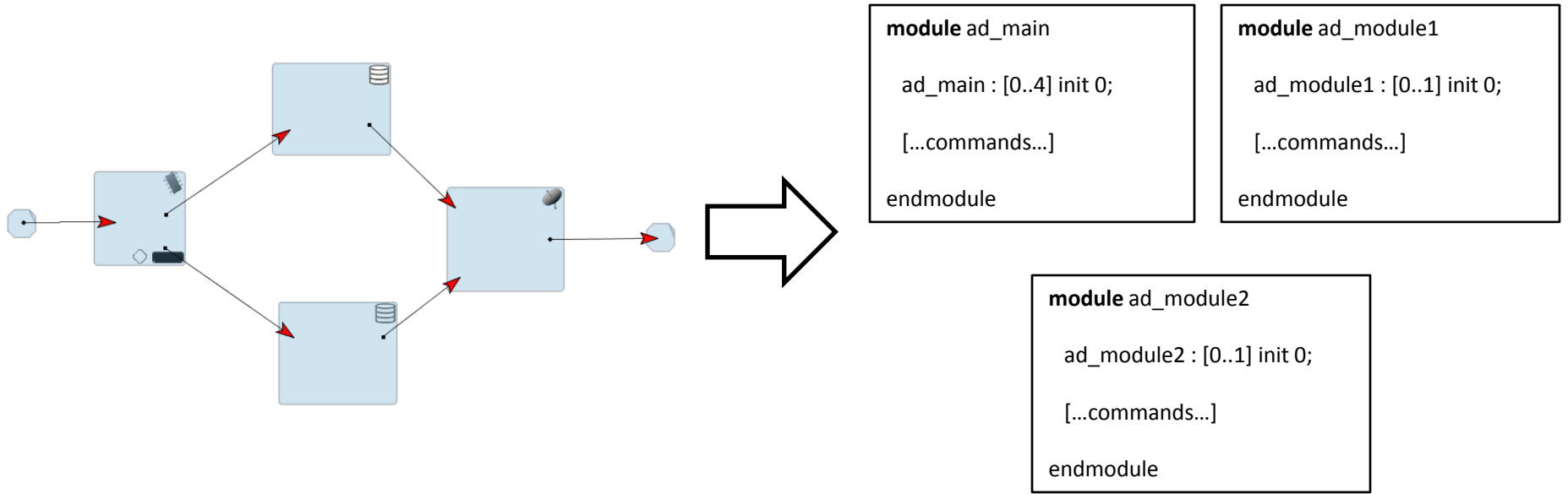
- The design is translated into the input language of the Prism Model Checker, a probabilistic model checker
- In turn, Prism uses a Continuous Time Markov Chain to solve the model

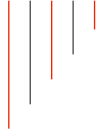




Prism Model

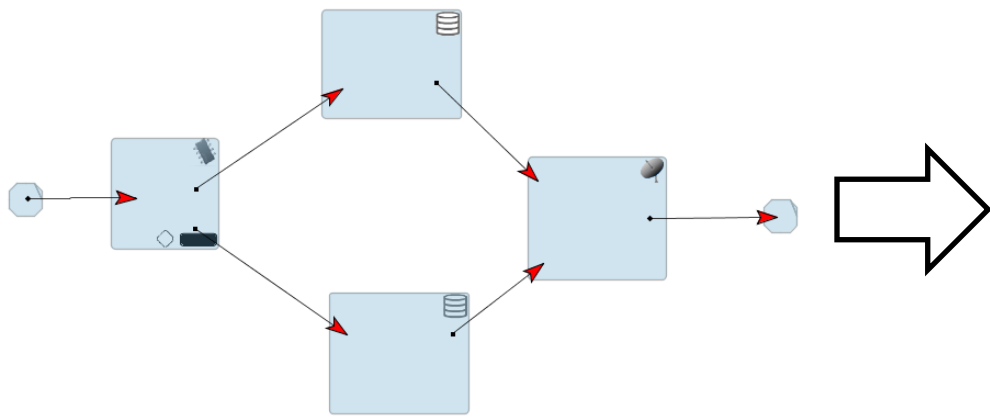
- The Activity Diagram is translated into a set of Prism modules
 - the main module, and a set of modules corresponding to all the fork/join path in the diagram





Prism Model

- Each Prism Module has
 - **States:** corresponding to the different activity nodes
 - **Commands:** referring to the connectors
 - **Execution rates for commands:** corresponding to the tags applied to the activity nodes



```
module ad_main

  ad_main : [0..4] init 0;

  [...commands...]

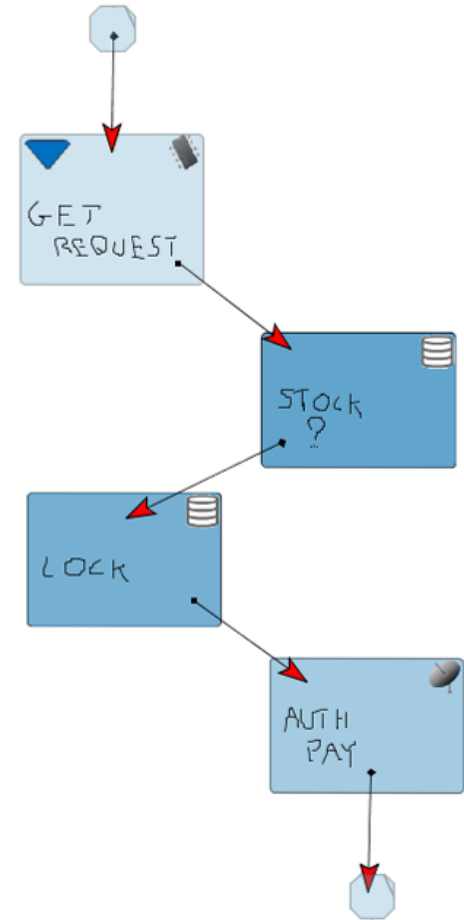
  [] ad_main = 3
    ->
    1/NET_VALUE : (ad_main' = 4);

  [...commands...]

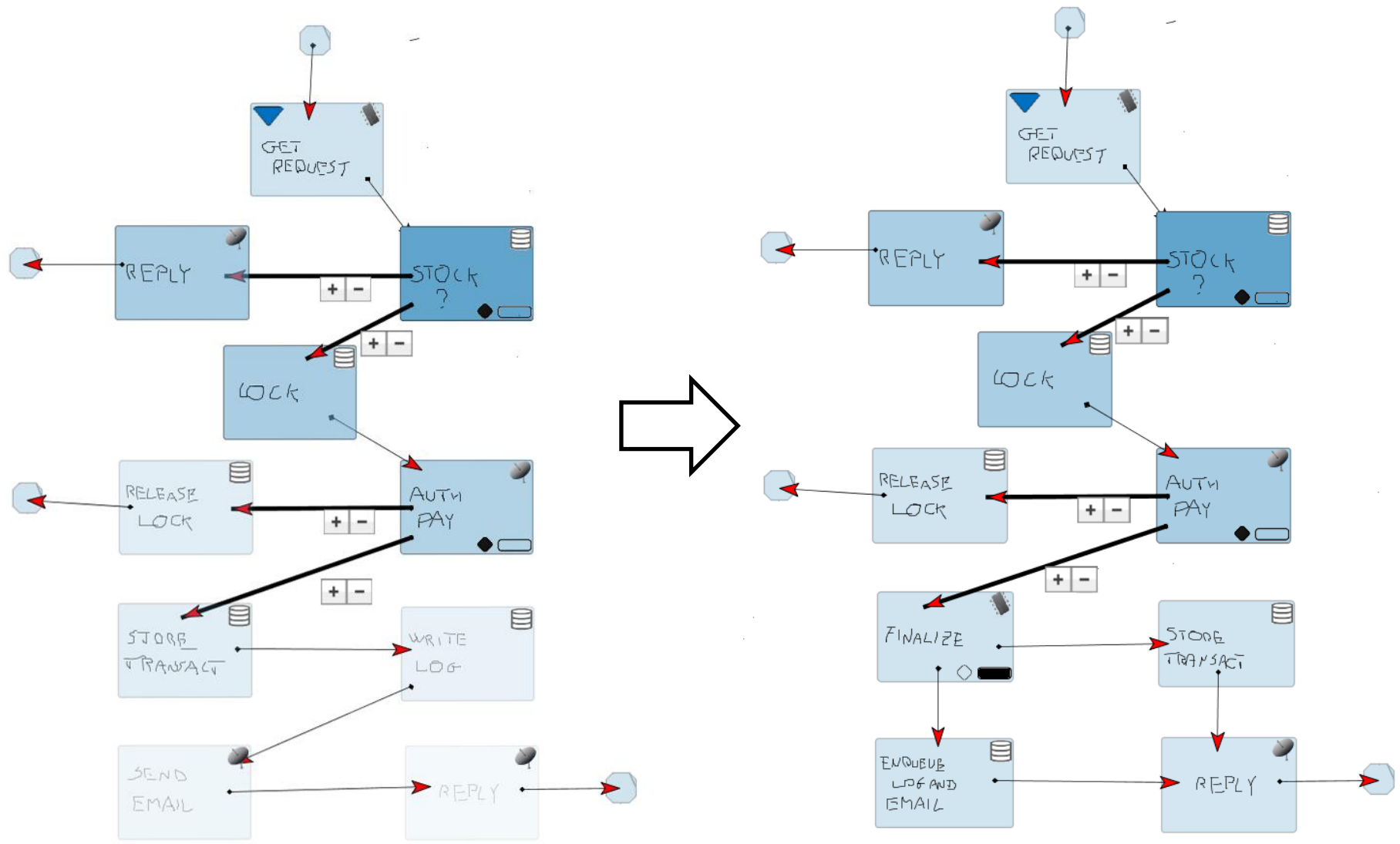
endmodule
```

Prism Property

- Given i , the node with the analysis tag
- For all the activity nodes j
- Compute probability of reaching node j starting from node i within t time units



Example: Designing with Feedback





- The tool offers the opportunity to
 - use models during the early stages
 - obtain immediate feedback about the design
- Potentially this could lead to
 - improved software designs because of feedback
 - more widespread exploitation of software models at the whiteboard

Thank you!