UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR SOFTWARE ENGINEERING
AND PROGRAMMING LANGUAGES

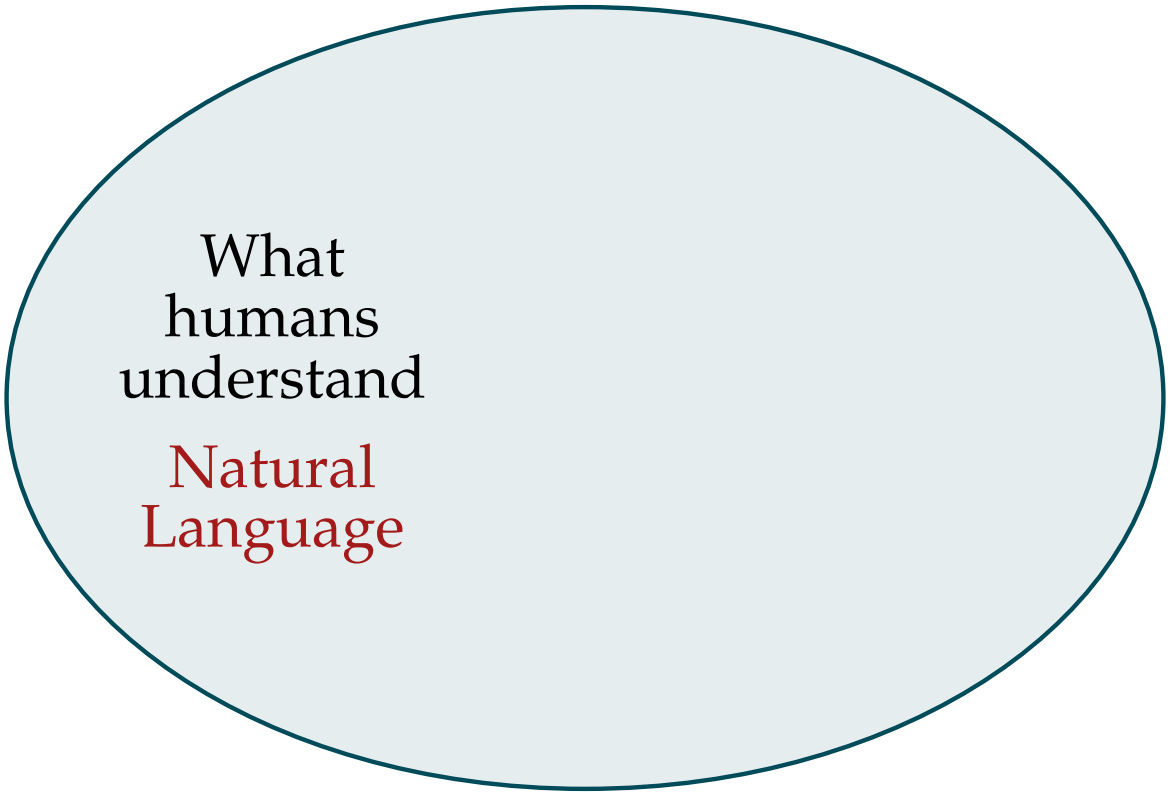
HØGSKOLEN I BERGEN
BERGEN UNIVERSITY COLLEGE

Integration of Runtime Verification into Metamodeling for Simulation and Code Generation (Position Paper)

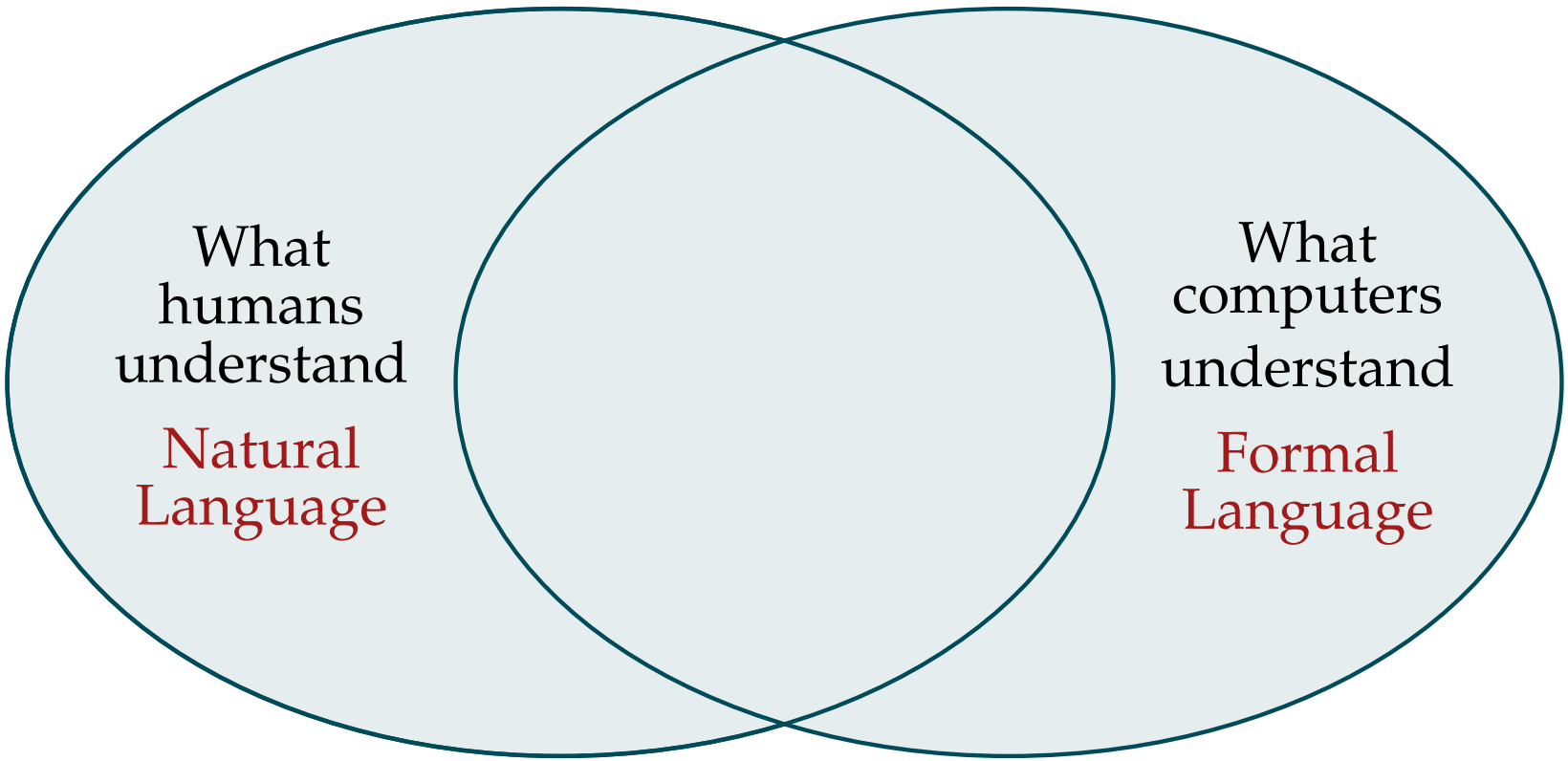
F. Macias T. Scheffel M. Schmitz R. Wang
{fernando.macias,rui.wang}@hib.no
{scheffel,schmitz}@isp.uni-luebeck.de

16th International Conference on Runtime Verification, September 2016, Madrid, Spain

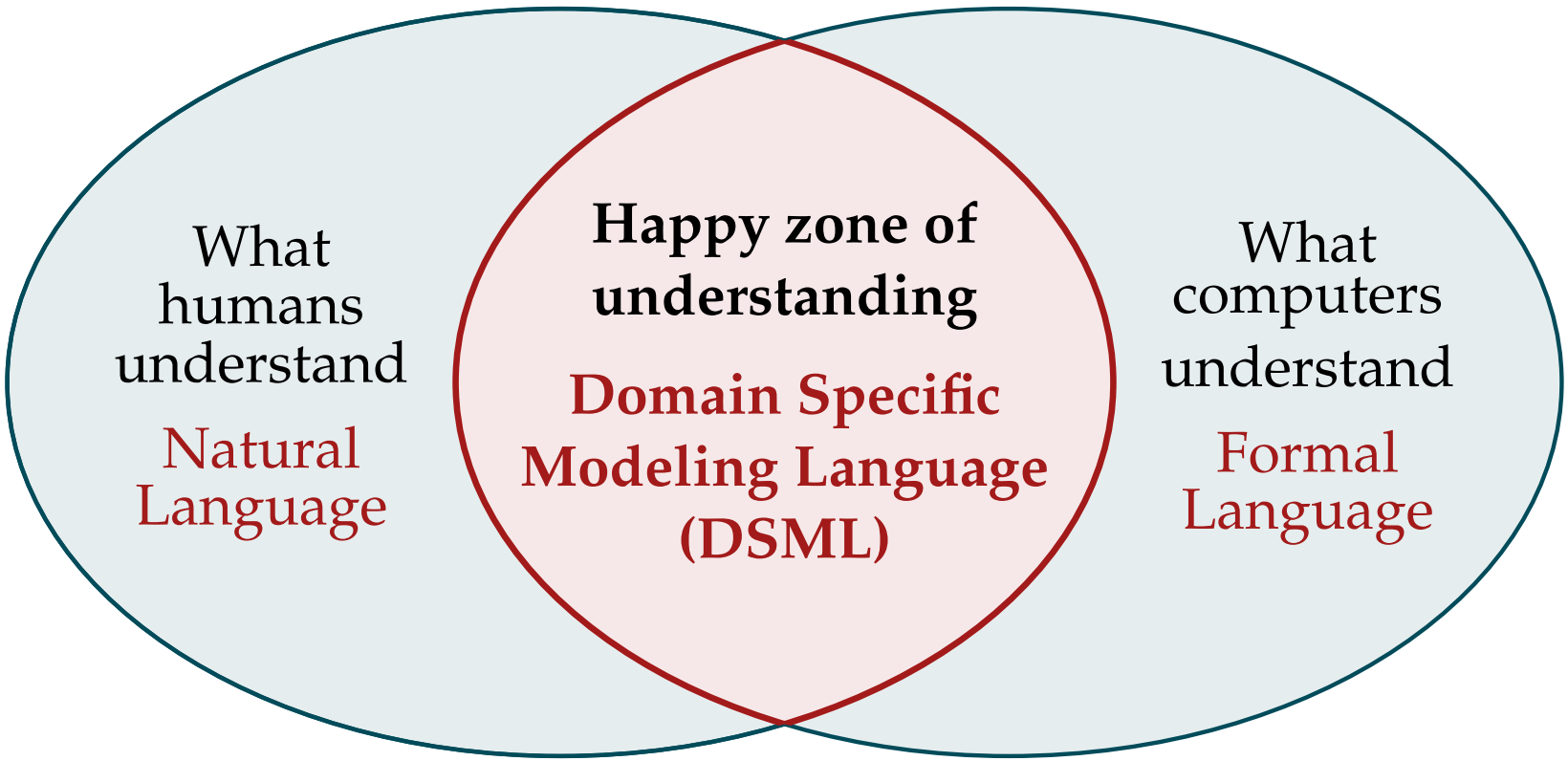
Why Modeling?



Why Modeling?



Why Modeling?



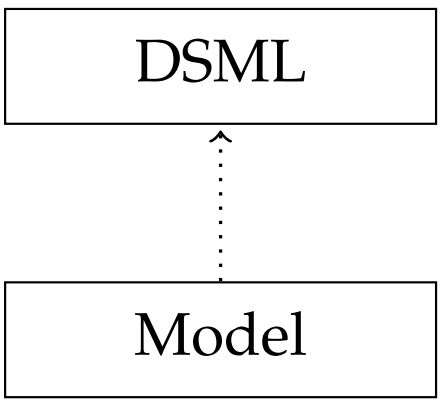
Runtime Verification

- DSMLs do not shield the software from design errors
- Testing is seldom exhaustive
- Model Checking can be performed on most models, but how to detect the unexpected on target platforms?
- Runtime Verification checks execution of real system
  - Consider environmental influences
  - React to failures

Integrating RV into Modeling

Goal

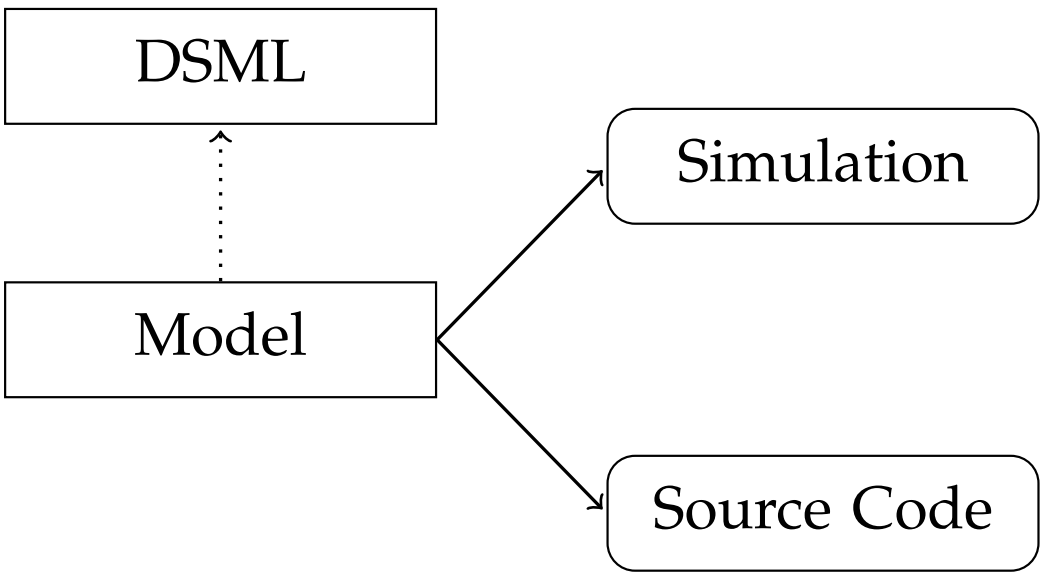
$\Rightarrow$ Integrate verification into software engineering process.
$\Rightarrow$ Domain experts and verification engineers use same model.



Integrating RV into Modeling

Goal

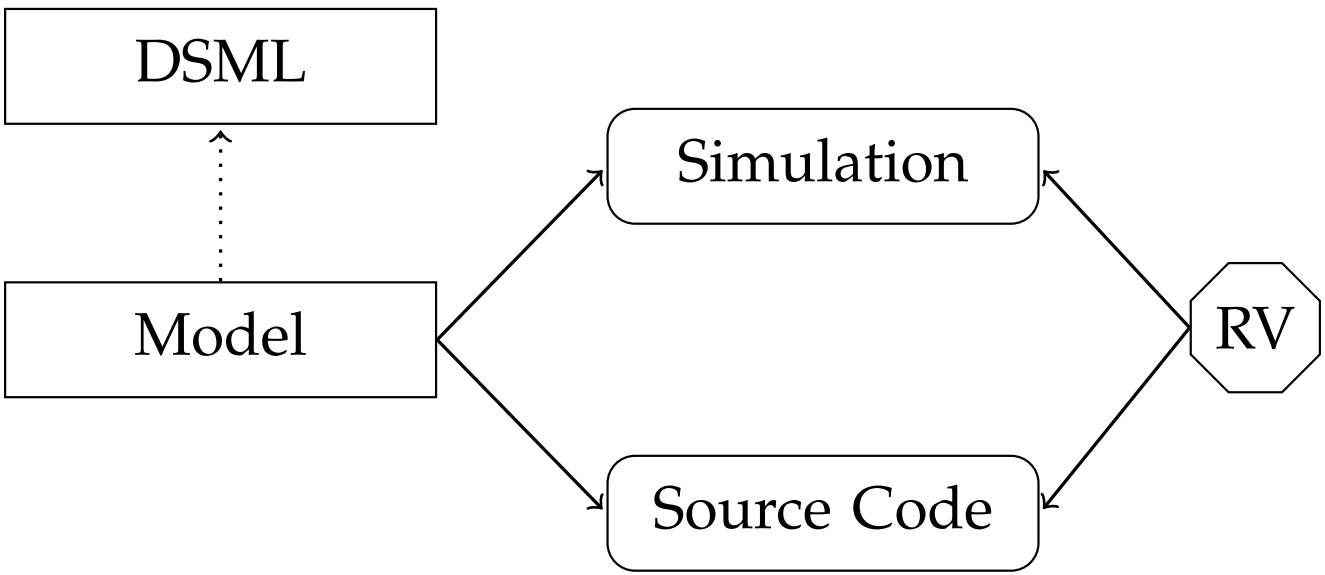
⇒ Integrate verification into software engineering process.
⇒ Domain experts and verification engineers use same model.



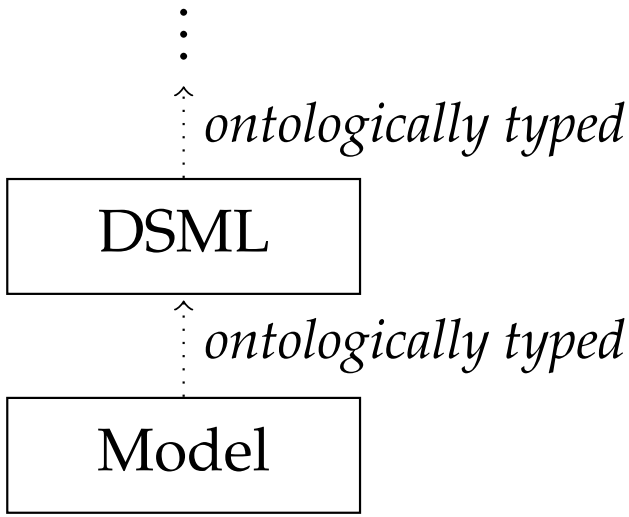
Integrating RV into Modeling

Goal

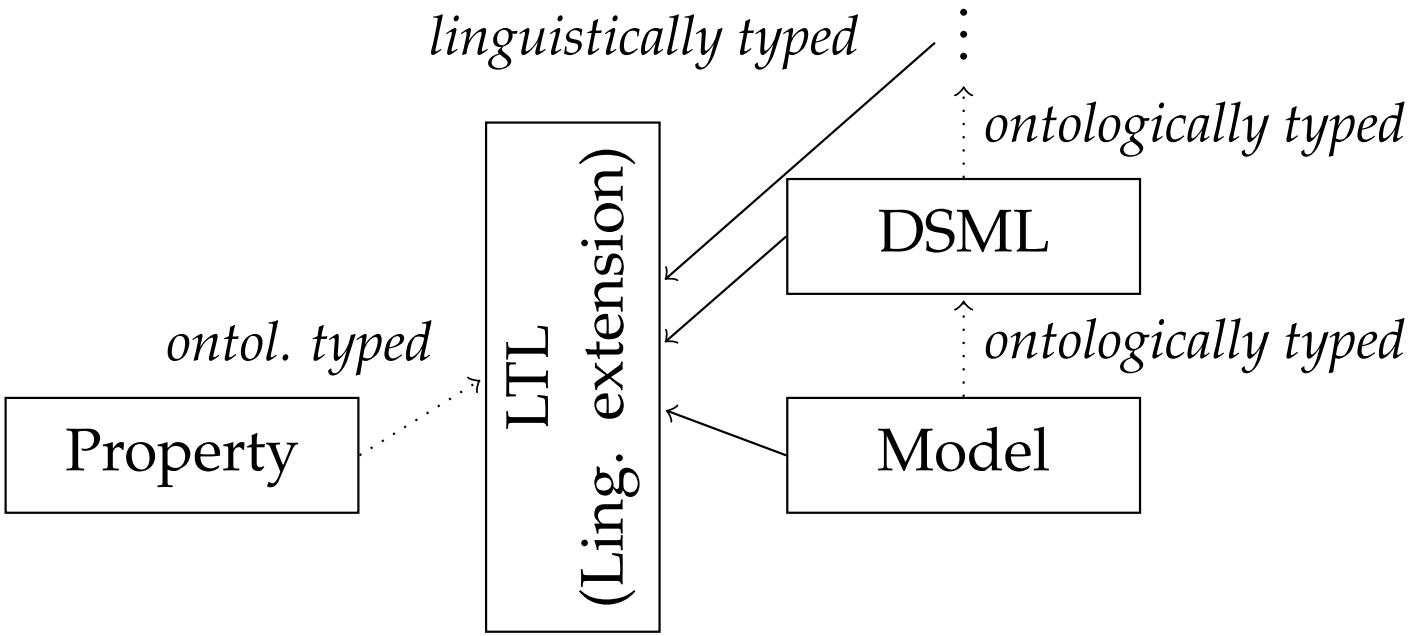
⇒ Integrate verification into software engineering process.
⇒ Domain experts and verification engineers use same model.



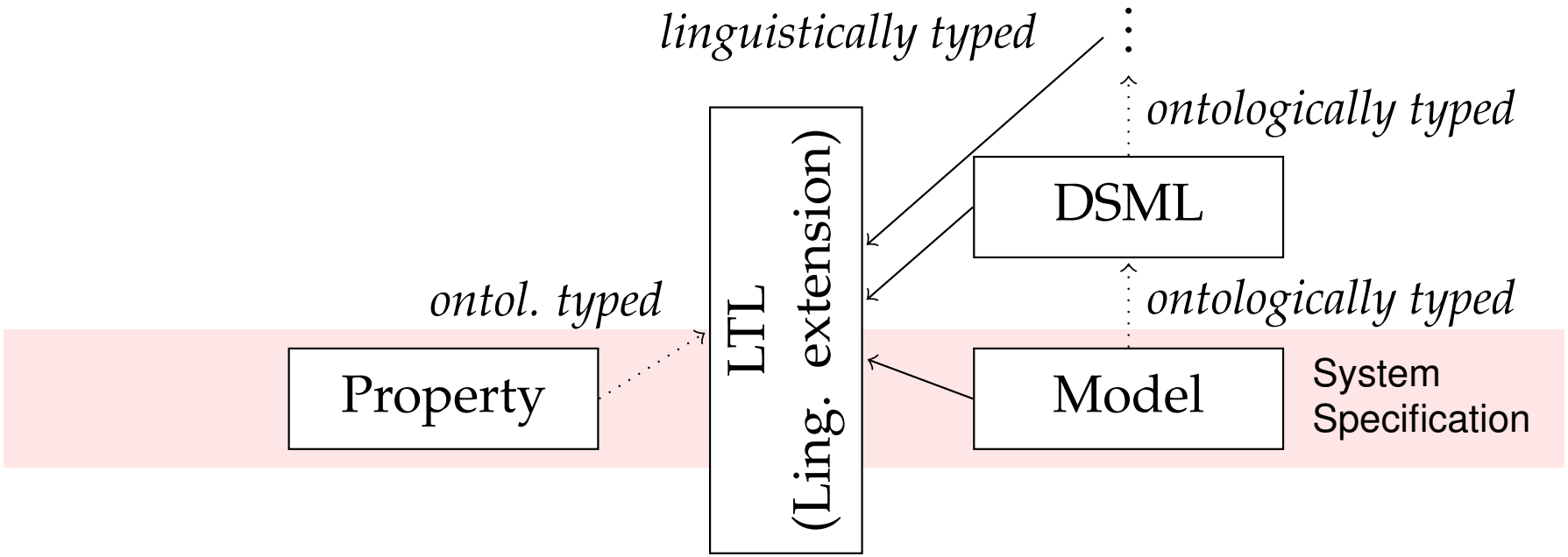
Multilevel Metamodeling



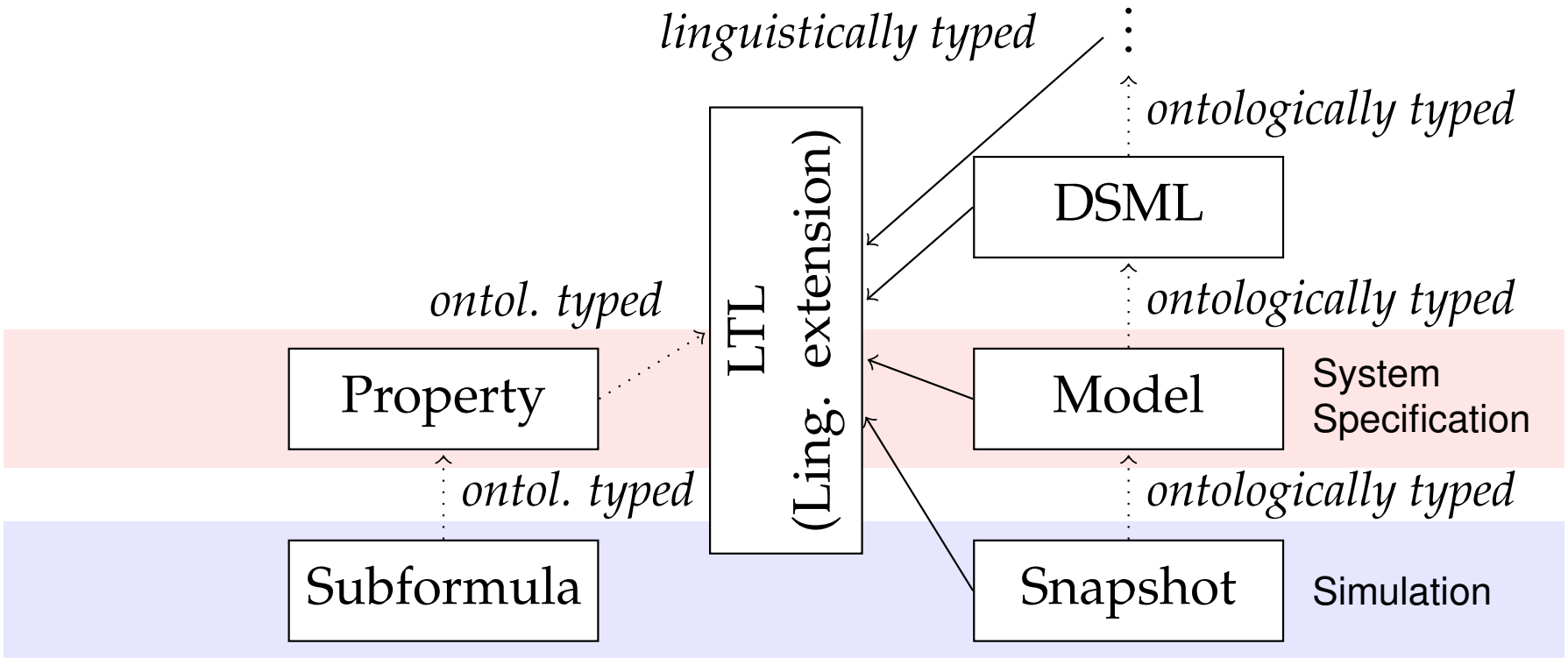
Multilevel Metamodeling



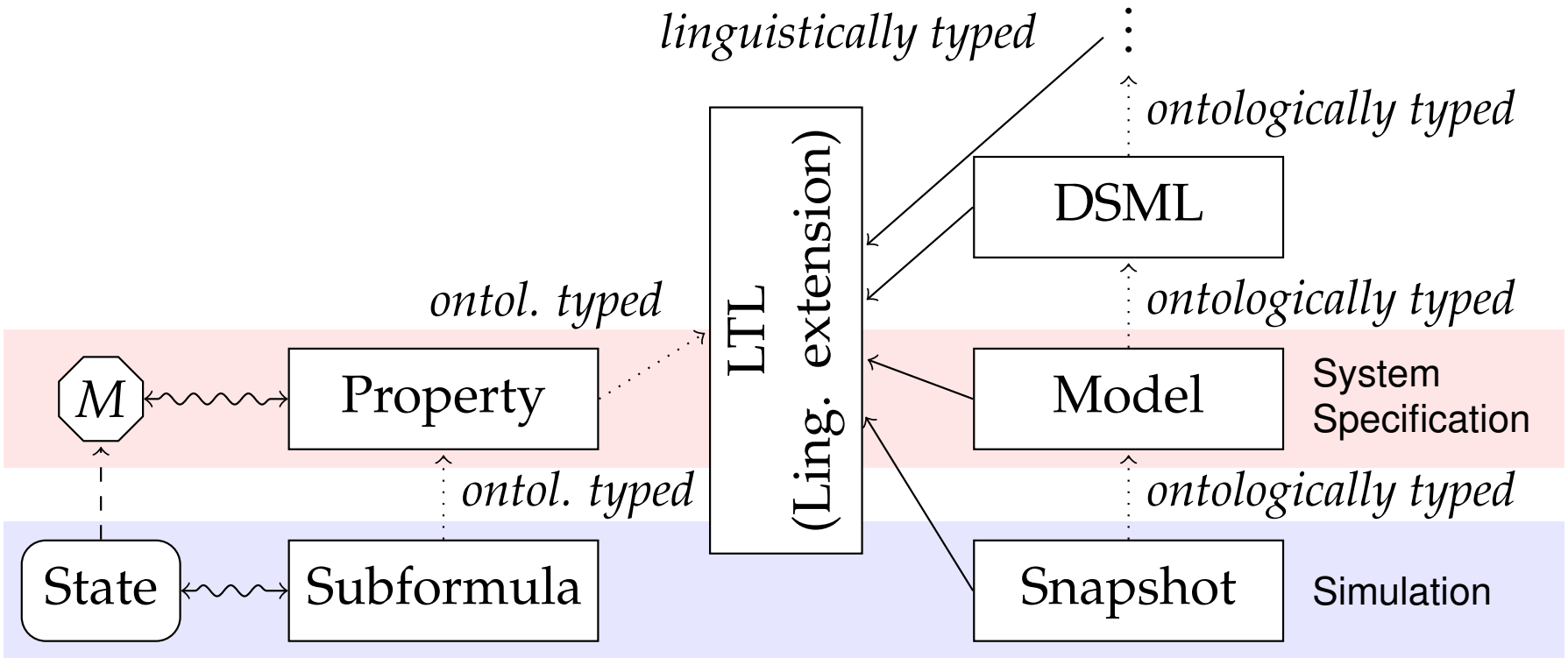
Multilevel Metamodeling



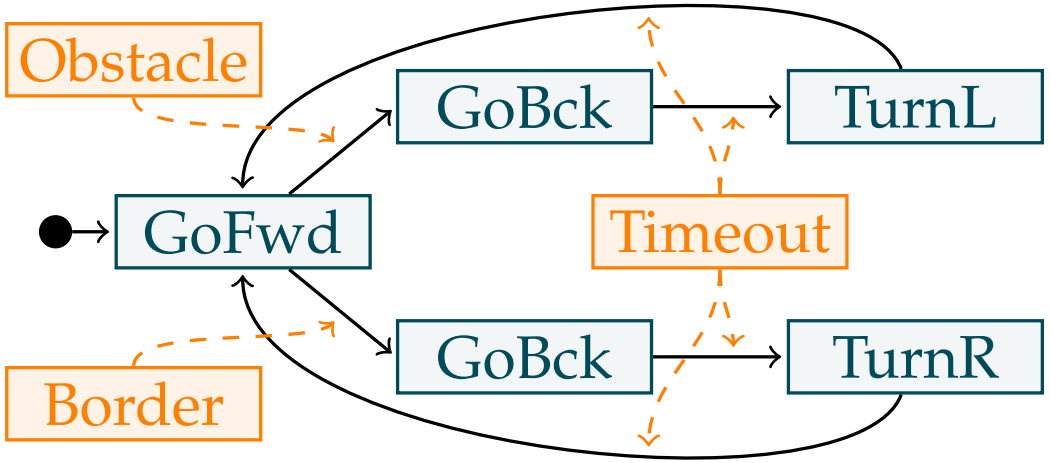
Multilevel Metamodeling



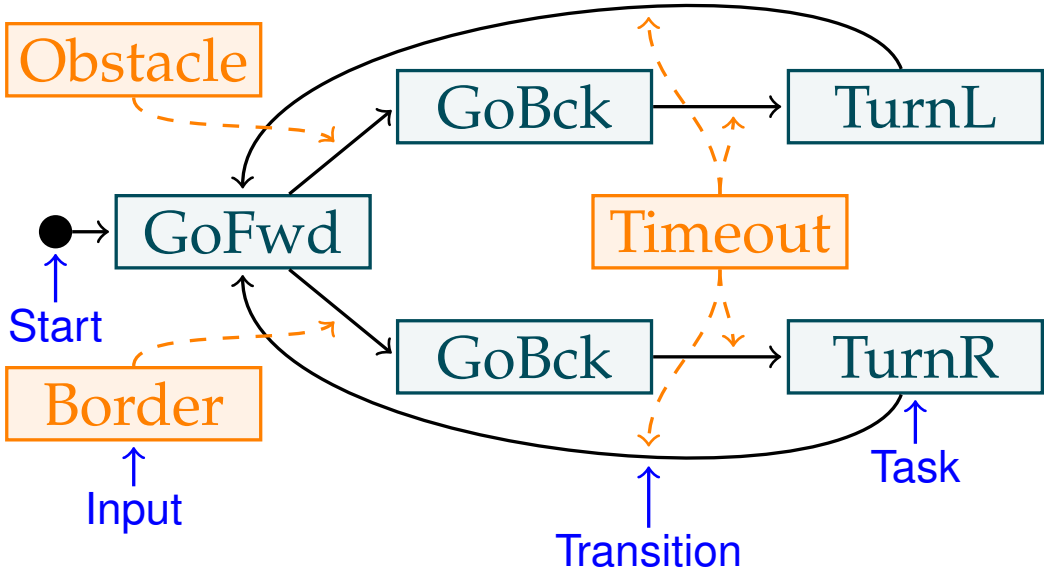
Multilevel Metamodeling



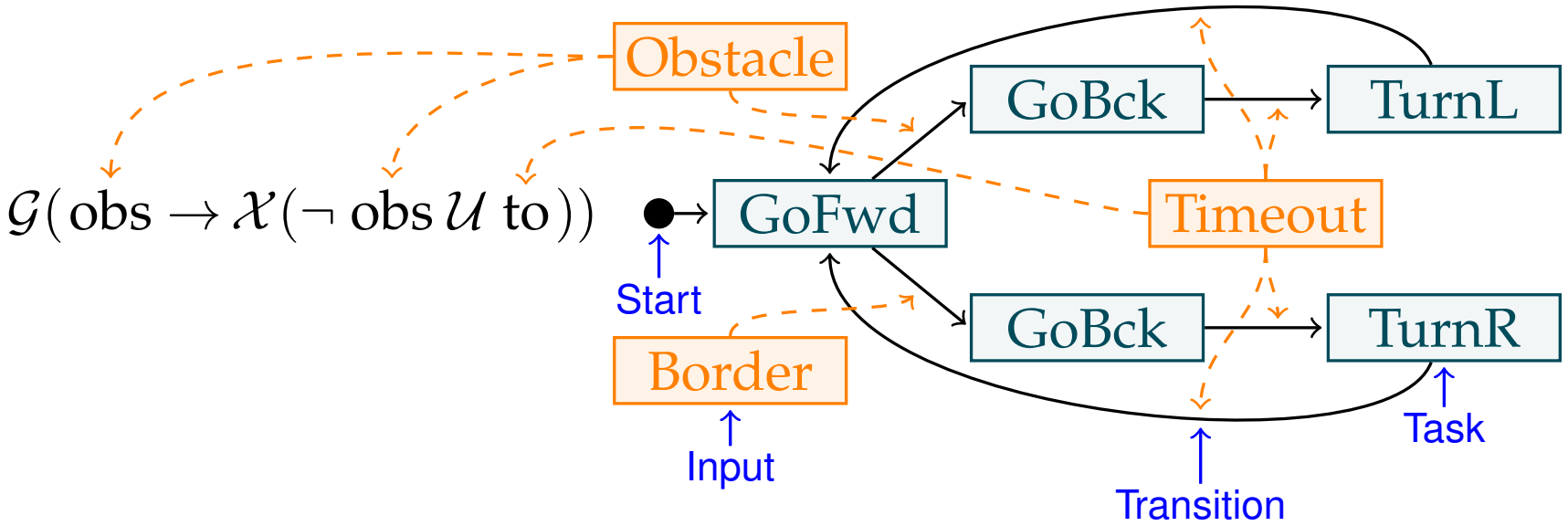
Example Scenario



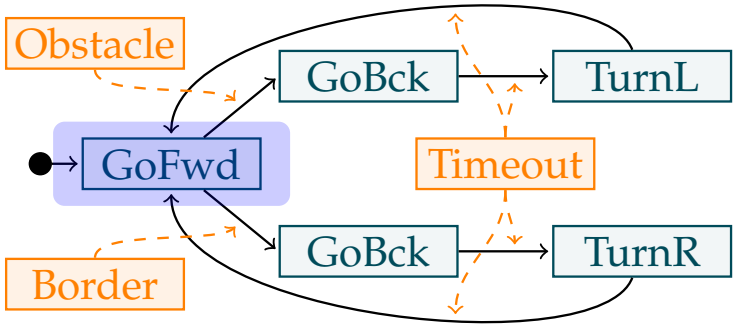
Example Scenario



Example Scenario

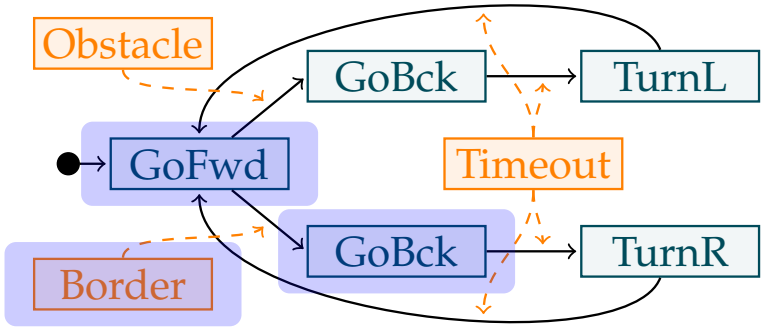


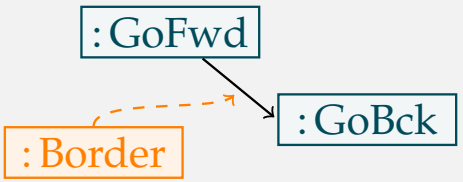
Example Simulation



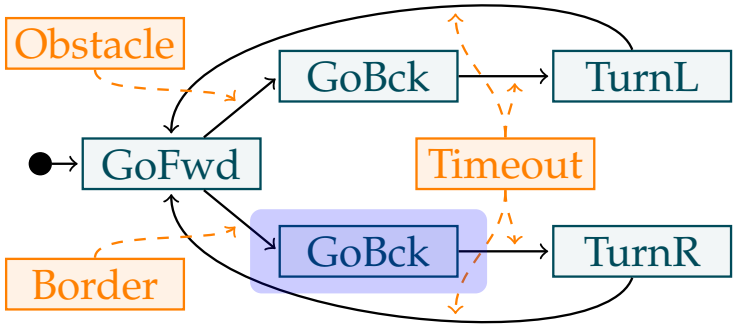
:GoFwd

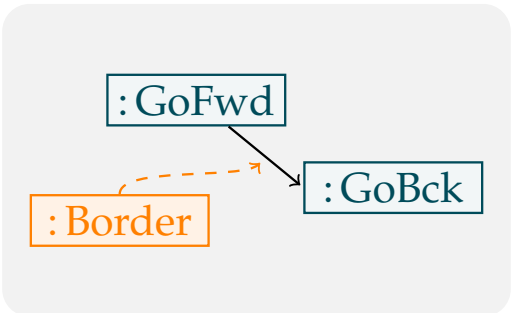
Example Simulation



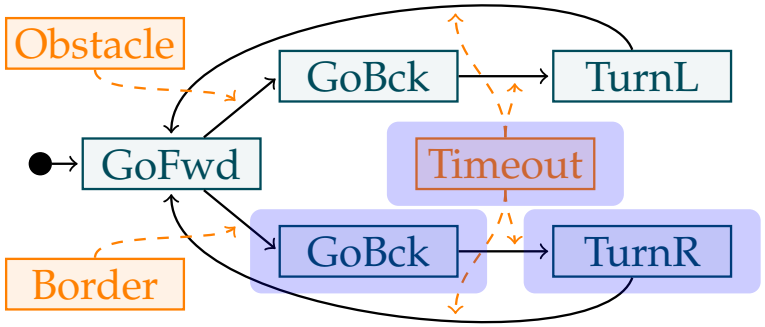


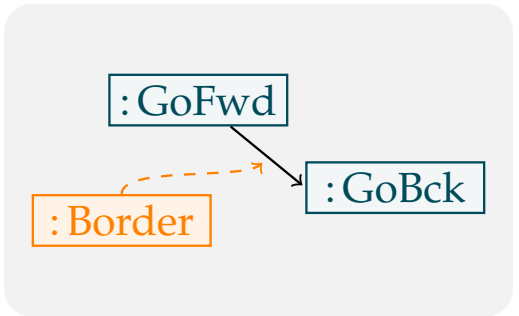
Example Simulation

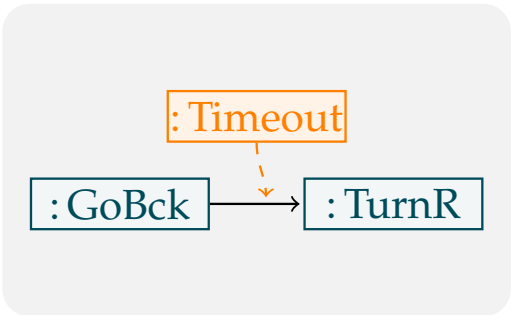




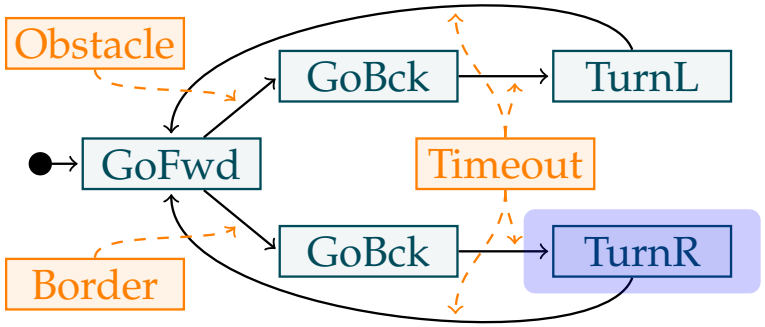
Example Simulation

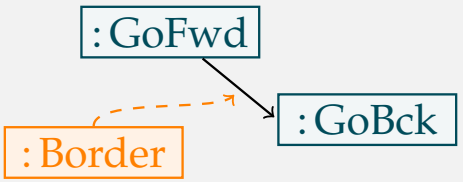


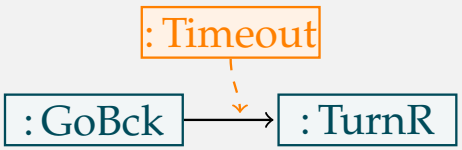




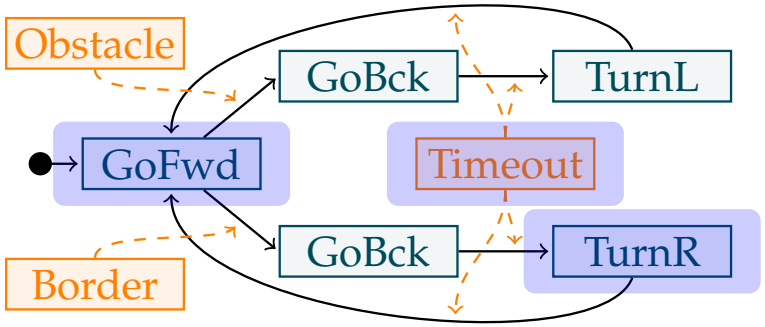
Example Simulation

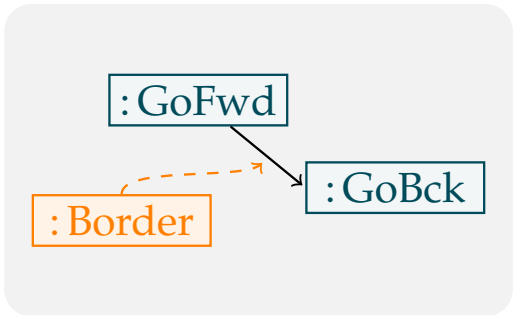


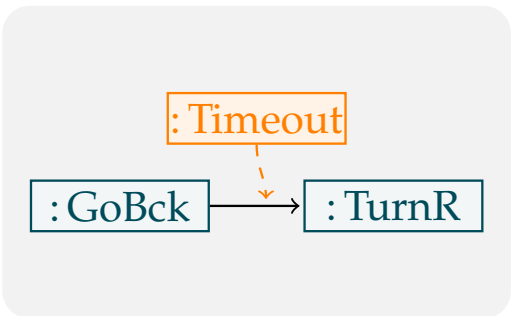


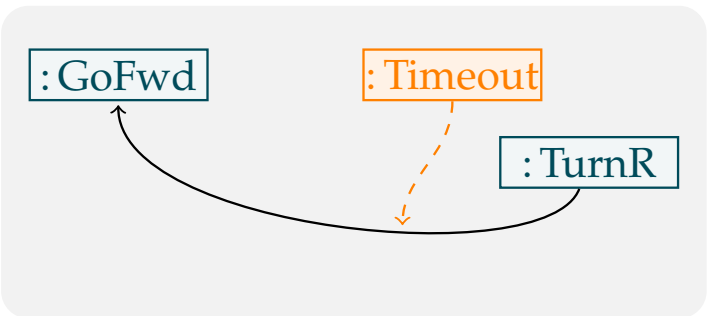


Example Simulation

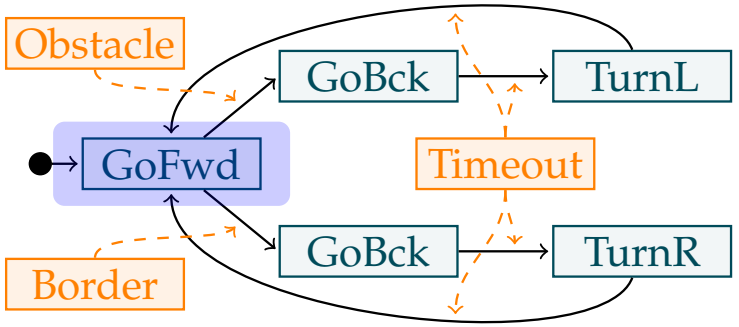








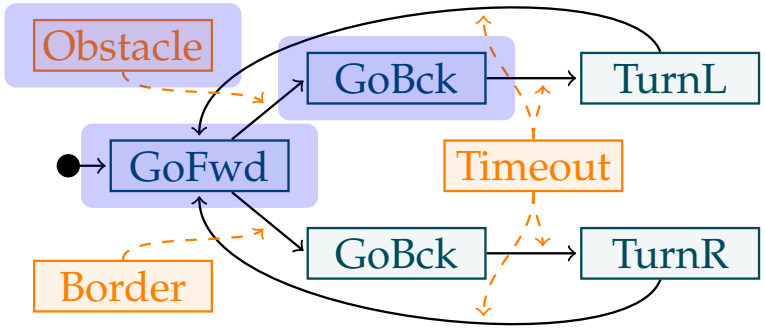
Example Simulation II



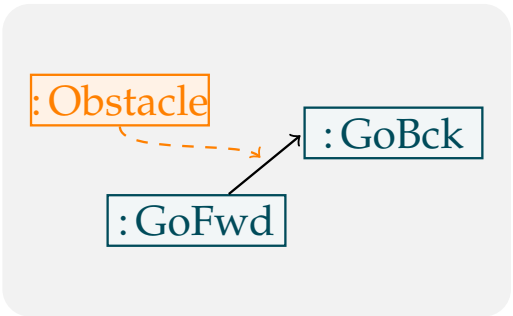
: GoFwd

$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

Example Simulation II

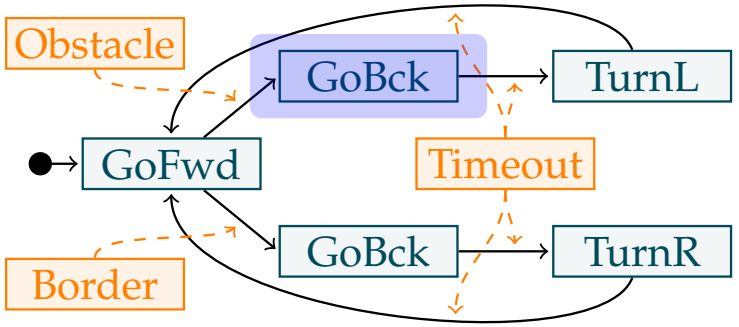


$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

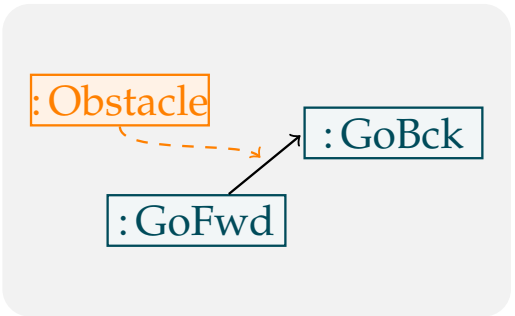


$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

Example Simulation II



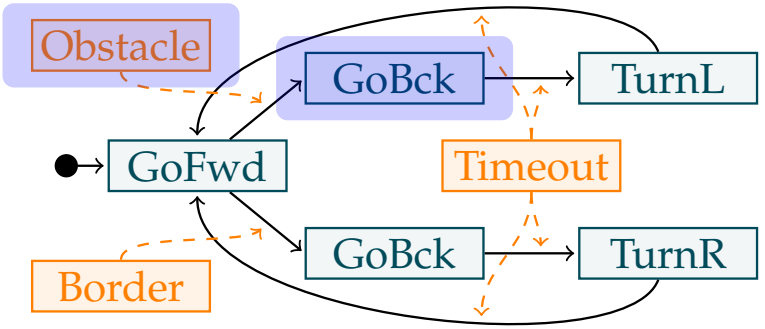
$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$



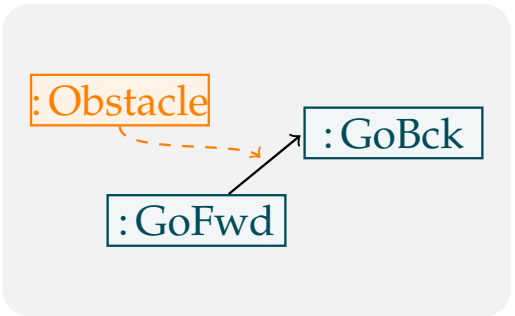
$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

$\neg\, \text{obs}\, \mathcal{U}\, \text{to}$

Example Simulation II



$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

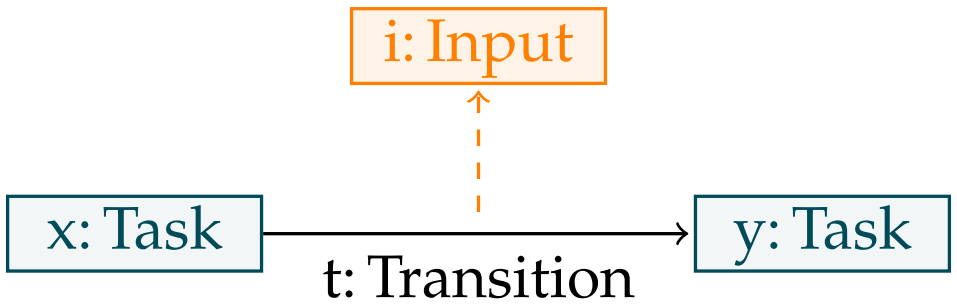


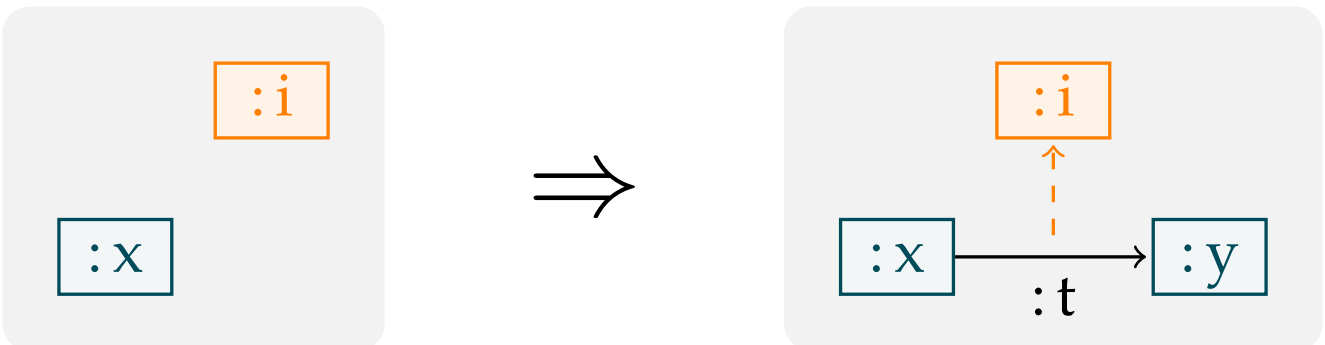
$\mathcal{G}(\text{obs} \rightarrow \mathcal{X}(\neg\, \text{obs}\, \mathcal{U}\, \text{to}))$

$\neg\, \text{obs}\, \mathcal{U}\, \text{to}$

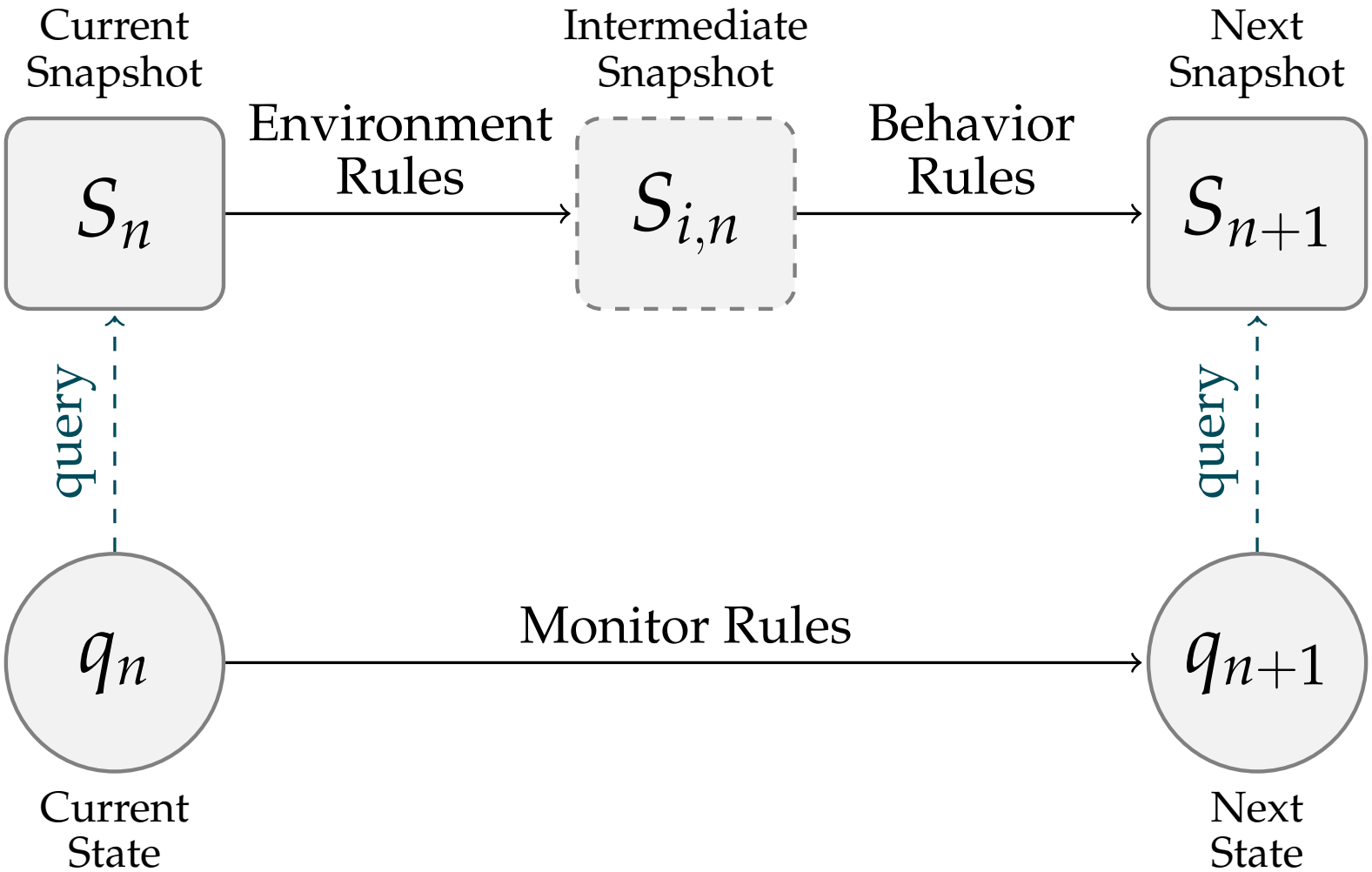
$\neg\, \text{obs}\, \mathcal{U}\, \text{to}$

Coupled Model Transformation Rules

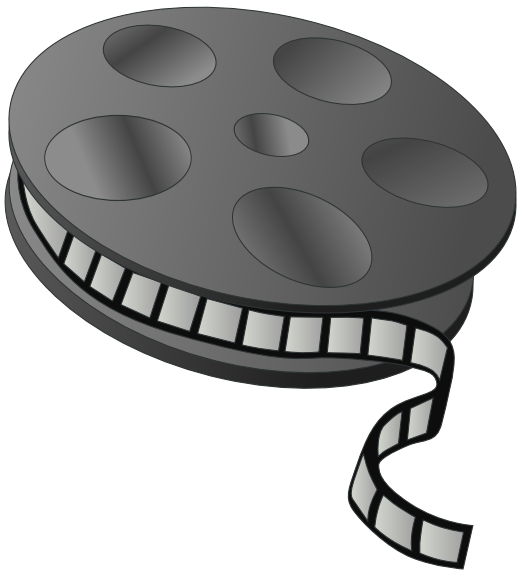




Model Transformation Rules



Code Generation



Conclusion

- Synchronize correctness property specification with behavioral model
- Check correctness of behavior model in
  - Simulation through coupled model transformation and
  - Target hardware through code generation.
- Integrate verification into early stages of development and check same properties on deployed software.

Outlook

- More Formalization
  - Coupled Model Transformation Rules
- More Application
  - Distributed Systems
  - Healthcare workflows
- More Expressiveness
  - Time
  - Data