

Consistency Analysis for User Requirements Notation Models

Okhaide Akhigbe, Daniel Amyot,
Amal Ahmed Anda, Lysanne Lessard, and
Daoyang Xiao

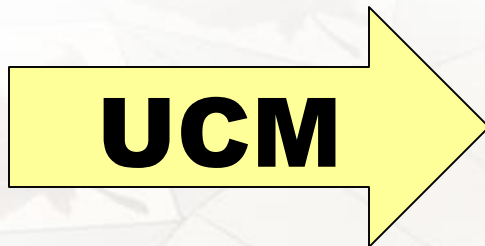
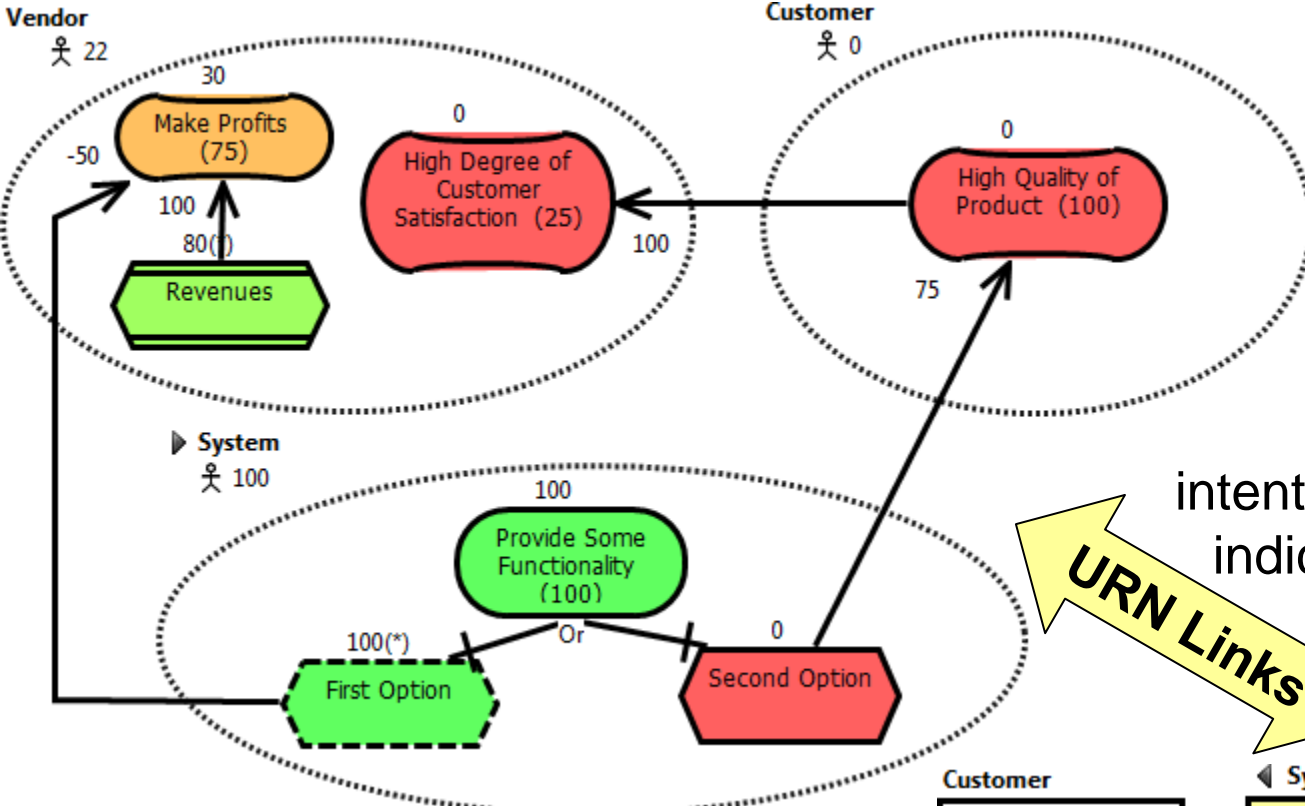
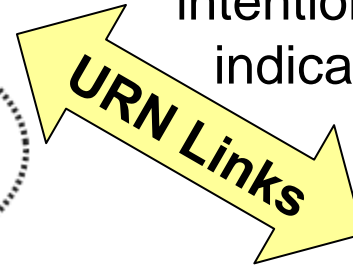
damyot@uottawa.ca

iStar 2016, Beijing

User Requirements Notation

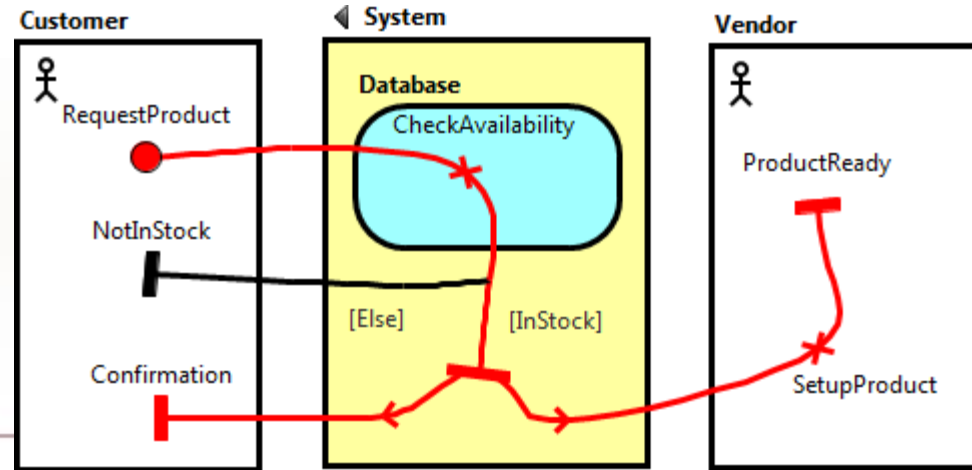


intentional elements + actors + indicators + links + strategies



responsibilities + causality + components + scenarios

<http://www.itu.int/ITU-T/studygroups/com17/index.html>





About This Presentation

- *“You can use URN links to check the consistency and completeness of your URN models”*
- Standard URN does not (yet!) provide means of checking consistency between the GRL and UCM views
- Models can hence be potentially inconsistent/erroneous!
- **New!** Preliminary **set of rules** for checking common consistency properties in URN models
- Implementation: user-selectable **OCL constraints** in the **jUCMNav** tool
- Not as trivial as one would think!
- Only the beginning... Future work identified



Consistency Analysis?

- For detecting contradictions when multiple views are used to specify different subsets of a model
- Inconsistencies occur frequently if
 - these views are provided by different modelers, or
 - when a language includes different sub-notations
- UML models are good candidates for inconsistencies
 - There is even a workshop on inconsistencies in UML!
- URN models too!
 - Goals and scenarios come from different stakeholders
- Resolving consistency issues often involves adding, deleting, or modifying elements in one or many views



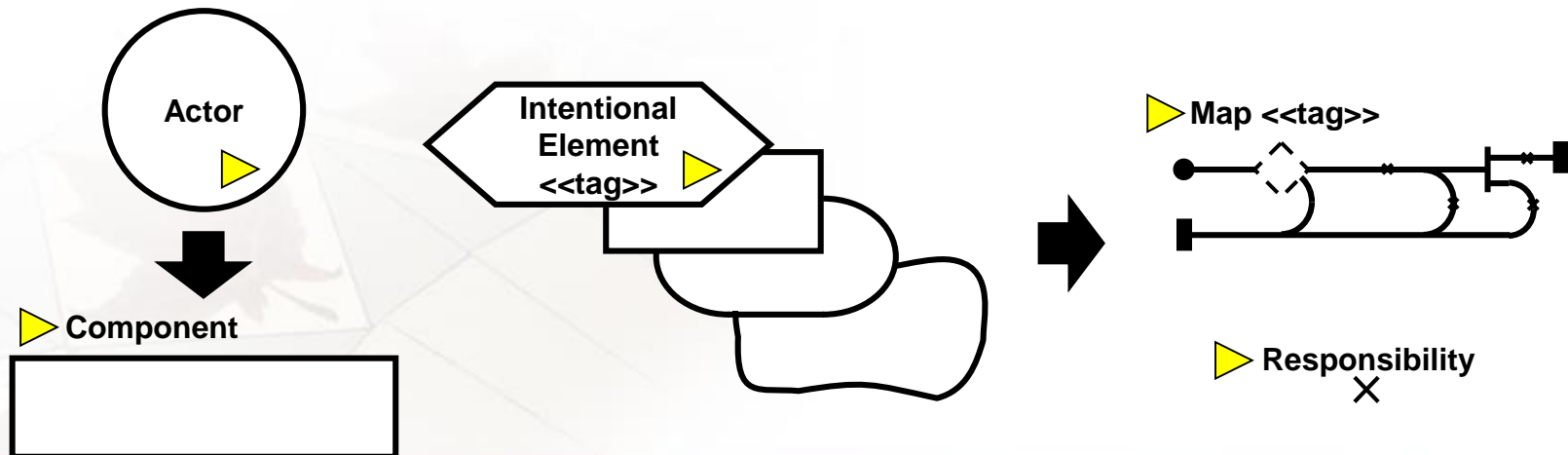
Goal/Process Inconsistencies: Not a New Problem

- Alves et al. (2013)
 - Mappings between BPMN and i* models
- Guizzardi and Reis (2015)
 - Mappings between BPMN and Tropos models
- Koliadis and Ghose (2016)
 - Trace BPMN to KAOS goals through annotations
- Sousa and Leite (2014)
 - Merge BPMN, i* and indicators into GoalBPM
- *Good source of inspiration (consistency by construction)*
- *But: no checking possible if transformed models change... Still need consistency rules*



URN Links and Metadata

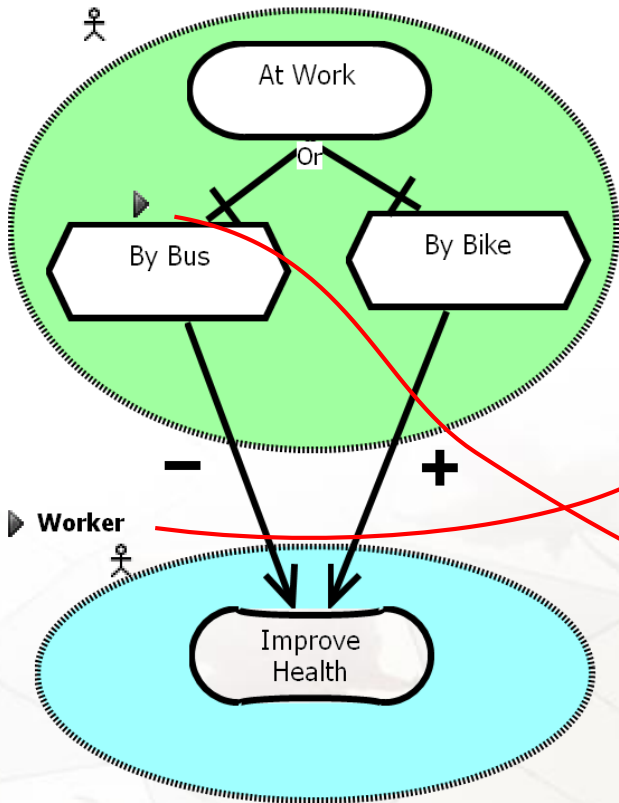
- Typed **URN links** (▶) connect any pair of URN model elements for **traceability**
 - Actors in GRL models to components in UCM models
 - Tasks/features in GRL models to maps or responsibilities in UCM models...
- **Metadata** (<<...>>) can also be attached to any URN model element for **extensibility** (name/value pair)



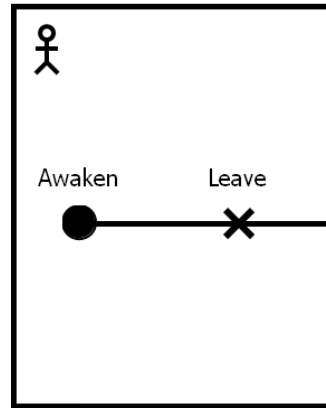


Sample Model

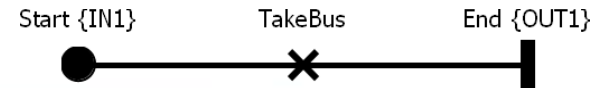
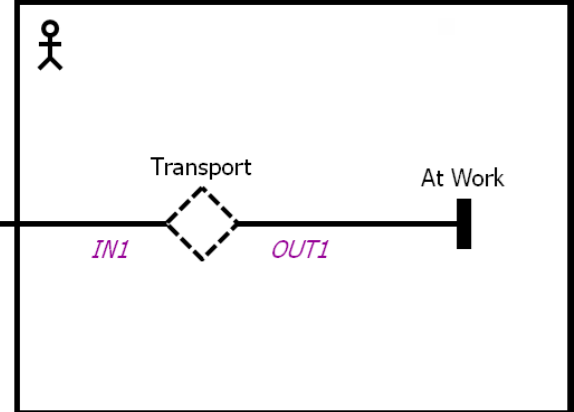
Transportation System



Worker

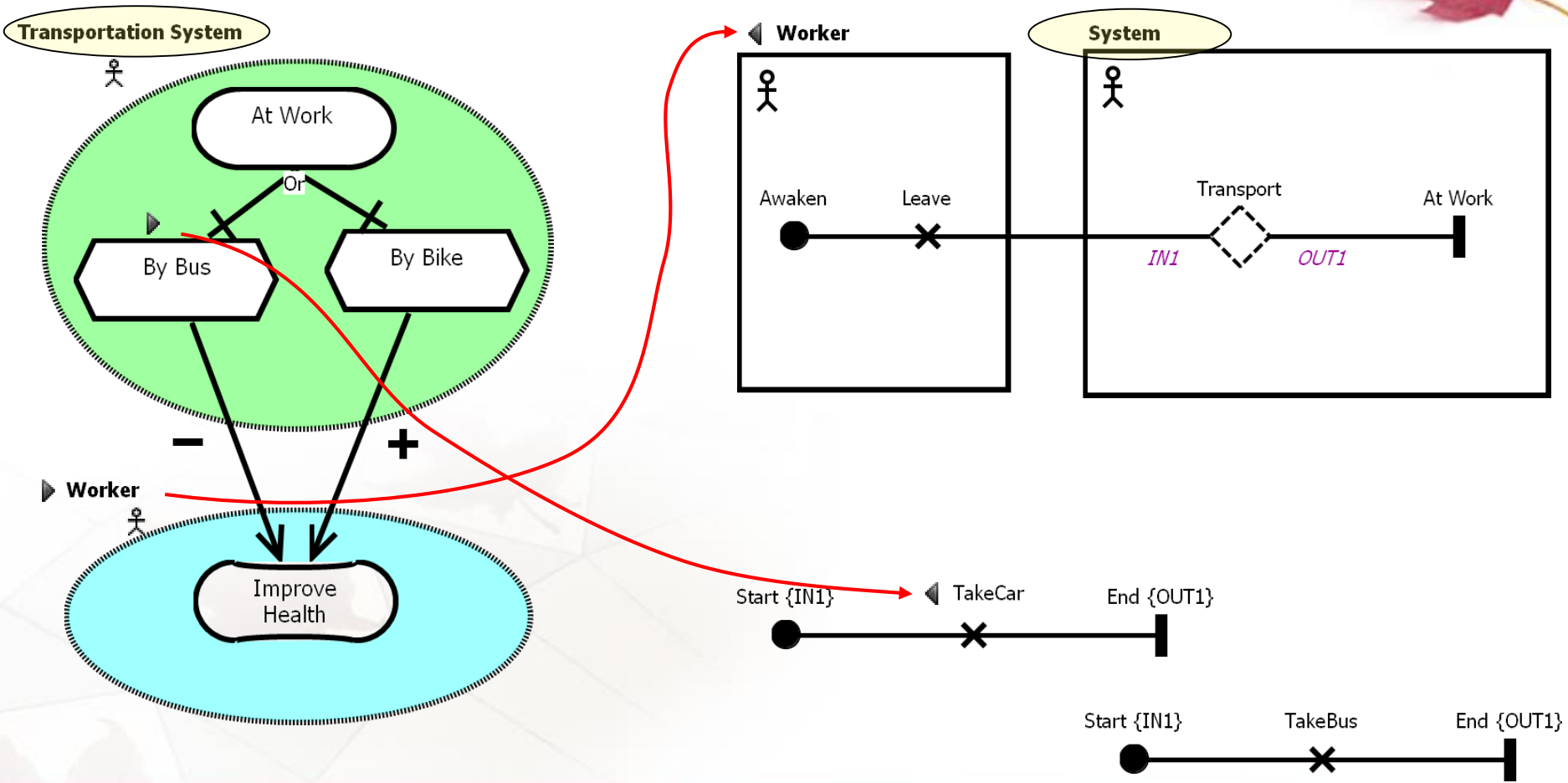


System





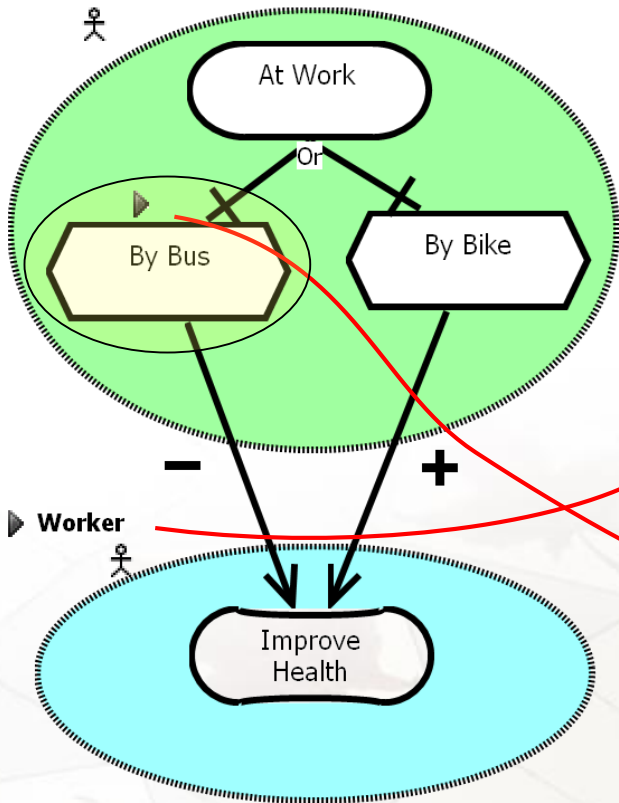
Likely Missing Links (Syntactic)





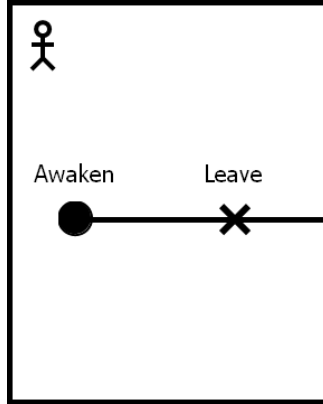
Incorrect Link (Semantic)

Transportation System

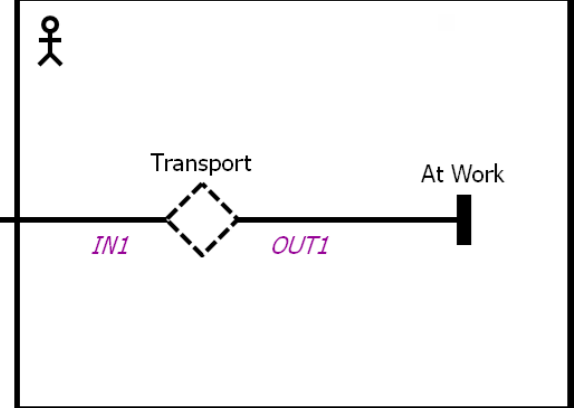


Worker

Worker



System



Start {IN1}



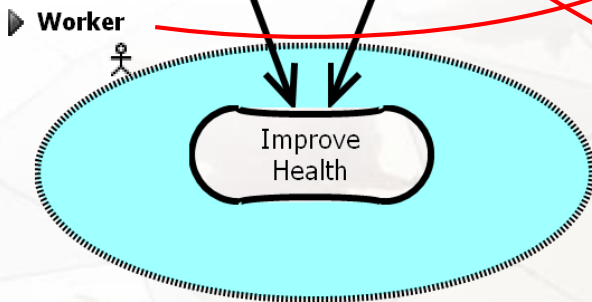
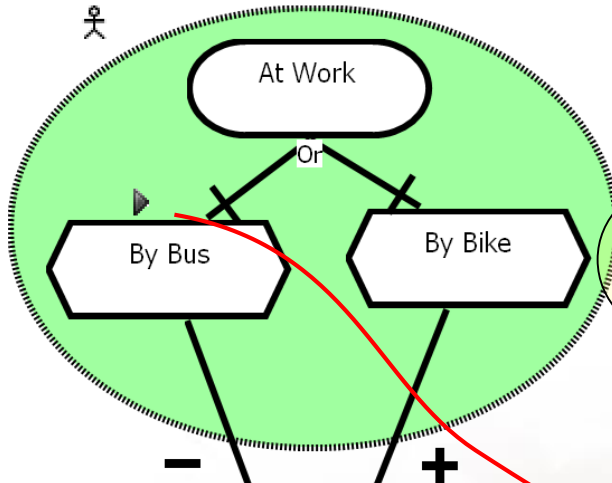
Start {IN1}



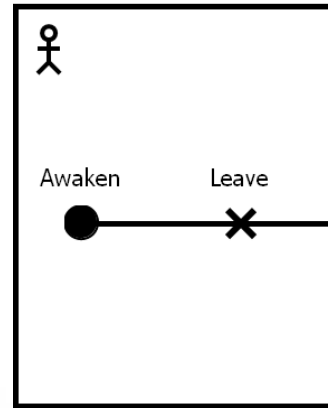


Missing/Extra Element (Syntactic)

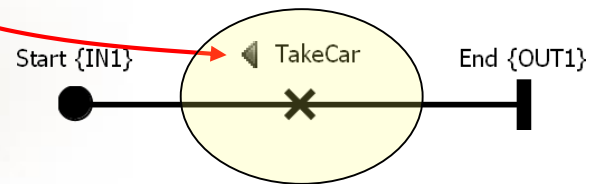
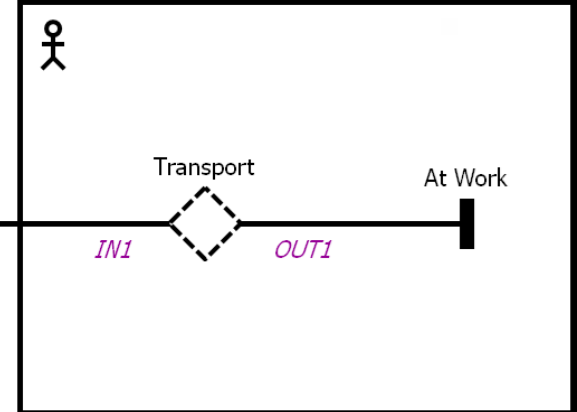
Transportation System



Worker



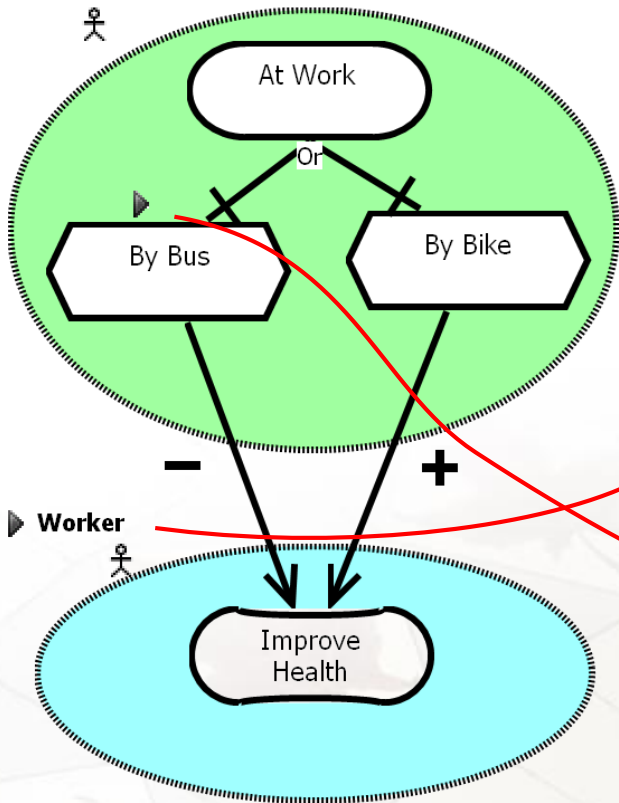
System



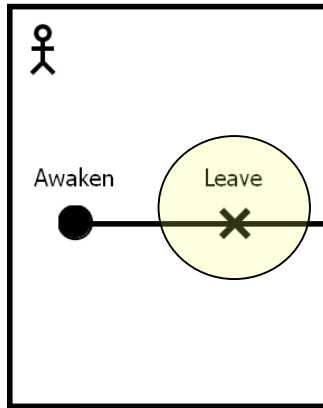


Element that Should Not Have a Link (Syntactic)

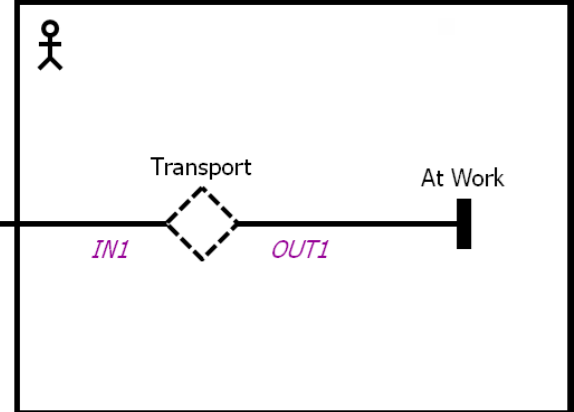
Transportation System



Worker



System





Solution Investigated

- Rules as OCL constraints
- Modeler-provided information
 - URN links (directional) of type *Traces*
 - Metadata (name=Traces, value=No) for elements that should not be linked (no rule violation)
- Try to take advantage of decompositions in GRL and component/stub containment in UCM
- Implementation with jUCMNav

type filter text

- [-] General
- [-] Ant
- [-] ATL
- [-] CDO
- [-] Ecore Diagram
- [-] Ecore Tools Diagram
- [-] EMF Compare
- [-] Help
- [-] Install/Update
- [-] Java
- [-] JET Transformations
- [-] jUCMNav
 - [-] AutoLayout Preferences
 - [-] Color Preferences
 - [-] GRL Strategy Evaluation /
 - [-] KPI Monitoring Preference
 - [-] Metrics Preferences
 - [-] Outline Preferences
 - [-] Reference/Definition Prefi
 - [-] Report Generation Preferen
 - [-] Scenario Export Preferenc
 - [-] Static Semantics Checking
 - [-] UCM Scenario Traversal A
- [-] Model Validation
- [-] Plug-in Development
- [-] Run/Debug
- [-] Tasks
- [-] Team
- [-] Teneo
- [-] UML2 Diagrams
- [-] Usage Data Collector
- [-] Validation
- [-] XML
- [-] Xtend/Xpand
- [-] Xtext
- [-] Xtext Languages

Static Semantics Checking Preferences

Show rule description in the rule violation report

Rules defined:

Name	Context	Query Expr	
<input checked="" type="checkbox"/> All			
<input type="checkbox"/> Performance Scenario			
<input type="checkbox"/> Aspect			
<input type="checkbox"/> Performance			
<input checked="" type="checkbox"/> iStarLoose			
<input checked="" type="checkbox"/> NoBeliefAsLinkDestination	grl::Belief	getAllBeliefs()	Beliefs should not be the dest...
<input checked="" type="checkbox"/> TasksAsDestOfAndLinks	grl::Decomposition	getAllDecompositions()	AND decomposition links shoul...
<input checked="" type="checkbox"/> MeansEndMustHaveGoalDest	grl::Decomposition	getAllDecompositions()	OR/IOR decomposition (mean...
<input checked="" type="checkbox"/> IsPartOfBetweenSimilarActors	grl::Actor	getAllActors()	Is Part Of association should ...
<input checked="" type="checkbox"/> AllGRLdiagramsSRorSD	grl::GRLGraph	getAllGRLGraphs()	All GRL diagrams should be ta...
<input checked="" type="checkbox"/> SRdependencyAlwaysHasDependums	grl::Dependency	getAllDependencies()	Dependency links in an SR mo...
<input checked="" type="checkbox"/> OnlyDependenciesCrossing	grl::LinkRef	getAllLinkRefs()	The only links that cross actor...
<input type="checkbox"/> Feature			
<input type="checkbox"/> ConsistencyChecks			
<input checked="" type="checkbox"/> iStarStrict			
<input checked="" type="checkbox"/> SoftgoalAsContributionDestination	grl::Contribution	getAllContributions()	The destination of a Contribu...
<input checked="" type="checkbox"/> DecompositionLinkForbidenDestinations	grl::Decomposition	getAllDecompositions()	Decomposition links must not ...
<input checked="" type="checkbox"/> DecompositionLinkForbidenSources	grl::Decomposition	getAllDecompositions()	Decomposition links must not ...
<input checked="" type="checkbox"/> ISAbetweenSimilarActors	grl::Actor	getAllActors()	ISA (generalization) must be ...
<input checked="" type="checkbox"/> CoversFromPositionToRole	grl::Actor	getAllActors()	A Covers association must be...
<input checked="" type="checkbox"/> PlaysFromAgentToRole	grl::Actor	getAllActors()	A Plays association must be fr...
<input checked="" type="checkbox"/> OccupiesFromagentToPosition	grl::Actor	getAllActors()	An Occupies association must...
<input checked="" type="checkbox"/> INSbetweenAgents	grl::Actor	getAllActors()	An INS association must only ...
<input checked="" type="checkbox"/> NoDependencyInsideActor	grl::LinkRef	getAllDependencies()....	Dependency links must never ...
<input checked="" type="checkbox"/> DependencyInSDwithoutDependum	grl::GRLGraph	getAllGRLGraphs()	Dependency links in an SD mo...

Constraints can be created/deleted, grouped, and selected

New Rule New Group Edit Delete Import Export

Restore Defaults Apply

OK Cancel



type filter text

- General
- Ant
- ATL
- CDO
- Ecore Diagram
- Ecore Tools Diagram
- EMF Compare
- Help
- Install/Update
- Java
- JET Transformations
- jUCMNav
 - AutoLayout Preferences
 - Color Preferences
 - GRL Strategy Evaluation
 - KPI Monitoring Preferen
 - Metrics Preferences
 - Outline Preferences
 - Reference/Definition Pre
 - Report Generation Prefe
 - Scenario Export Prefere
 - Static Semantics Checkin
 - UCM Scenario Traversal
- Model Validation
- Plug-in Development
- Run/Debug
- Tasks
- Team
- Teneo
- UML2 Diagrams
- Usage Data Collector
- Validation
- XML
- Xtend/Xpand
- Xtext
- Xtext Languages

Static Semantics Checking Preferences

Show rule description in the rule violation report

Rules defined:

Edit a rule

Rule Name:

Context:

OCL query expression for collecting all objects to be checked:

OCL constraint expression:

Description:

Report as Warning instead of Error

New Edit Delete

OK Cancel

Example of a constraint definition

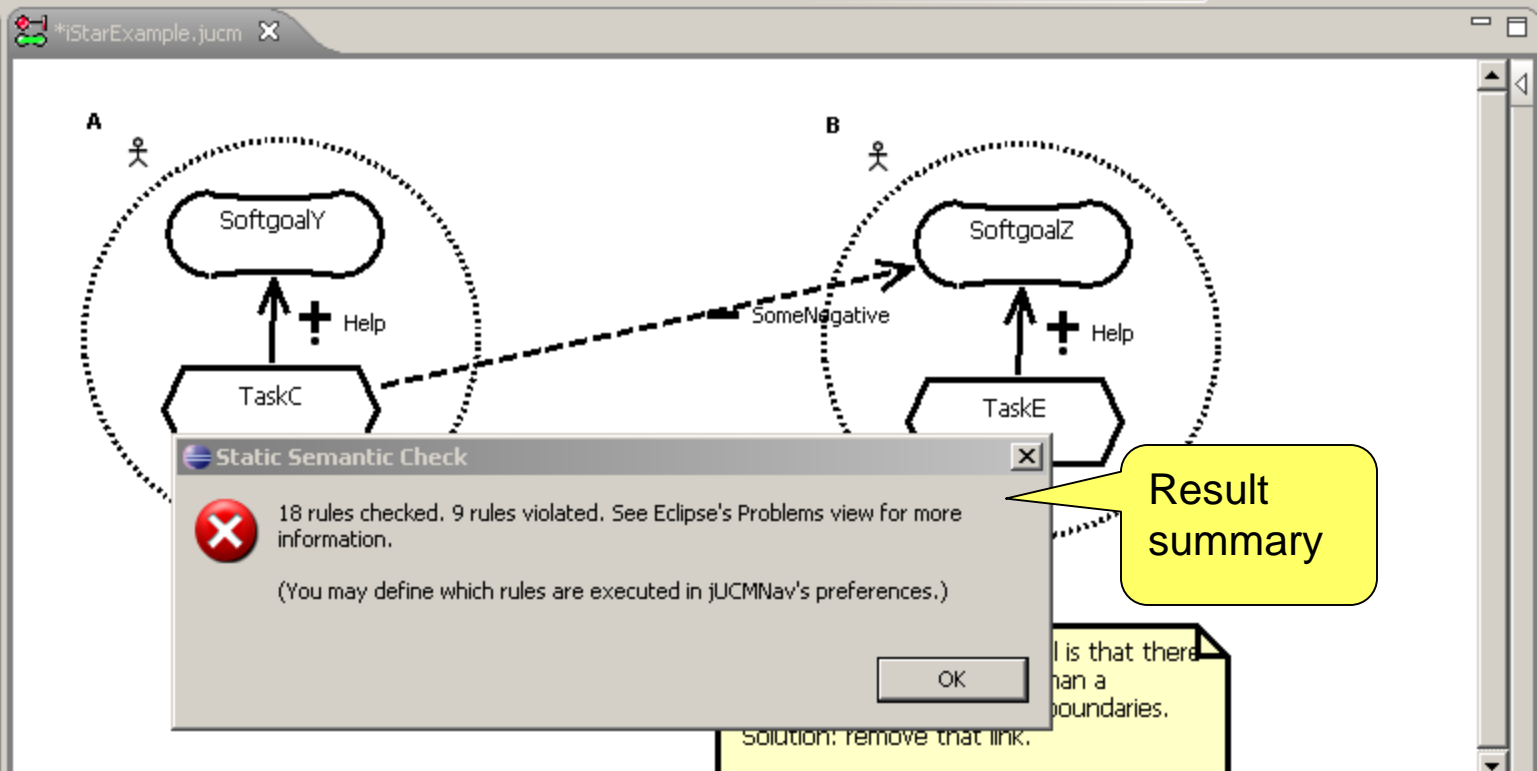
ld not be the dest...
 position links shoul...
 composition (mean...
 association should ...
 rams should be ta...
 y links in an SR mo...
 ks that cross actor...
 tion of a Contribu...
 ion links must not ...
 ion links must not ...
 alization) must be ...
 association must be...
 ociation must be fr...
 s association must...
 ociation must only ...
 y links must never ...
 y links in an SD mo...

Restore Defaults Apply

OK Cancel

Outli Elem

- Actor Definitions
 - A
 - AnotherSubactor
 - B
 - MyAgent
 - SomeRole
 - SubActor
 - SuperActor
- Intentional Element Definition
 - Dependee
 - Depender
 - Goal1
 - Goal3
 - Goal6
 - GoalY
 - Resource5
 - Softgoal2
 - Softgoal7
 - SoftgoalX
 - SoftgoalY
 - SoftgoalZ
 - SomeGoal
 - SomeOtherTask
 - SomeSoftgoal
 - SomeTask
 - Task4
 - Task8



Result summary

is that there than a boundaries.
 SOLUTION: remove that link.

SomeGRLGraph SD model SR model SR-Crossing SR Missing Dependum

Problems

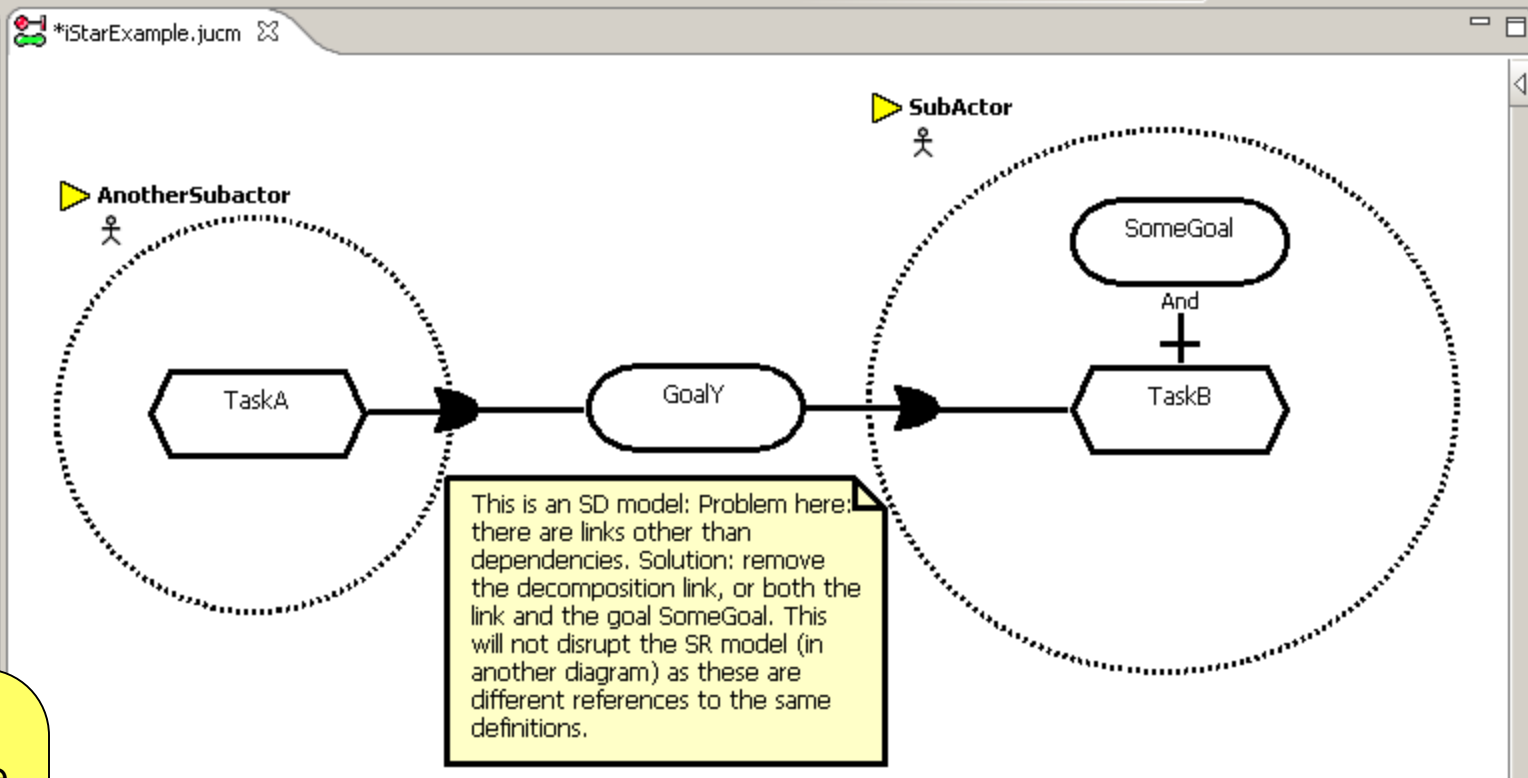
6 errors, 7 warnings, 1 other

Description	Location	Type
Errors (6 items)		
✗ The destination of a Contribution must be a Softgoal (SoftgoalAsContributionDestination)	Contribution50	Problem
✗ Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition74	Problem
✗ Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition77	Problem
✗ Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition95	Problem
✗ SD models must not have links other than dependency and actor association links (SDmodelsRestrictedLinks)	SD model	Problem
✗ Dependency links must never completely be inside of an actor boundary (NoDependencyInsideActor)	grl.impl.LinkRef...	Problem
Warnings (7 items)		
⚠ AND decomposition links should only have tasks as destinations (TasksAsDestOfAndLinks)	Decomposition74	Problem

Errors and warnings generated

Outli Elem

- Actor Definitions
 - A
 - AnotherSubactor
 - B
 - MyAgent
 - SomeRole
 - SubActor
 - SuperActor
- Intentional Element Definiot
 - Dependee
 - Depender
 - Goal1
 - Goal3
 - Goal6
 - GoalY
 - Resource5
 - Software3



Clicking on a problem brings the violating model element on screen (in the Editor or in the Outline)

Errors, 7 warnings, 1 other

Description	Location	Type
Errors (6 items)		
The destination of a Contribution must be a Softgoal (SoftgoalAsContributionDestination)	Contribution50	Problem
Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition74	Problem
Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition77	Problem
Decomposition links must not have softgoals, resources or beliefs as a destination. (DecompositionLinkForbide	Decomposition95	Problem
SD models must not have links other than dependency and actor association links (SDmodelsRestrictedLinks)	SD model	Problem
Dependency links must never completely be inside of an actor boundary (NoDependencyInsideActor)	grl.impl.LinkRef...	Problem
Warnings (7 items)		
AND decomposition links should only have tasks as destinations (TasksAsDestOfAndLinks)	Decomposition74	Problem

- TaskA
- TaskB
- TaskC
- TaskD
- TaskE
- KPI Information Element Defi



Sample Rules: Actor → Component

```
context gr1::Actor
  inv URNconsAllActorsToComponents:
    -- #1a: Each GRL actor must have a Traces link to a UCM component,
    -- unless tagged with Traces=No
    not(getMetadata('Traces')='No') implies
      (getLinksToForType('Traces')->size() > 0)

  inv URNconsActorsToComponentsOnly:
    -- #1b: Traces Links from a GRL actor must only be to
    -- a UCM *component*
    not(getMetadata('Traces')='No') implies
      ( getLinksToForType('Traces')->
        forAll(me:urncore::URNmodelElement |
              me.oclIsKindOf(urncore::Component) ) )
```



Only Half the Story...

- Each GRL actor must have a Traces link to a UCM component, unless tagged with Traces=No
- Traces links from a GRL actor must only be to a UCM *component*

But also from the UCM point of view

- Each UCM component must have a Traces link from a GRL actor, unless tagged with Traces=No
- Traces links to a UCM component must only be from a GRL *actor*

Alternatives Can Be Considered



- GRL **intentional element** to UCM map or responsibility
 - Six OCL invariants
- GRL **task** (only) to UCM map or responsibility
 - Six OCL invariants
- The user would choose between these two options for a given model



Save Time, Exploit Structures

- Risk: too many links to create manually... Burdensome!
- Solution: exploit structural information in models
- GRL decomposition
 - If parent intentional element is linked, this also covers the children, recursively
- UCM components
 - If parent component is linked, this also covers the sub-components, recursively
- UCM stubs
 - If stub is linked, this also covers the plugin maps, recursively



Set of Rules (So Far...)

ID	Description
1a	Each <u>GRL actor</u> must have a Traces link <u>to a UCM component</u> , unless tagged with Traces=No
1b	Traces links <u>from a GRL actor</u> must only be <u>to a UCM *component*</u>
1c	Each <u>UCM component</u> must have a Traces link <u>from a GRL actor</u> , unless tagged with Traces=No
1d	Traces links <u>to a UCM component</u> must only be <u>from a GRL *actor*</u>
2a	Each <u>GRL intentional element</u> must have a Traces link <u>to a UCM map or responsibility</u> , unless tagged with Traces=No
2b	Traces links <u>from a GRL intentional element</u> must only be <u>to a UCM *map or responsibility*</u>
2c	Each <u>UCM map</u> must have a Traces link <u>from a GRL intentional element</u> , unless tagged with Traces=No
2d	Traces links <u>to a UCM map</u> must only be <u>from a GRL *intentional element*</u>
2e	Each <u>UCM responsibility</u> must have a Traces link <u>from a GRL intentional element</u> , unless tagged with Traces=No
2f	Traces links <u>to a UCM responsibility</u> must only be <u>from a GRL *intentional element*</u>
3a	Each <u>GRL task</u> must have a Traces link <u>to a UCM map or responsibility</u> , unless tagged with Traces=No
3b	Traces links <u>from a GRL task</u> must only be <u>to a UCM *map or responsibility*</u>
3c	Each <u>UCM map</u> must have a Traces link <u>from a GRL task</u> , unless tagged with Traces=No
3d	Traces links <u>to a UCM map</u> must only be <u>from a GRL *task*</u>
3e	Each <u>UCM responsibility</u> must have a Traces link <u>from a GRL task</u> , unless tagged with Traces=No
3f	Traces links <u>to a UCM responsibility</u> must only be <u>from a GRL *task*</u>
4c	Each <u>UCM component</u> (or one of its parents) must have a Traces link <u>from a GRL actor</u> , unless tagged with Traces=No
4d	Traces links <u>to a UCM component</u> (or one of its parents) must only be <u>from a GRL *actor*</u>



Questions and Future Work

- Have different types of links connecting GRL and UCM?
- Can UCM variables capturing the satisfaction levels of GRL intentional elements be used as traceability links?
- What recursive rules exploiting containment structures are beneficial?
- Should different rules apply to different parts/elements of a URN model?
- Can NLP or other approaches be used for creating (some) URN links automatically?
- How can jUCMNav's interface be made more usable (automatic link creation, quick fixes)?
- In addition to syntactic rules, should semantic rules and evolution rules (consistency in versions) be considered?
- Usability study, and standardization

jUCMNav 7.0.0 Available!



<http://softwareengineering.ca/jucmnav>

Thanks! Question?