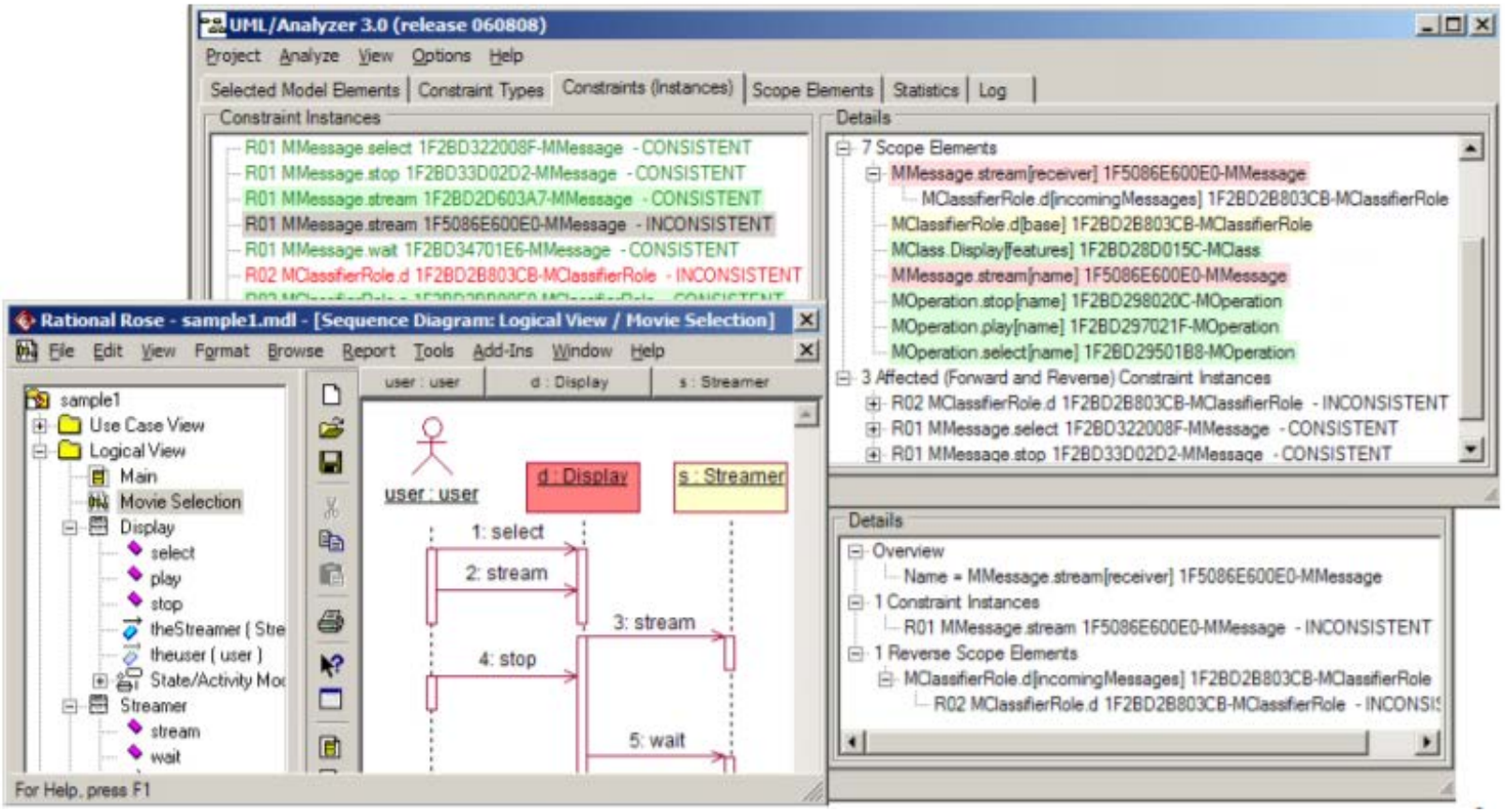
Generating and Evaluating Choices for Fixing Inconsistencies in UML Design Models

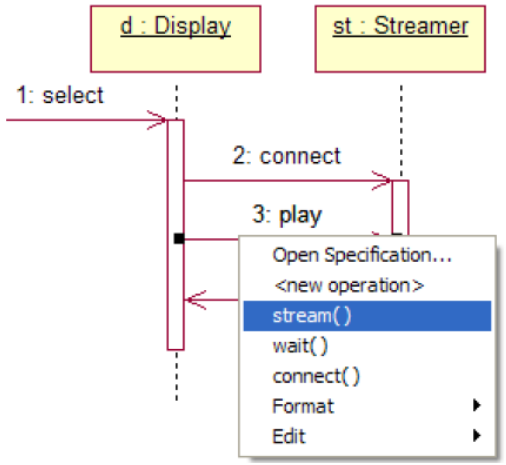
Alexander Egyed, Emmanuel Letier, and Anthony Finkelstein

CSC2125: Topics in Software Engineering

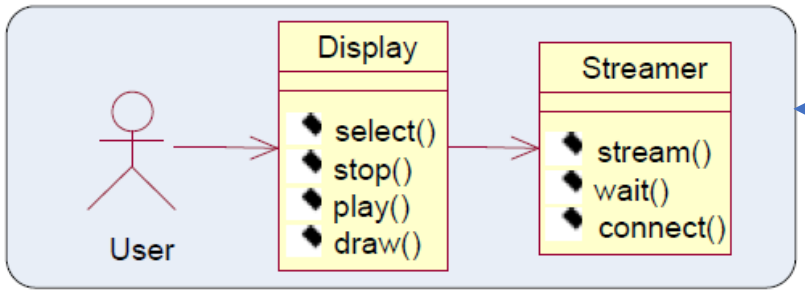
Or Aharoni

UML Model

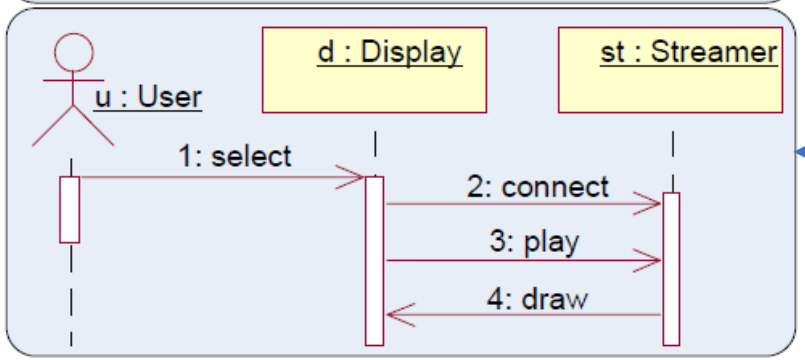

UML/Analyzer 3.0 (release 060808)
Project Analyze View Options Help
Selected Model Elements | Constraint Types | Constraints (Instances) | Scope Elements | Statistics | Log
Constraint Instances
R01 MMessage.select 1F2BD322008F-MMessage - CONSISTENT
R01 MMessage.stop 1F2BD33D02D2-MMessage - CONSISTENT
R01 MMessage.stream 1F2BD2D603A7-MMessage - CONSISTENT
R01 MMessage.stream 1F5086E600E0-MMessage - INCONSISTENT
R01 MMessage.wait 1F2BD34701E6-MMessage - CONSISTENT
R02 MClassifierRole.d 1F2BD2B803CB-MClassifierRole - INCONSISTENT
Details
7 Scope Elements
MMessage.stream[receiver] 1F5086E600E0-MMessage
MClassifierRole.d[incomingMessages] 1F2BD2B803CB-MClassifierRole
MClassifierRole.d[base] 1F2BD2B803CB-MClassifierRole
MClass.Display[features] 1F2BD28D015C-MClass
MMessage.stream[name] 1F5086E600E0-MMessage
MOperation.stop[name] 1F2BD298020C-MOperation
MOperation.play[name] 1F2BD297021F-MOperation
MOperation.select[name] 1F2BD29501B8-MOperation
3 Affected (Forward and Reverse) Constraint Instances
R02 MClassifierRole.d 1F2BD2B803CB-MClassifierRole - INCONSISTENT
R01 MMessage.select 1F2BD322008F-MMessage - CONSISTENT
R01 MMessage.stop 1F2BD33D02D2-MMessage - CONSISTENT
Rational Rose - sample1.mdl - [Sequence Diagram: Logical View / Movie Selection]
File Edit View Format Browse Report Tools Add-Ins Window Help
sample1
Use Case View
Logical View
Main
Movie Selection
Display
select
play
stop
theStreamer (Stre
theuser (user)
State/Activity Mo
Streamer
stream
wait
user : user
d : Display
s : Streamer
1: select
2: stream
3: stream
4: stop
5: wait
Details
Overview
Name = MMessage.stream[receiver] 1F5086E600E0-MMessage
1 Constraint Instances
R01 MMessage.stream 1F5086E600E0-MMessage - INCONSISTENT
1 Reverse Scope Elements
MClassifierRole.d[incomingMessages] 1F2BD2B803CB-MClassifierRole
R02 MClassifierRole.d 1F2BD2B803CB-MClassifierRole - INCONSIS
For Help, press F1


d : Display
st : Streamer
1: select
2: connect
3: play
Open Specification...
<new operation>
stream()
wait()
connect()
Format
Edit

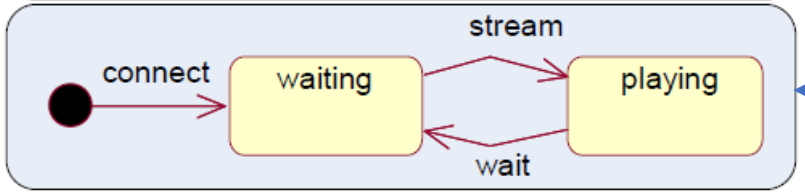
Video-on-demand (VOD) model in UML



Class diagram

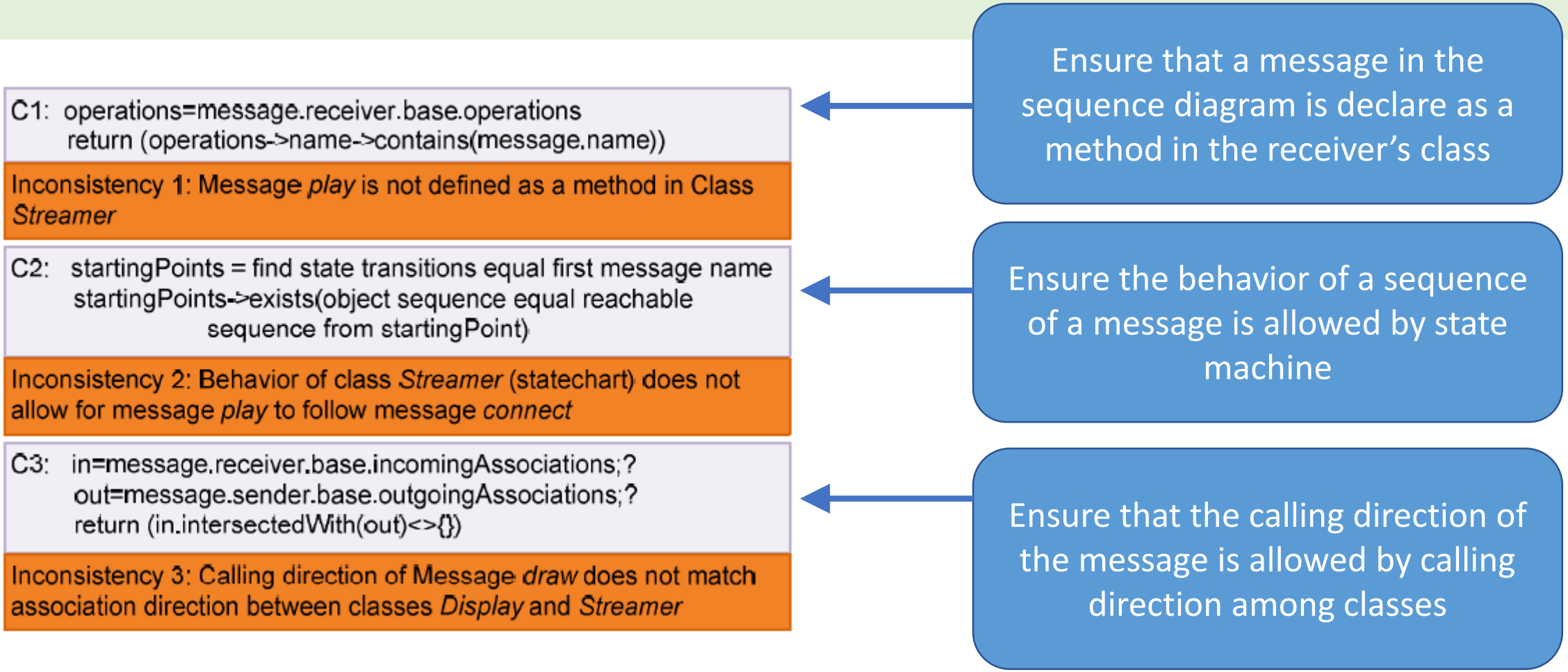


Sequence diagram

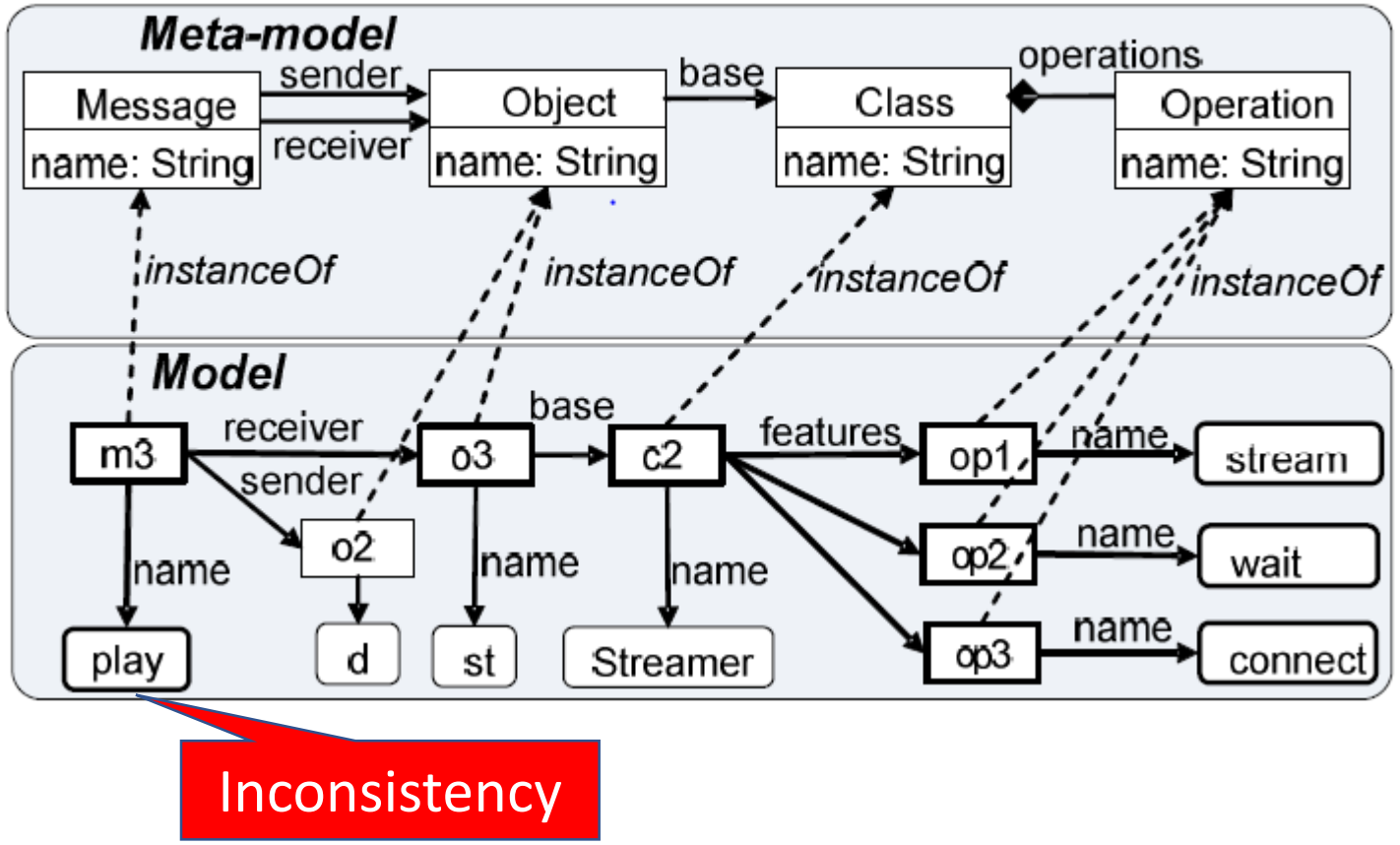


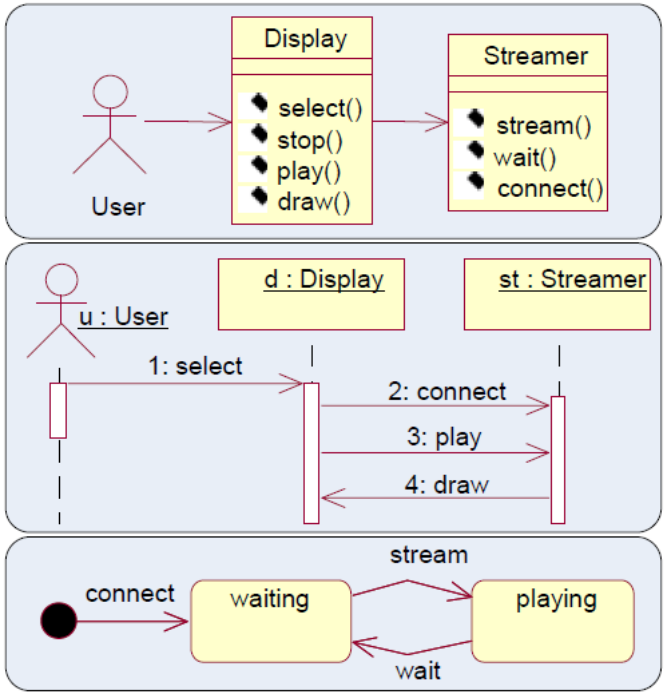
Statechart diagram
of the class Streamer

Consistency and inconsistencies Rules in UML



Inconsistencies Rules in UML from Meta-model





The Problem

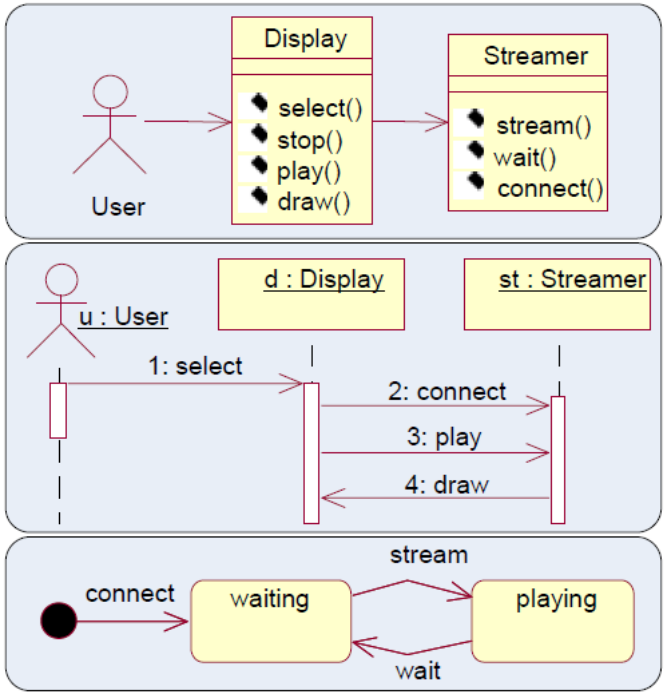
- Identify all potential locations where to fix this inconsistency.
- Identifying how to change that location.

- Fixing Rules for all Locations
- Fixing rule for Consistency Rules
- Consistency rules differ among users

The Problem: Fixing Rules for all Locations

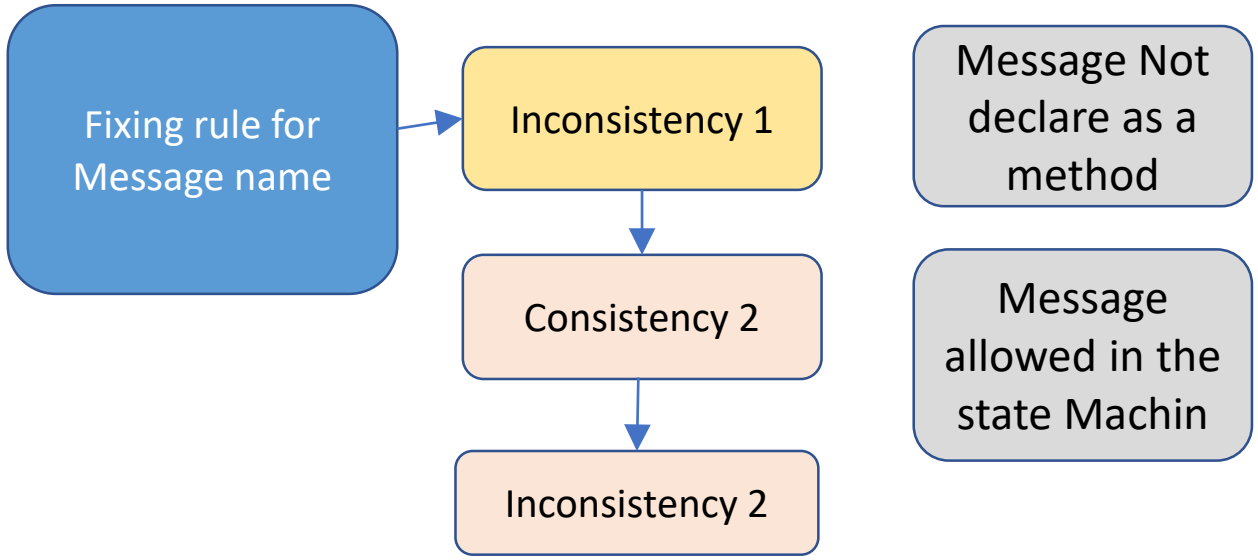
- Change method name
  - Message.Name
- Change the receiver of the message
  - Base.operation

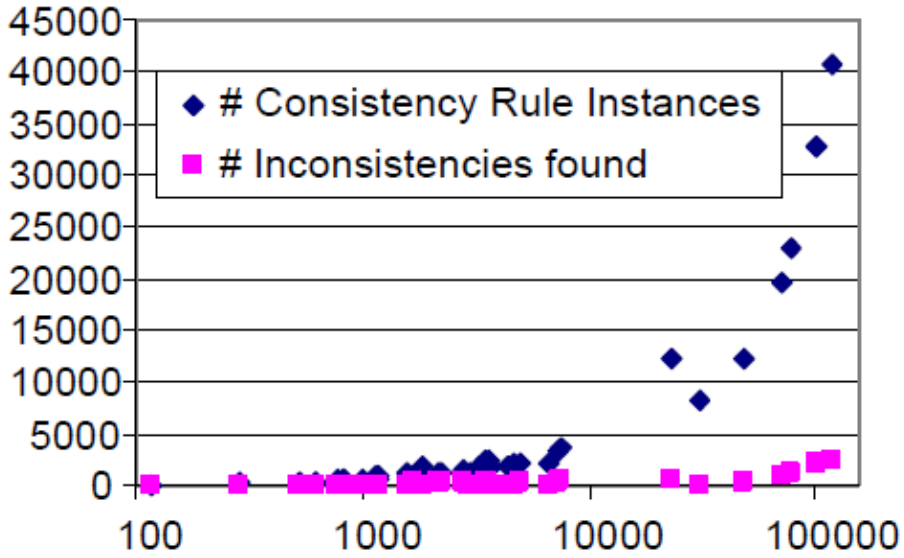
R1 4	message *stream*[name]	I-only
	message *stream*[receiver]	I-only
	object *d*[base]	C&I
	class *Display*[methods]	C-only
	method *select*[name]	C-only
	method *play*[name]	C-only
	method *stop*[name]	C-only



The Problem: Fixing rule for Consistency Rules

Interplay among multiple consistency rule

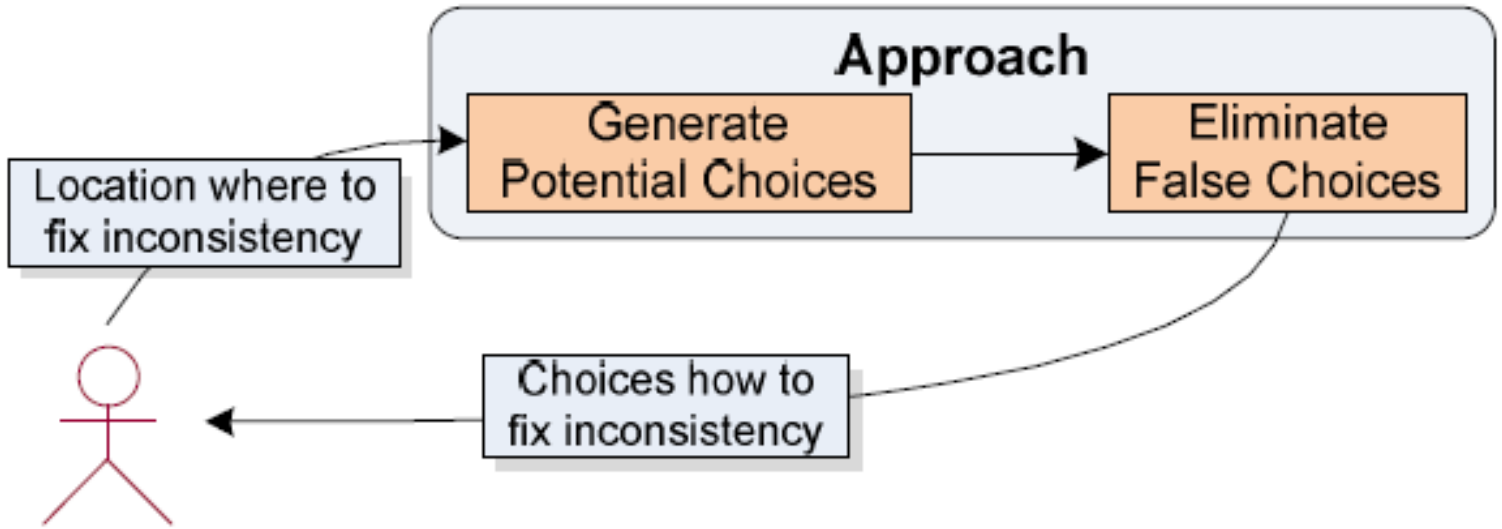




The Problem: Consistency rules differ among users

- Different designers often use different consistence rules
- Resolution rule define for one designer is useless for another

The Approach:



How we generate the initial set of choices?

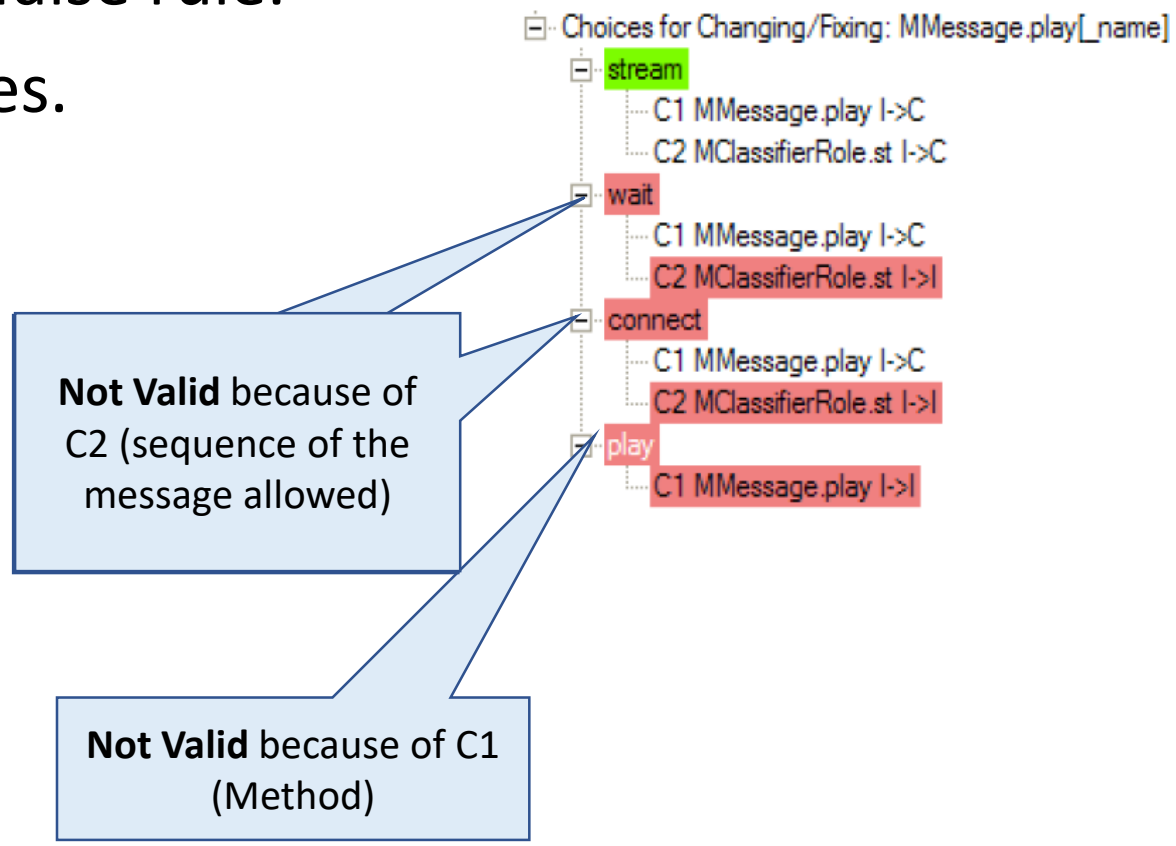
Choice Generation Functions:

- Generate all possible values that specific fields of a model element.
- Independent from the consistency rules

1	m:Message.receiver: choices = m.interaction.classifierRoles
2	m:Message.name: choices = {} foreach (method in m.receiver.base.methods) choices.insert(method.name)
3	ae:AssociationEnd.multipliciy choices = {1, 0..1, 1..n, 0..n}
4	c:Class.namespace choices = {} foreach (a in c.associations) foreach (oc in a.classifiers) choices.insert(oc.namespace)

What rules to re-evaluate to eliminate false choices?

- Choice generation function produce false rule.
- The elimination relay on the attributes.
- Check all the ways to fix the problem
  - Changing the receiver of the message
  - Changing the name of the message

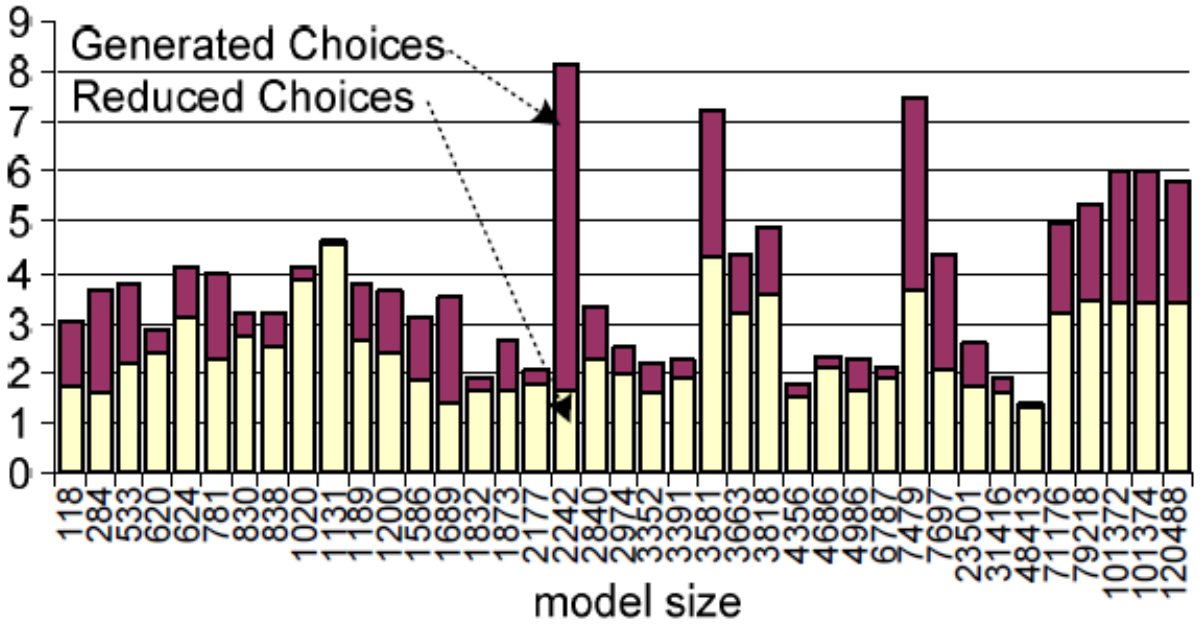


Impact of changes: Is the approach correct ?

- Only correct if it can identify all consistency rule instances affected by the fix
- One issue affects multiple elements
- Limitation of work is it restricted to single change

The Approach

- 39 small to large UML models
- 24 types of consistency rules found in industry.
- 14 types of typical locations for fixing inconsistencies.
- Only 17% of all relevant locations were evaluated



The Approach

- False location is a location for which no valid choice exists, bar suggest 11.2%
- By exploring choices we are able to automatable detect these false locations

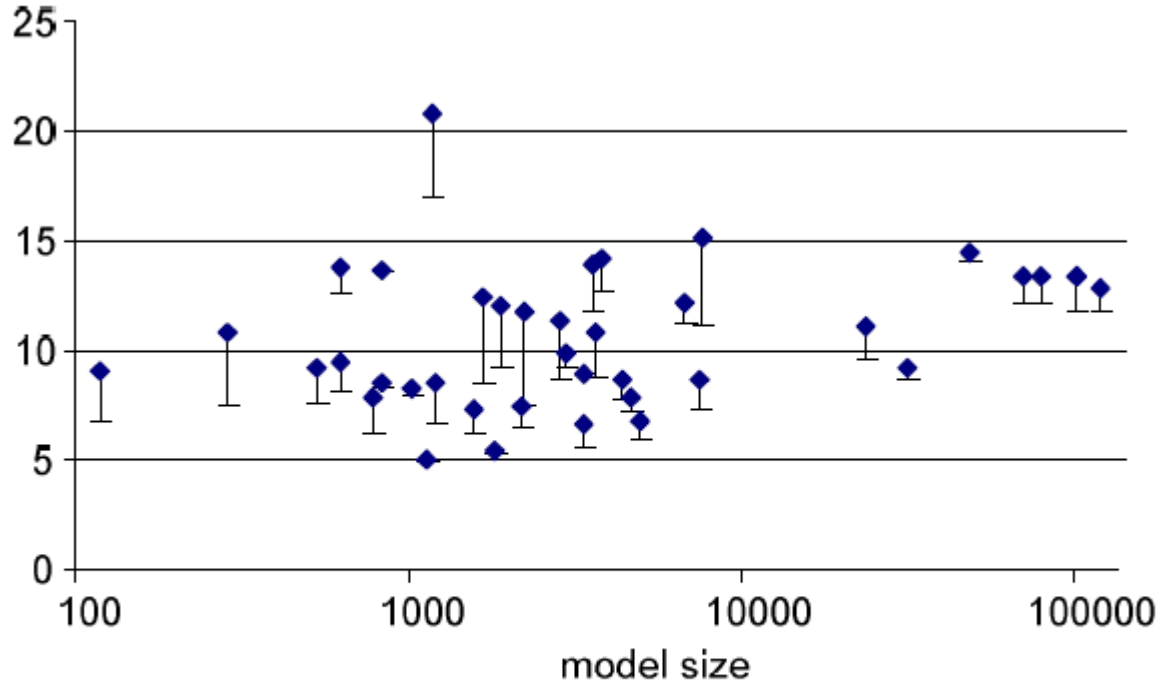


Figure 10. Number of Locations for Fixing Inconsistencies (with/without false ones)

Conclusion

Strength of the approach

- Approach does not suggest false choices
  - Able to identify all consistency rules affected by the rules automatically
- Use a white-box constraints.
- The approach was made on UML, how ever it can be transfer to other meta-models.
- The approach tool provide “On-line” suggestions of choices.
- 40% had a single valid choice
  - Possible automated correction

Weaknesses of the approach

- Technique can make a change in one location at a time
- The approached was checked on IBM Tool only .
- The approach tested on small group of models.

Discussion

- Do you think in the future there will be possible auto-correction?
- What rollback mechanism shell be put ? If any?
- Do you think that we need to have logs in the system ?
- Do you think the approach be develop from the code to the model? (Like MVVM)