
UiO : Universitetet i Oslo

universidad SANJORGE GRUPO SANVALERO

# Leveraging Models at Run-time to Retrieve Information for Feature Location

Lorena Arcega, Jaime Font, Øystein Haugen, Carlos Cetina

International Workshop on Models@run.time

29th September 2015, Ottawa, Canada

# Agenda

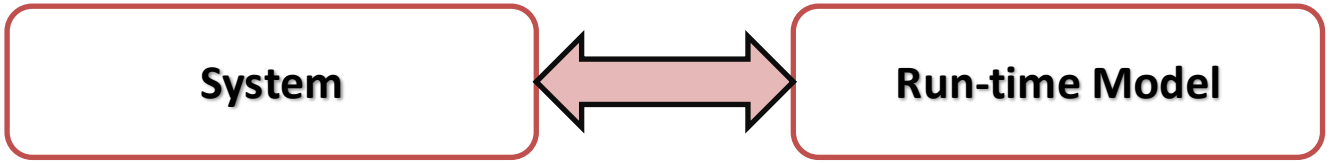
- Introduction
- Goal to achieve
- Solution proposed
- The Smart Hotel
- Feature Location with Models at Run-time
- Discussion
- Conclusions

# Introduction

- Variability extraction
  - Separate out variance from source
- Feature location
  - Where is a piece of abstract variability located in source code?
- These activities are important and common in software maintenance and evolution

# Introduction

- Models at run-time
  - define a connection between the system and the run-time model

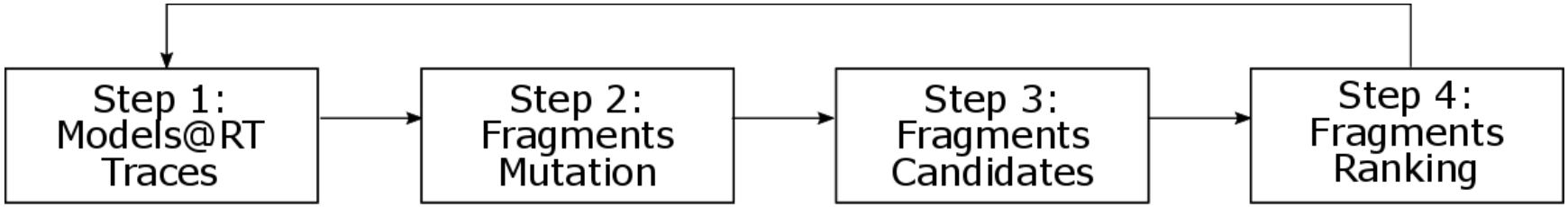


# Goal to achieve

**Perform feature location by model executions**

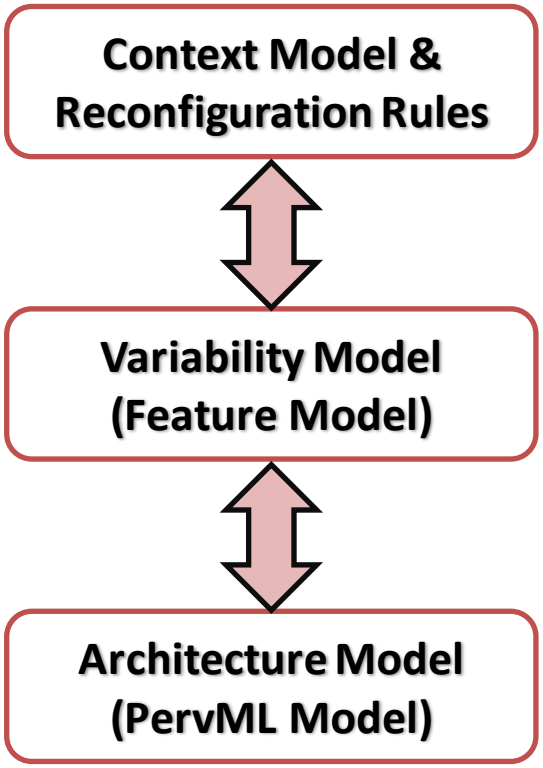
# Solution proposed

- The feature location process



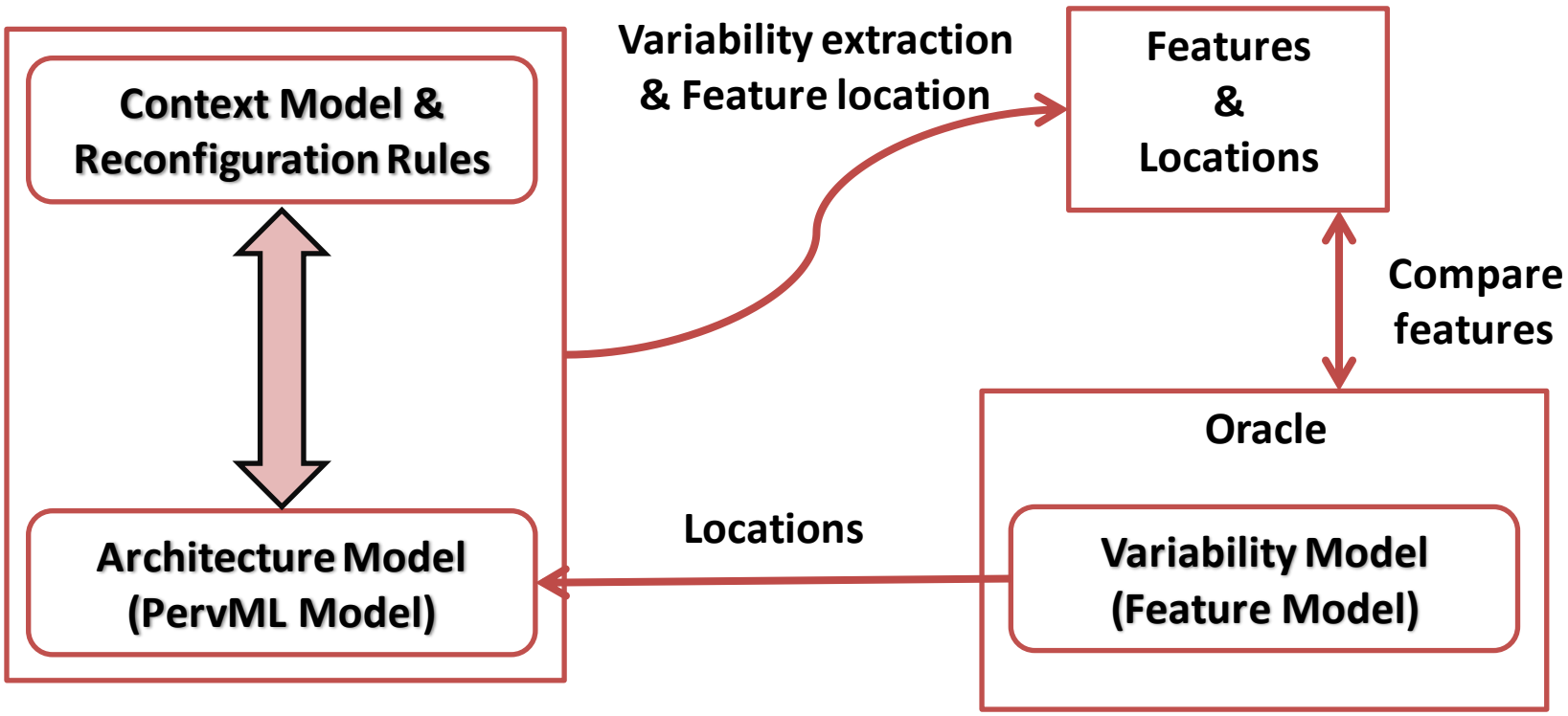
# The Smart Hotel

- Existing Smart Hotel definition

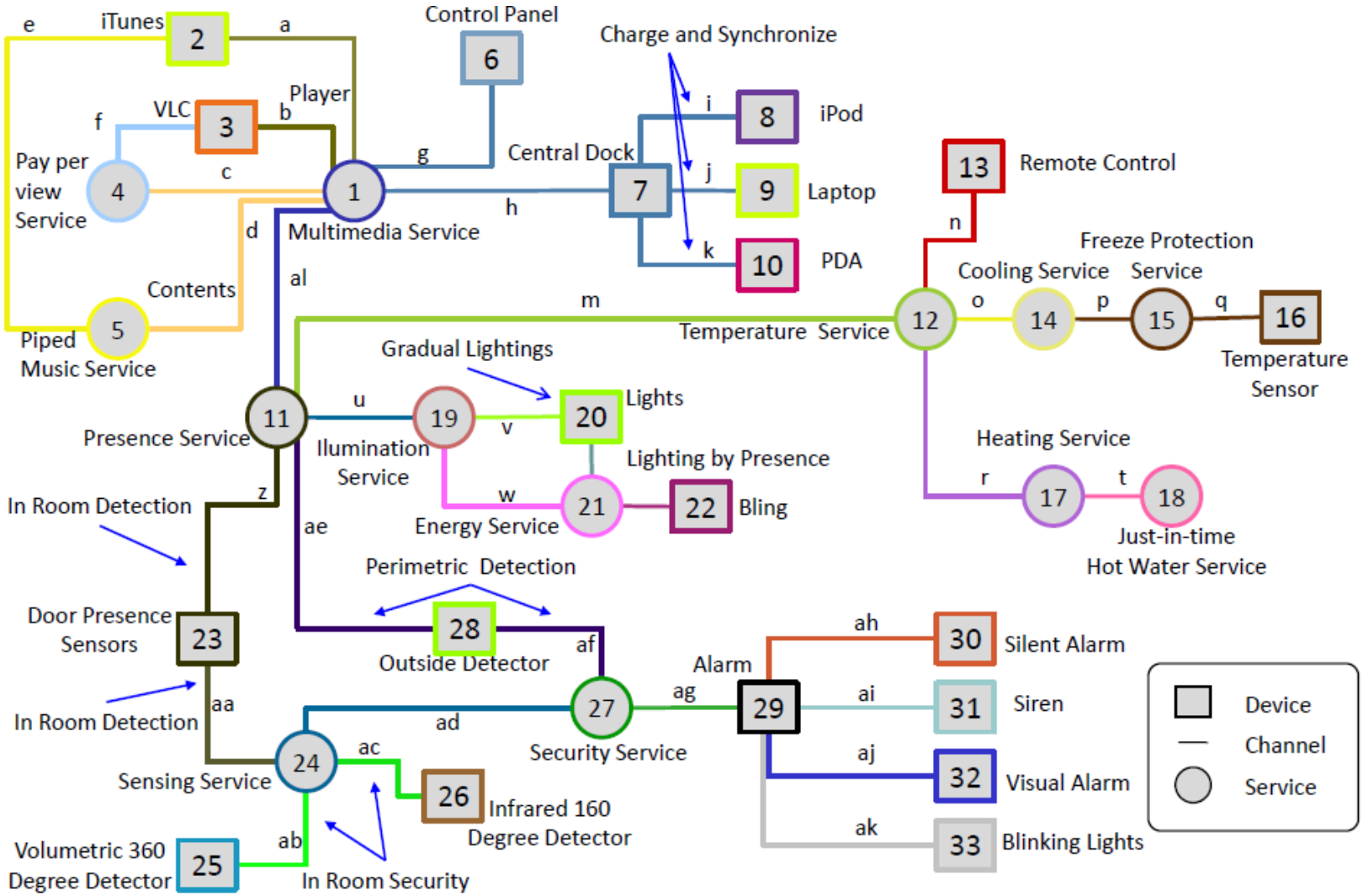


# The Smart Hotel

- Current Smart Hotel

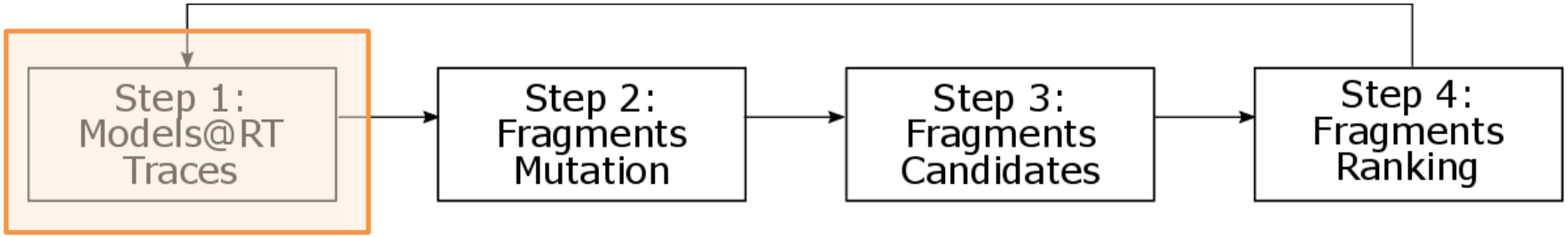


# The Smart Hotel



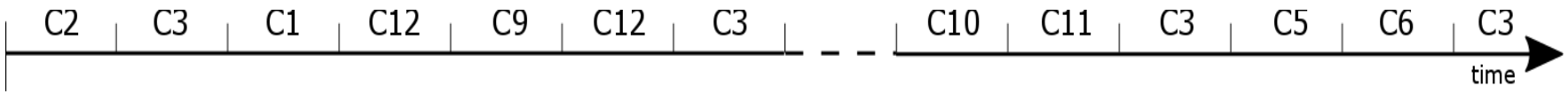
# Step 1
# Models@RT Traces

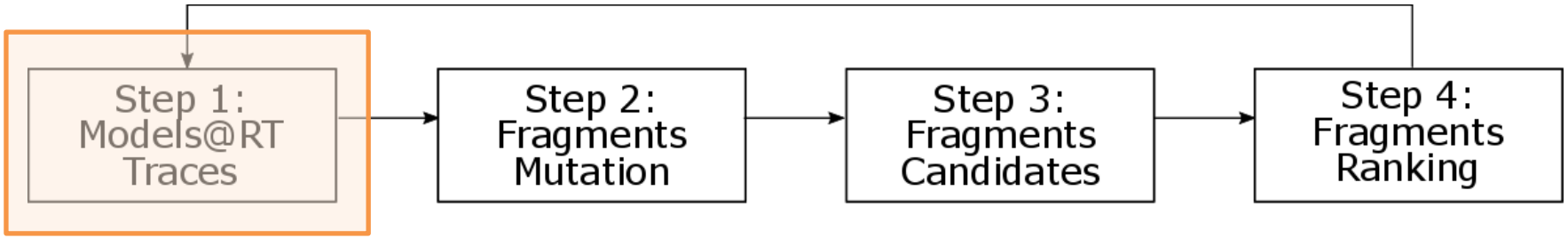
- Gets the trace of configurations resulting from a system that has been running for a specified time



# Step 1 in the Smart Hotel

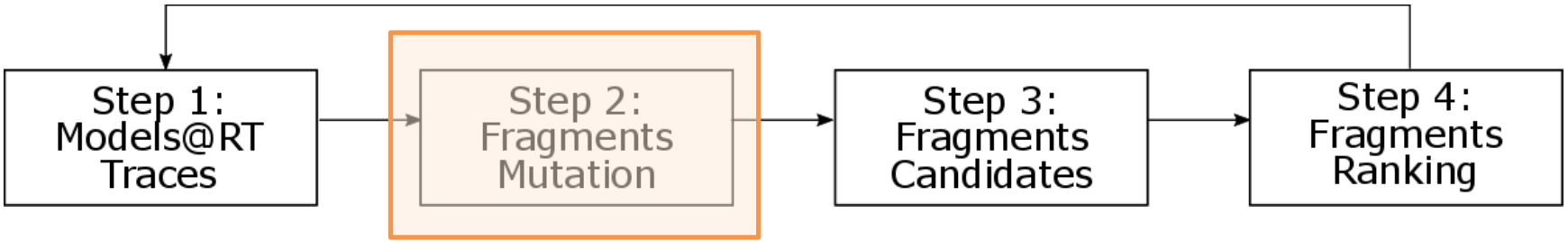
- Thirty reconfigurations
- Twelve different configurations



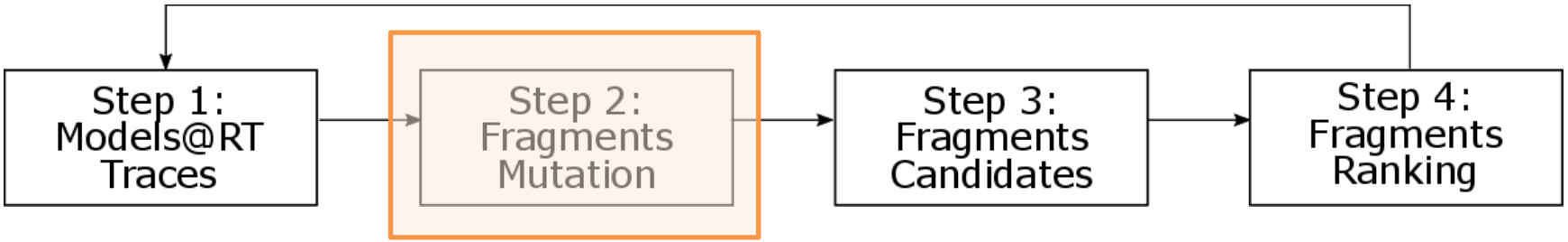
# Step 2
# Fragments Mutation

- A fragment is any part of the architecture model
- The software engineer decides which feature to locate (target feature)



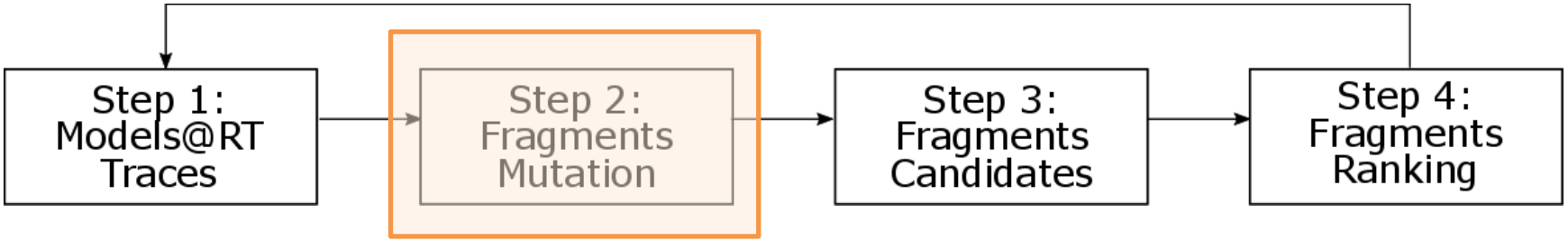
# Step 2
# Fragments Mutation

- Seed: The software engineer defines a seed fragment that he believes is associated with the target feature
- The step performs automatic mutations of the model fragment designated as seed



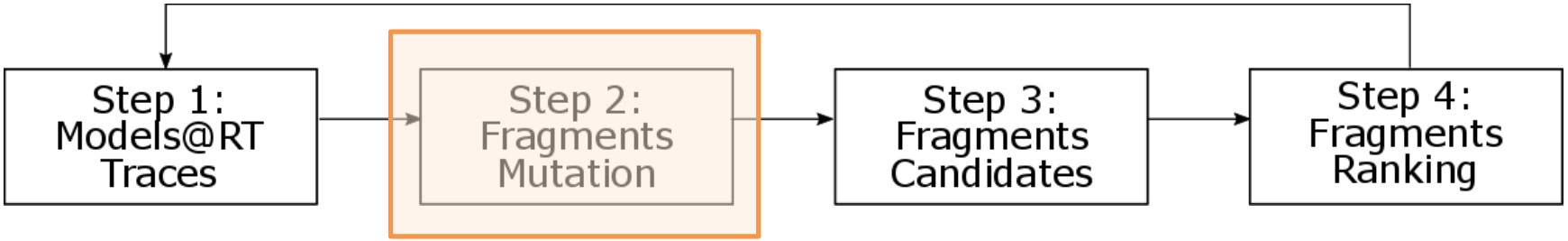
# Step 2
# Fragments Mutation

- Taking the seed fragment model as starting point, some model elements are added to or removed from the seed model fragment



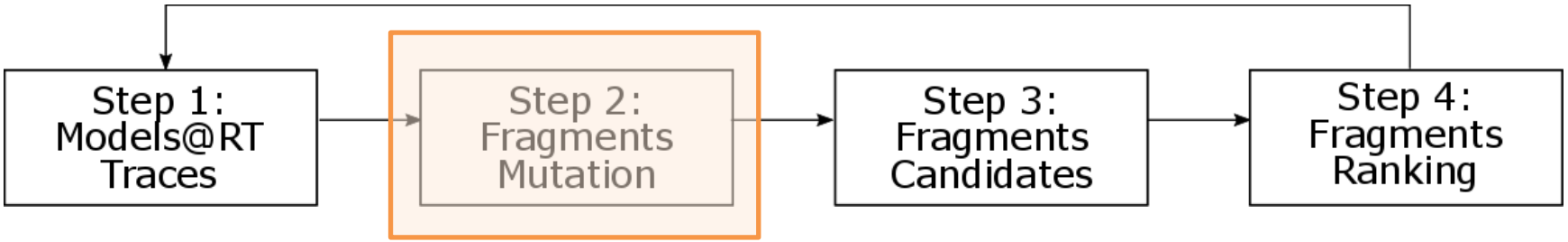
# Step 2
# Fragments Mutation

- The result is a set of fragments that are variations of the seed fragment

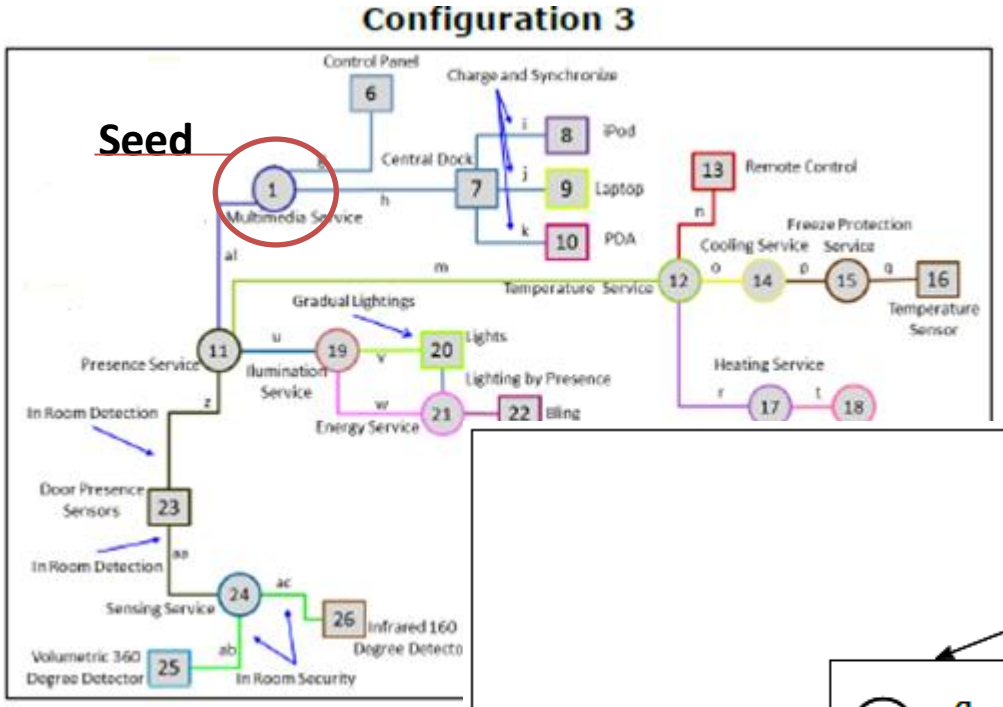


# Step 2 in the Smart Hotel

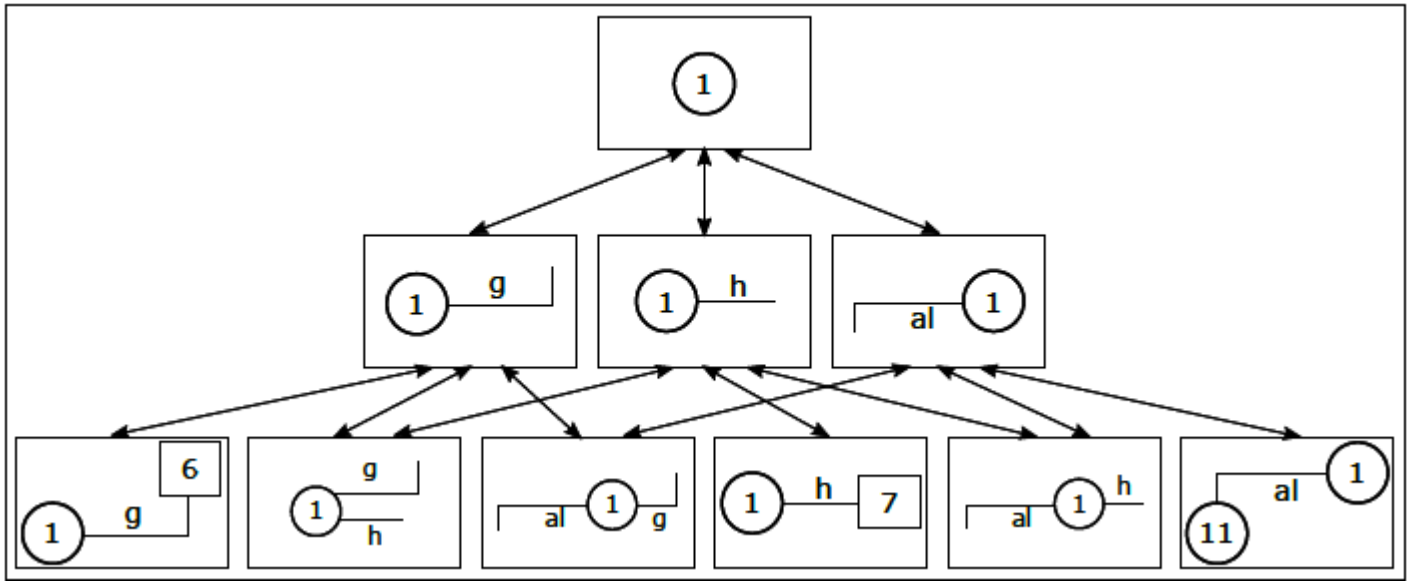
- Target feature: related to Multimedia services
- Seed: Multimedia Service
- Mutations for every configuration (12)



# Step 2 in the Smart Hotel

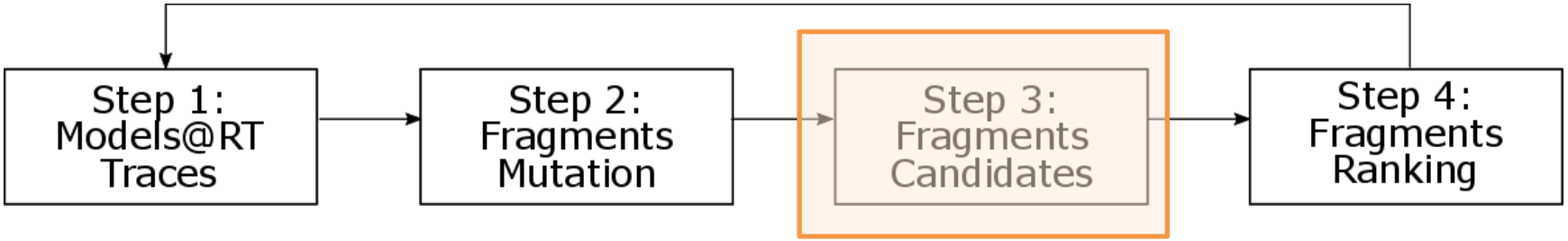



**Mutations from the Seed**



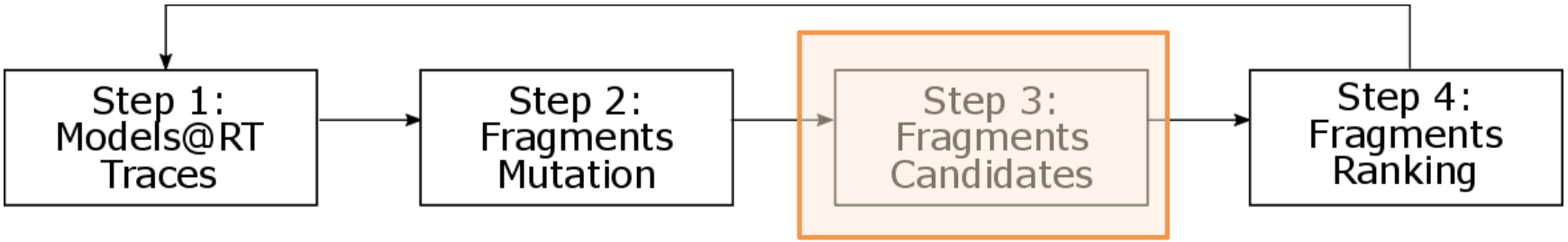
# Step 3
# Fragments Candidates

- This step assesses each fragment obtained in the second step

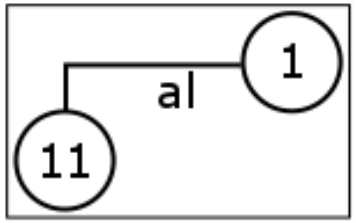


# Step 3
# Fragments Candidates

- The assessment of each fragment candidate depends on:
  - the configurations in which the fragment appears
  - the number of times that these configurations appear in the trace

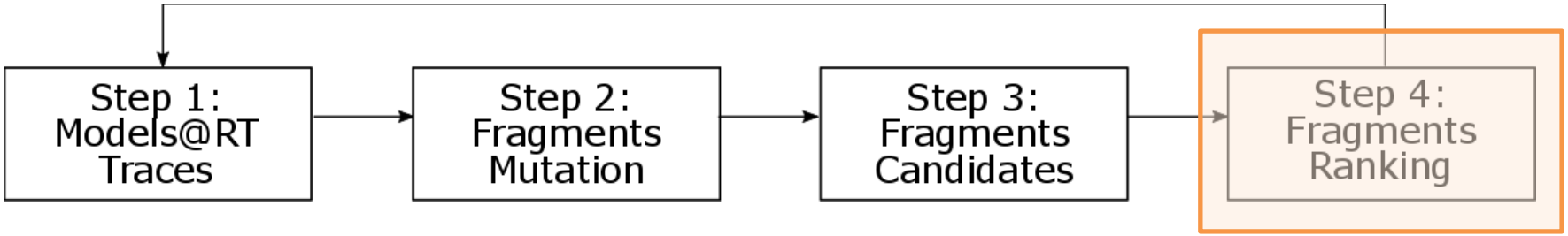


# Step 3 in the Smart Hotel

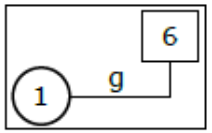
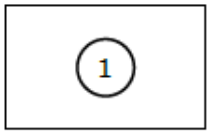
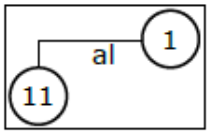
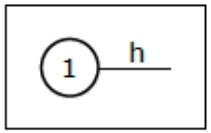
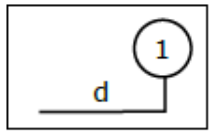
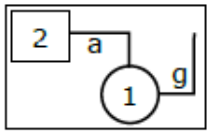
| Model fragment | Ocurrences in configurations | Ocurrences in reconfigurations |
|---|---|---|
|   | 8/12 | 19/30 |

# Step 4
# Fragments Ranking

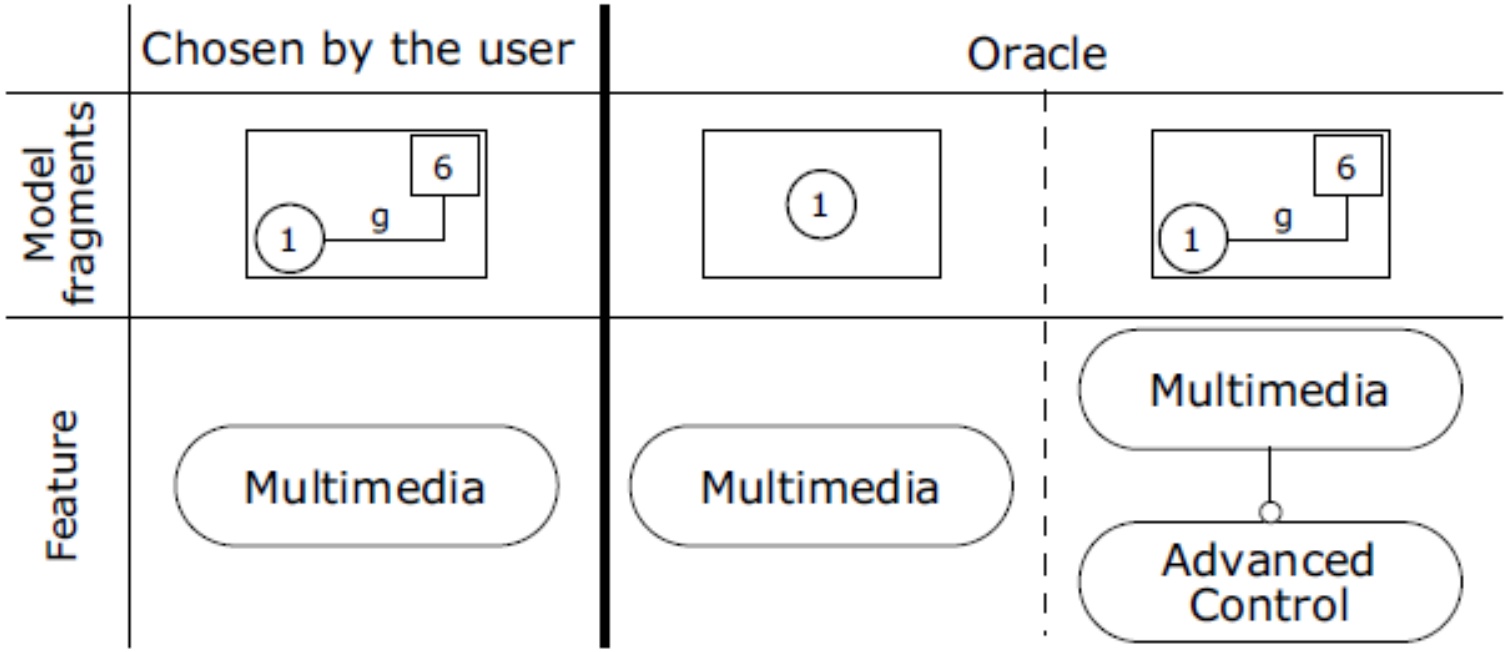
- The fragments are ordered in a ranking taking into account the values obtained in the previous step



# Step 4 in the Smart Hotel

| Model fragments |  |  | ... |  |  | ... |  |  |
|---|---|---|---|---|---|---|---|---|
| Ocurrences in configurations | 10/12 | 10/12 | ... | 8/12 | 7/12 | ... | 3/12 | 1/12 |
| Ocurrences in reconfigurations | 25/30 | 25/30 | ... | 19/30 | 19/30 | ... | 9/14 | 4/14 |

# Feature Location Validation



# Discussion

- The right model fragment has the same values in the ranking as the fragment selected
- Both appears in the top part of the ranking

# Discussion

- Improvements in our algorithm:
  - Heuristic to determine how many mutations levels are needed
  - Restrictions in the mutations of the fragments to limit the fragments that can appear in the mutations

# Discussion

- Improvements in the feature location:
  - Using a longer trace or a trace containing more relevant information
  - Extracting data from the transitions between the configurations

# Conclusion

- Our process retrieves information from the run-time models to perform feature location in reconfigurable systems

- Our validation shows some preliminary results of how the models at run-time can generate useful information at model level