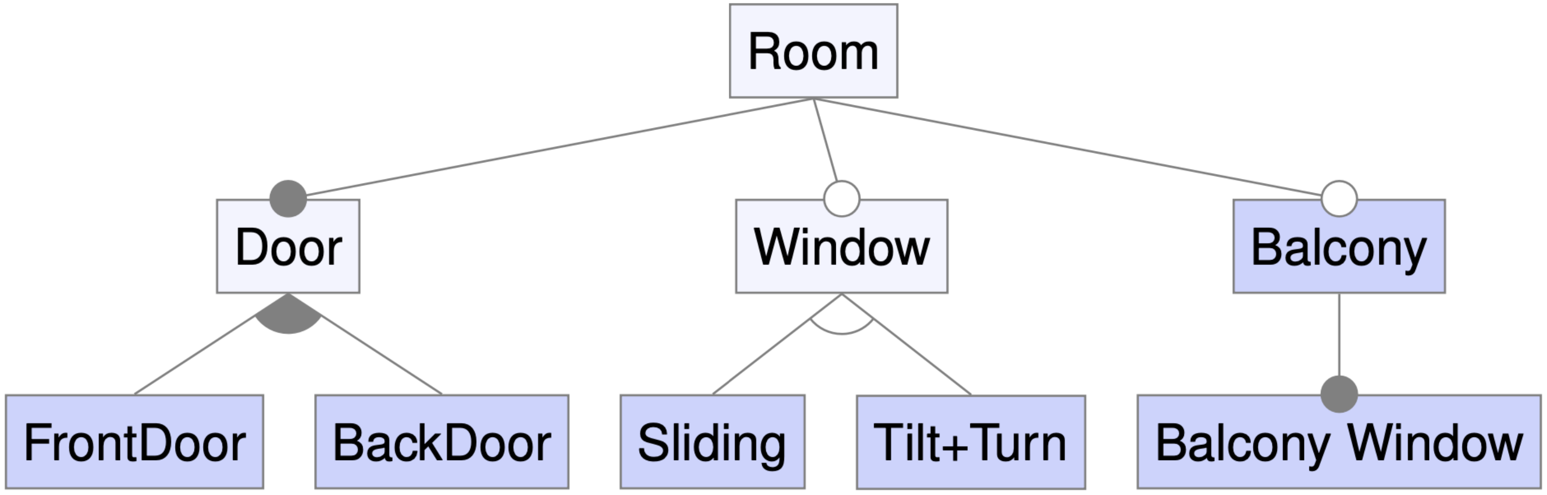
Karlsruhe Institute of Technology

Universität Siegen

Variant Prioritization for Testing Using Solution-Space Knowledge

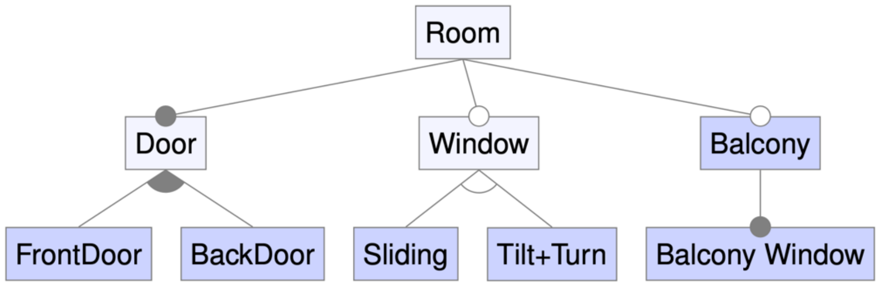
Lukas Güthing, Mathis Weiß, Malte Lochau, Ina Schaefer,
Kathrin Leonie Schmidt, Morten Harter

Complex Configurable Systems

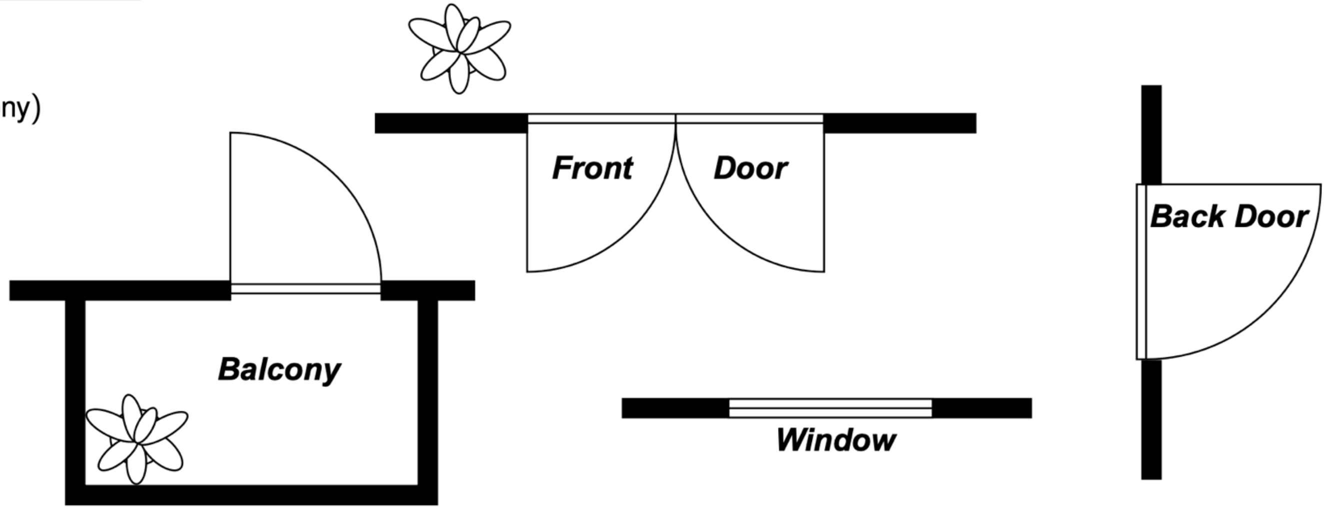


$\neg(\text{BackDoor} \land \text{Balcony})$

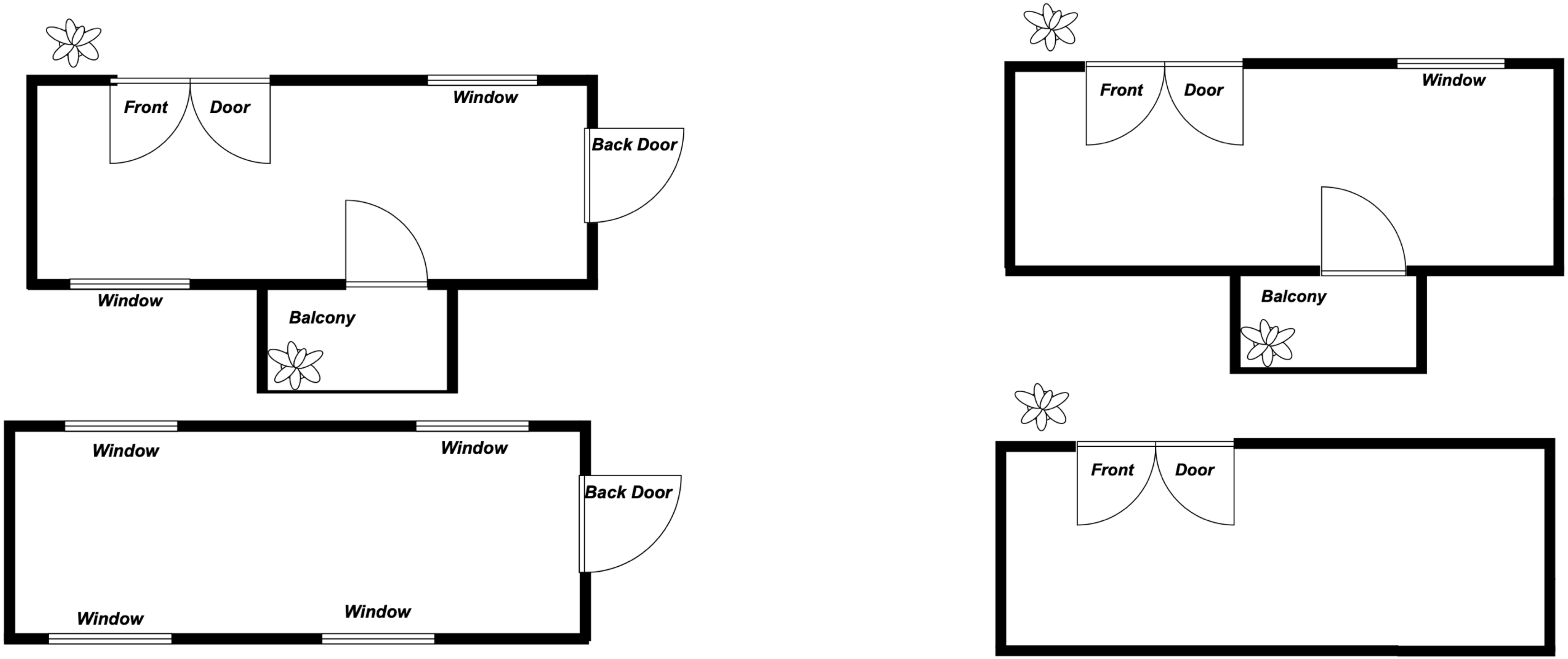
Complex Configurable Systems



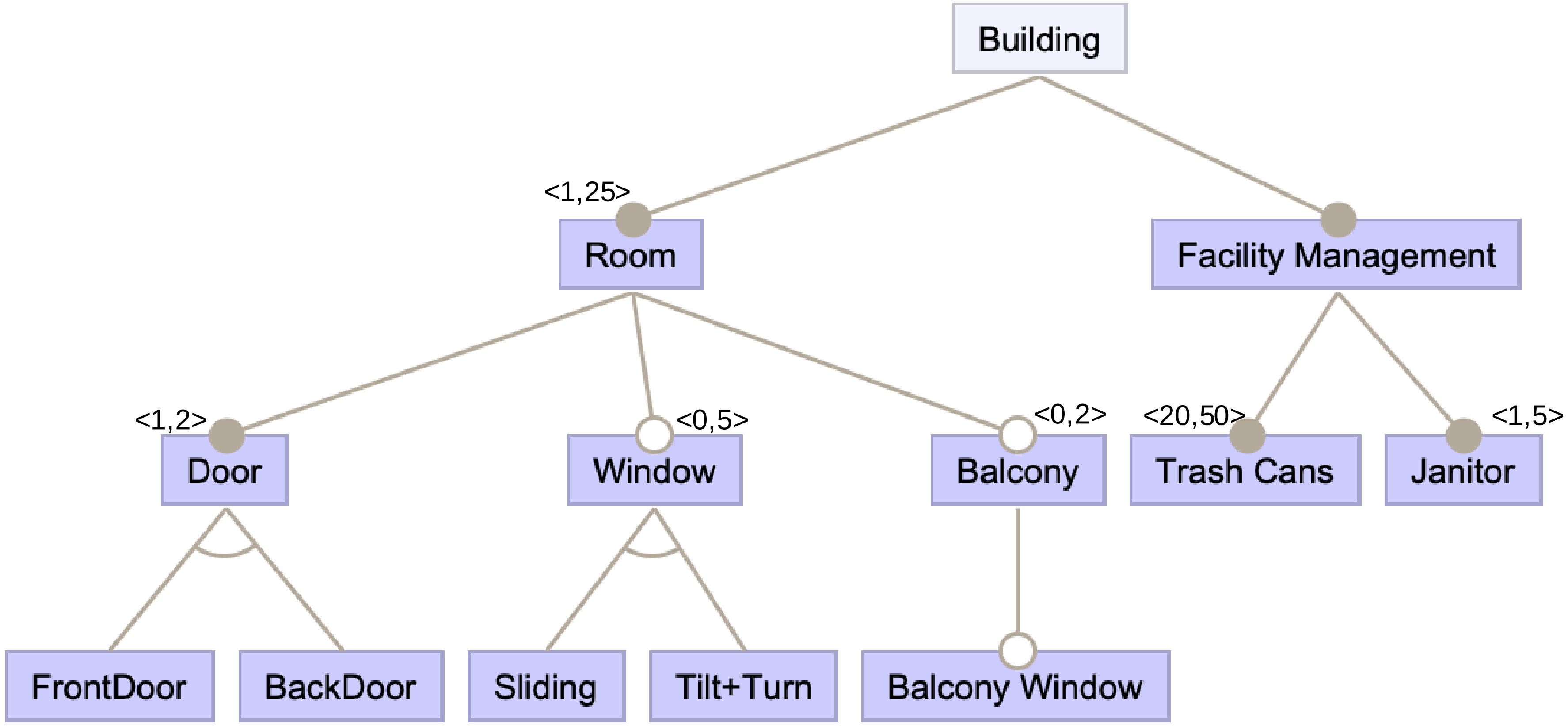
¬(BackDoor ∧ Balcony)



Complex Configurable Systems

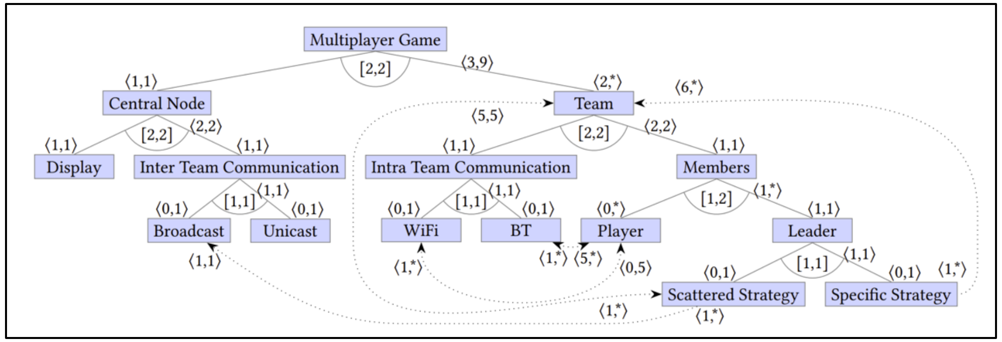


Complex Configurable Systems



Room<10,25> ⇒ Trash Cans<35,50>
Room<15,25> ⇒ Janitor<3,5>

Tool: CFM-Toolbox



Visualization

Analyses

Modification & Sampling

Translation & Parsing

Configuration

Motivation: Problems in Testing

- Single-variant systems are already faulty
  - Tests to find all/most faults

- Interactions between features introduce additional faults
  - Not present in every variant

- Which subjects to test?
  - All?
    - #configurations exceeds realistic ability to test
  - Which subset?

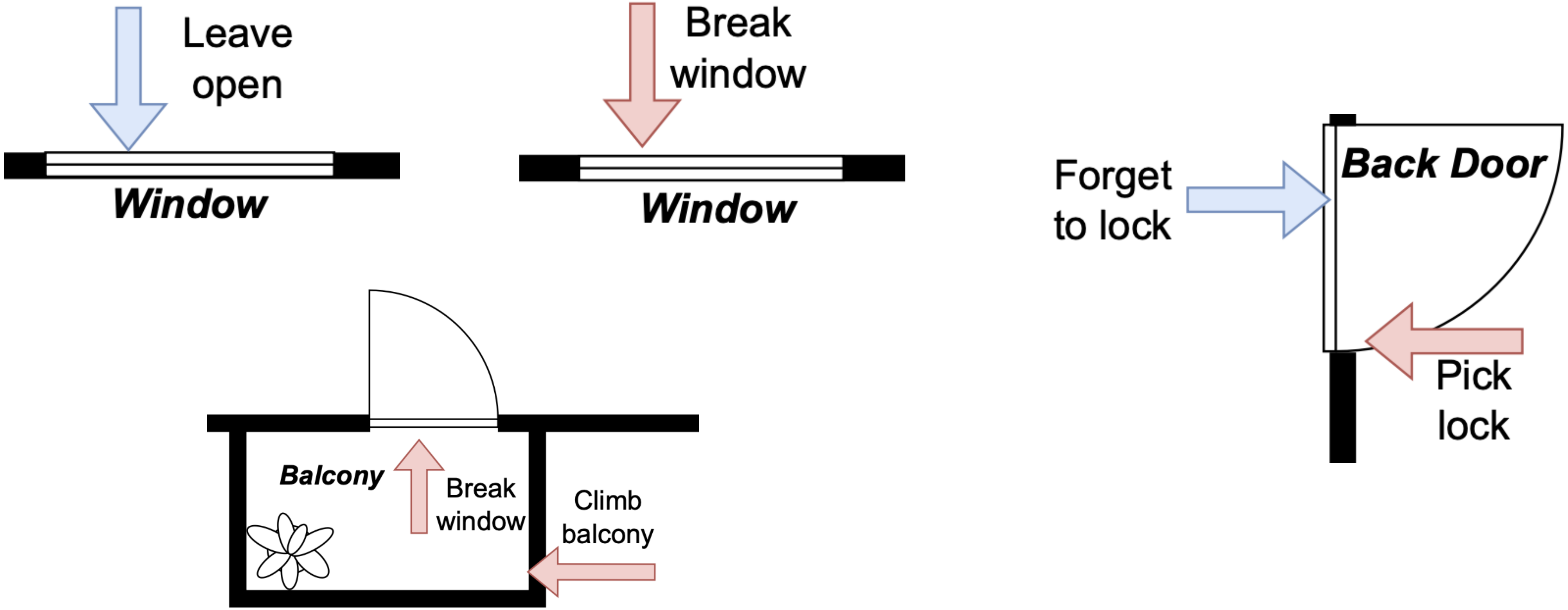
⇒ Prioritization needed

Solution: Solution-Space Prioritization

- Leverage system knowledge to find variants with
  - High risks of failure
  - Security flaws
  - Safety flaws

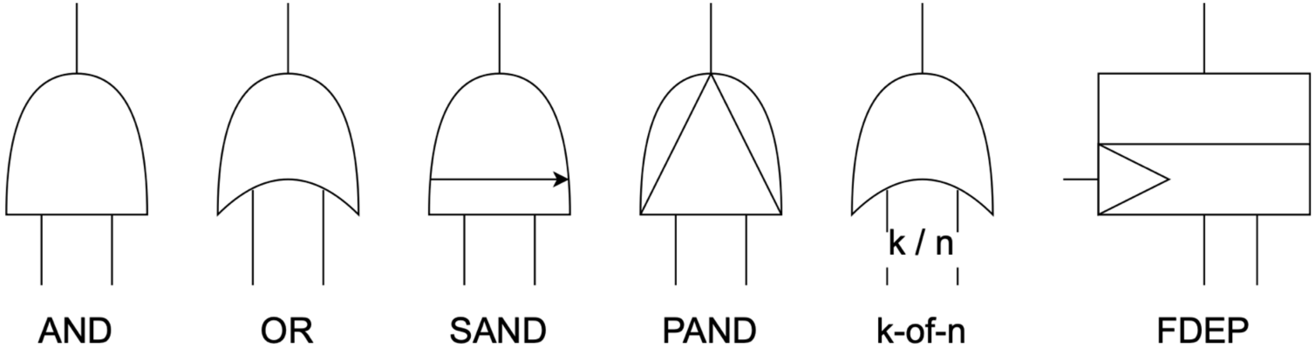
- Suitable solution-space models
  - Source code
  - Realizability mappings
  - Hazard/risk/threat models
  - Behavioral models

Attack-Fault-Trees



Attack-Fault-Trees

Gates:



Events:

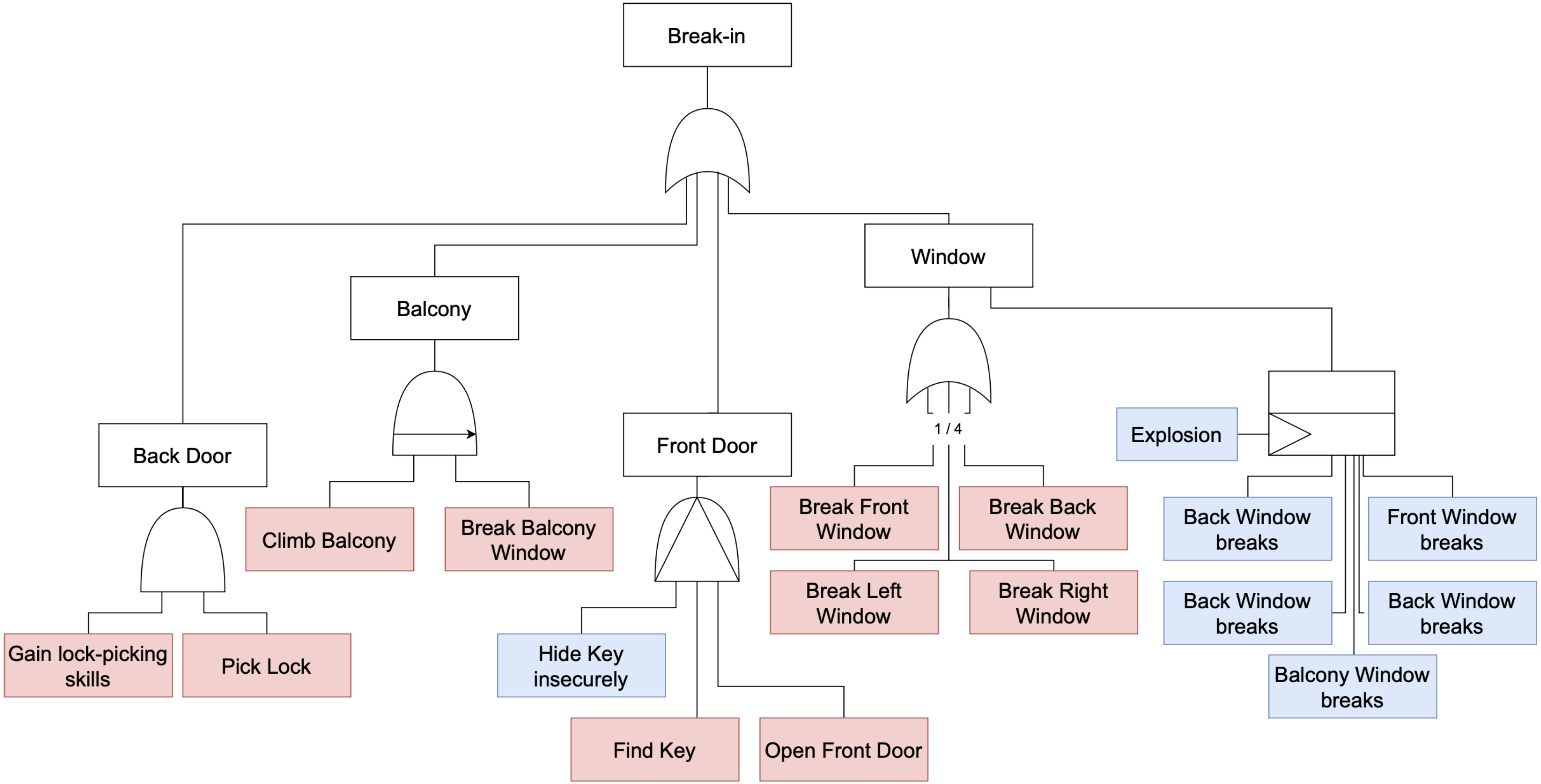
Pick Lock

Attack

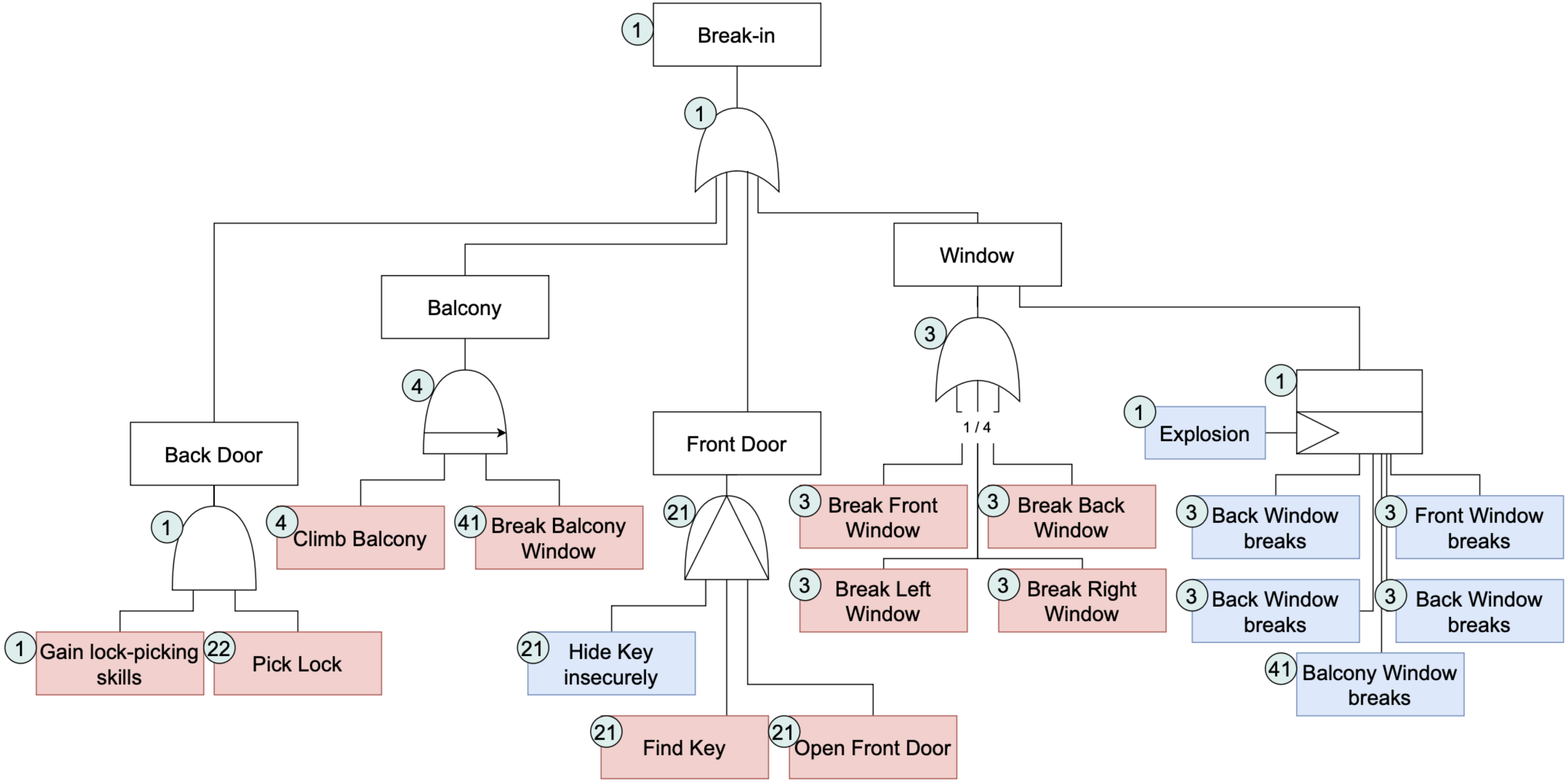
Window breaks

Fault

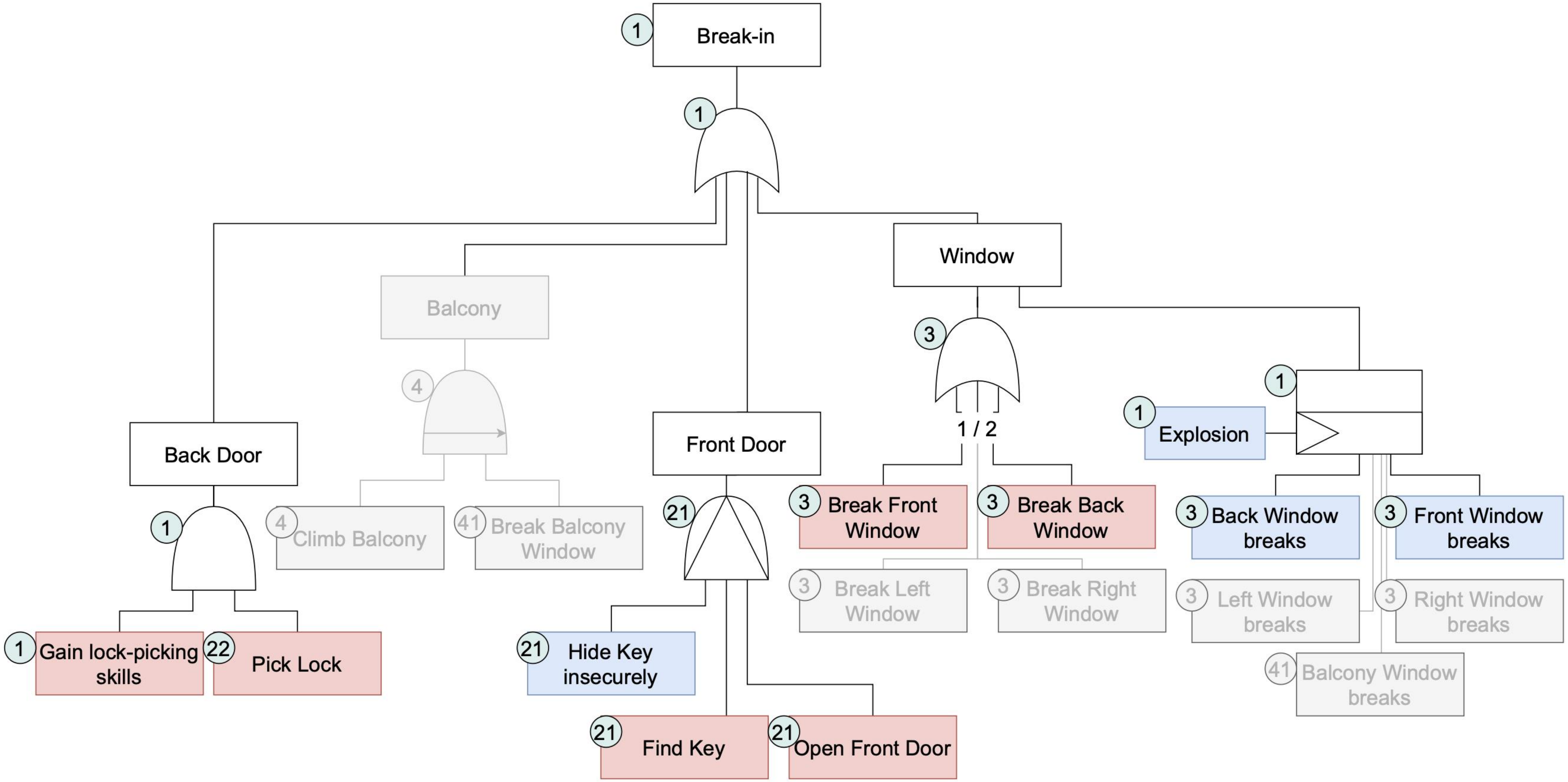
Attack-Fault-Trees



Variability-Aware Attack-Fault-Trees



Variability-Aware Attack-Fault-Trees



Analyses on VAFTs

- Product-based
  - Derive product AFT from VAFT
  - ”Classical” AFT analyses

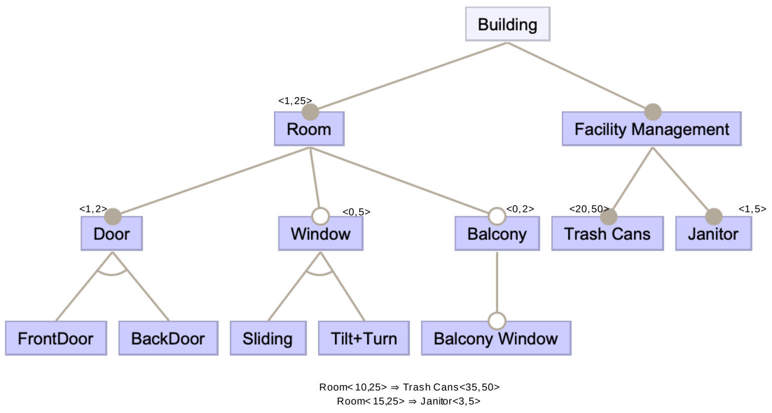
- Family-based
  - Find min/max configurations
    - TTF
    - Failure risk
  - Find volatile features
  - Find high-risk interactions

Analyses on VAFTs – Ongoing Work

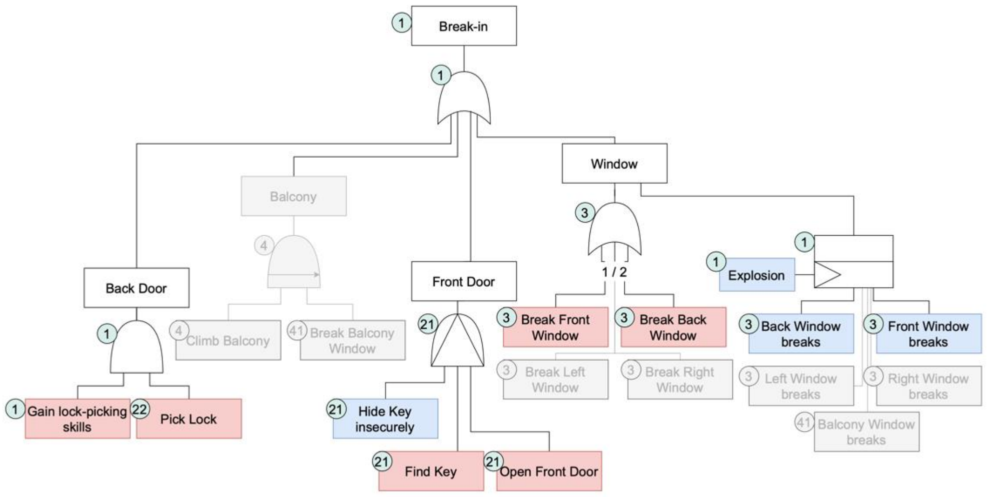
- Current status:
  - Family-based analyses are agnostic of FM
  - 150% model gets analyzed “as a variant”

- Family-based – FM-aware
  - Find min/max configurations
    - TTF
    - Failure risk
  - Find volatile features
  - Find high-risk interactions

Complex Configurable Systems



Variability-Aware Attack-Fault-Trees



Solution: Solution-Space Prioritization

- Leverage system knowledge to find variants with
  - High risks of failure
  - Security flaws
  - Safety flaws
- Suitable solution-space models
  - Source code
  - Realizability mappings
  - Hazard/risk/threat models
  - Behavioral models

Analyses on VAFTs – Ongoing Work

- Current status:
  - Family-based analyses are agnostic of FM
  - 150% model gets analyzed “as a variant”
- Family-based – FM-aware
  - Find min/max configurations
    - TTF
    - Failure risk
  - Find volatile features
  - Find high-risk interactions