Assessing Modularity of Feature Models with ACNs

Kanwarpreet Sethi, Sunny Huynh
Yuanfang Cai
Drexel University

FOMDD and Modularity

- Feature-Oriented Software Development
- Model Driven Software Development
- Software Modularity

FOMDD and Modularity

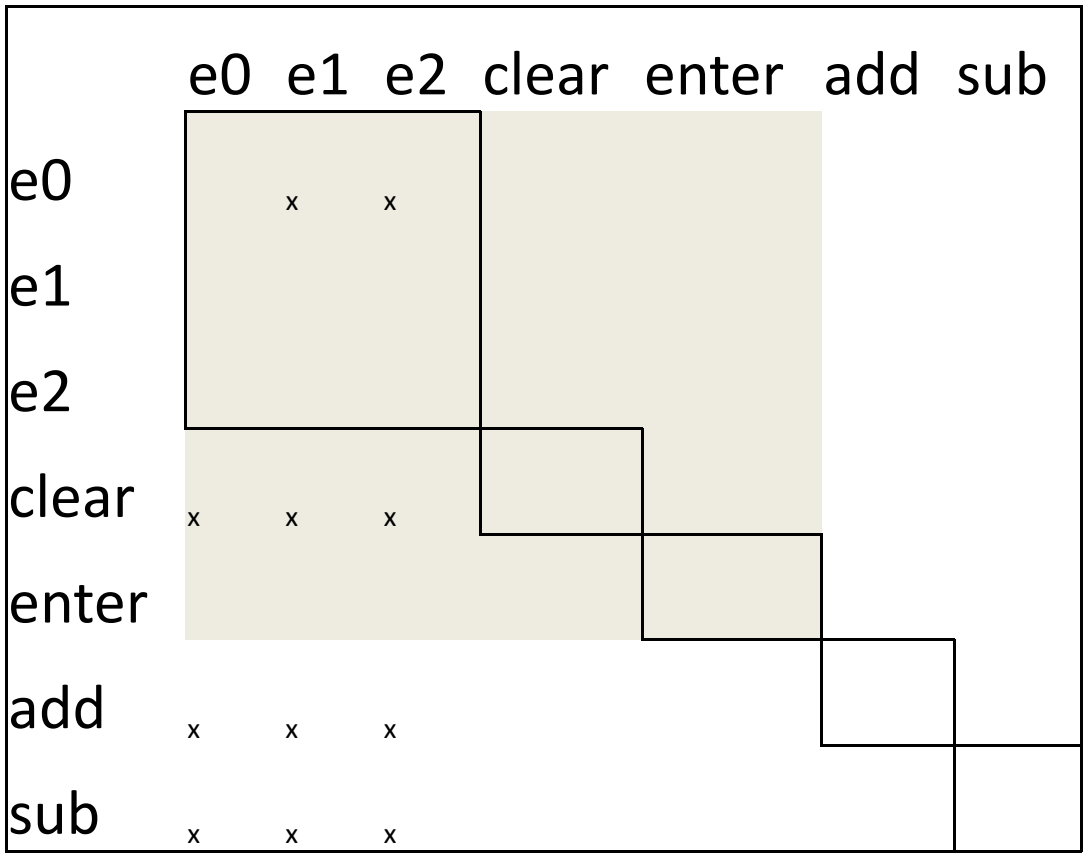
- Model and assess the transformation of FOMDD, instead of source code modularity
  - after implementation, the code that was cloned, shared or extended become undistinguishable at the source code level.
  - Important decisions became implicit
    - which parts represent commonality (should remain stable)
    - which parts reflect variations that can be changed independently
- Higher-level model
  - more abstract
  - smaller in scale

Design Rule Theory

- Design Rules
  - Stable Design Decisions
  - Decouple otherwise coupled decisions and generate modules
- Modules
  - Independent task assignment
- FOMDD and Design Rule Theory
  - Commonality: design rules
  - Variations: independent modules

Our approach

- Using ACN to model the transformation
  - An example
    - A small calculator:
      - Base module: e0, e1, e3, clear(), enter()
      - Two features: add, sub
      - Cal1 = base• add
      - Cal2 = base• add • sub
    - A lot of decisions embedded
- Modeling using augmented constraint network
  - Base ACN: clear => e0 && e1 && e2
  - ADD ACN : add => e0 && e1 && e2
  - Sub ACN: sub => e0 && e1 && e2
- ACN operations
- DSM generation


e0 e1 e2 clear enter add sub
e0
e1
e2
clear
enter
add
sub
x x
x x x
x x x
x x x