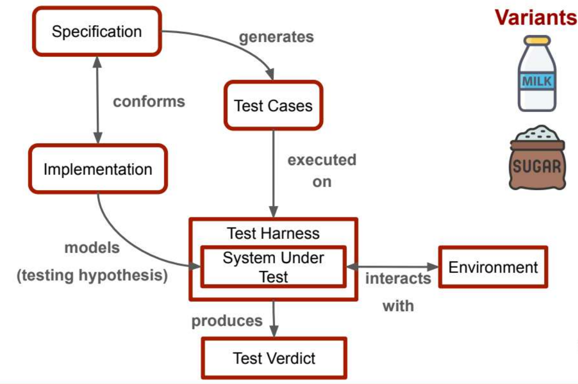
TiCToC: Testing in Times of Continuous Change

Lars van Arragon, Gijs van Cuyck, Tannaz Zameni

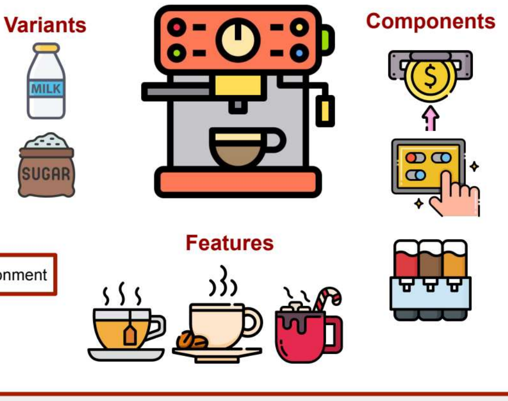
Radboud University, University of Twente

Develop formalisms, methods, and tools to **reduce the combinatorial explosion of testing** evolving systems with high variability

Model Based Testing (MBT)

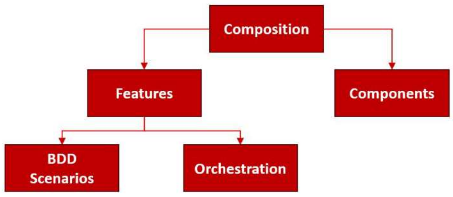


Coffee Machine

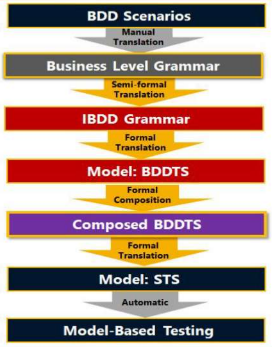


Problem Space

High **variability** makes **testing** and **specifying** systems difficult, we tackle this by applying **composition** over several axes



MBT + BDD



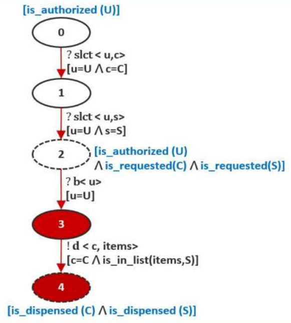
BDD Scenarios

- **Given** the user is authorized
- **When** the user selects coffee
- **And** the user selects sugar
- **Then** coffee with sugar is selected for the authorized user

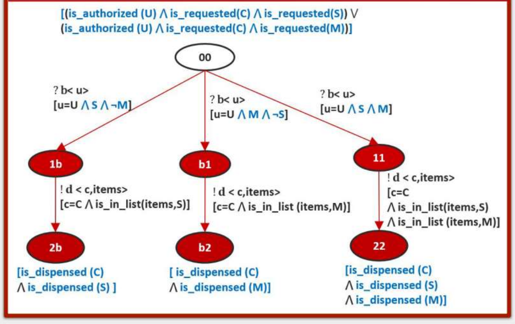
- **Given** the authorized user has selected coffee with sugar
- **When** the user presses the button to dispense coffee
- **Then** the coffee machine dispenses coffee with sugar

- **Given** the authorized user has selected coffee with milk
- **When** the user presses the button to dispense coffee
- **Then** the coffee machine dispenses coffee with milk

Pre-Post condition Composition

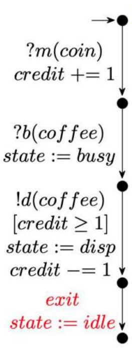

[is_authorized (U)]
0
? slct < u,c>
[u=U ∧ c=C]
1
? slct < u,s>
[u=U ∧ s=S]
2
[is_authorized (U) ∧ is_requested(C) ∧ is_requested(S)]
? b< u>
[u=U]
3
! d < c, items>
[c=C ∧ is_in_list(items,S)]
4
[is_dispensed (C) ∧ is_dispensed (S)]

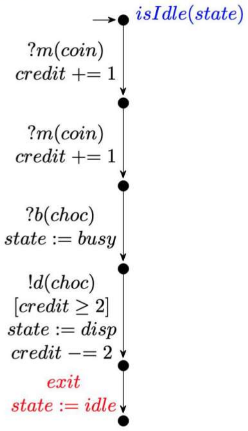
Exclusive Disjunction Composition



Feature Orchestration

A **feature** is a slice of user-visible behaviour of a system with respect to a particular quality characteristic

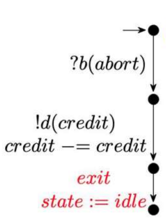




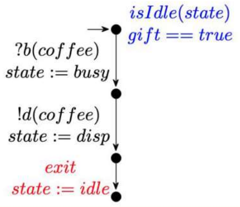
Two features: dispensing coffee and hot chocolate

In the basic case, the underlying behavior is **executing** one of the **enabled** (denoted by isIdle) features until **completion** (denoted by exit), and then **repeating**.

Orchestrating a more Complex System



When enabled, aborting **interrupts** dispensing chocolate or coffee



If the gift is active, dispensing free coffee **overwrites** the normal dispensing of coffee

Key Points

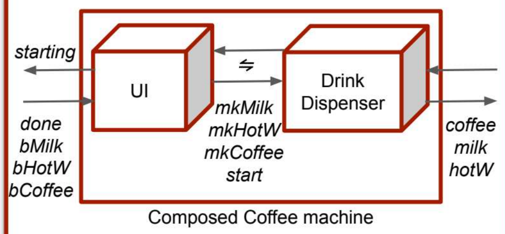
Aim & Future Work

- Define **features** in **compositional** small models
- Enable **fine-grained** control over specifying the interaction of **features** and their **constraints**

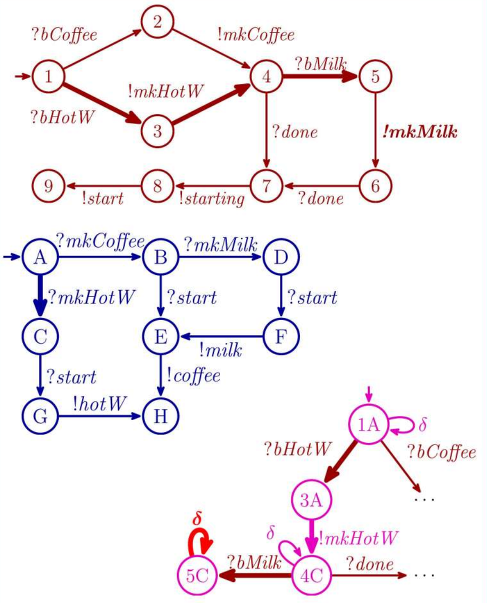
Advantages

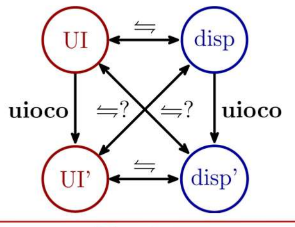
- Smaller, **compositional** models
- Better control over feature **interaction**
- Separating **domain** and **application** knowledge

Component Testing



Composition example





Advantages

- **Smaller models**
- **Easier to make, adapt and maintain**
- **Test earlier**
- **Easier retesting**
- **Easier diagnosis**
- **Replacing components**

Radboud Universiteit · UNIVERSITY OF TWENTE. · NWO · ESI · ASML · Canon CANON PRODUCTION PRINTING