

Tesztgenerálás állapotgép alapú modellekből

Unicsovics Milán György
konzulens: Dr. Micskei Zoltán

Honnan indultam?

- MBT alapok
- teszteszközök magasszintű tulajdonságai
- gráfelmélettel való kapcsolat
- CPP algoritmus implementáció

Motiváció

- hosszútávú cél meghatározása:
 - Papyrus statechartok alapján
 - tesztesetek generálása



Célok

Folyamat:

1. statechart elkészítése Papyrus-ban
2. modelltranszformáció statechart-on
3. tesztgeneráló algoritmusok futtatása
4. tesztesetek futtatása

köztes nyelv (modell) + tesztgeneráló algoritmus



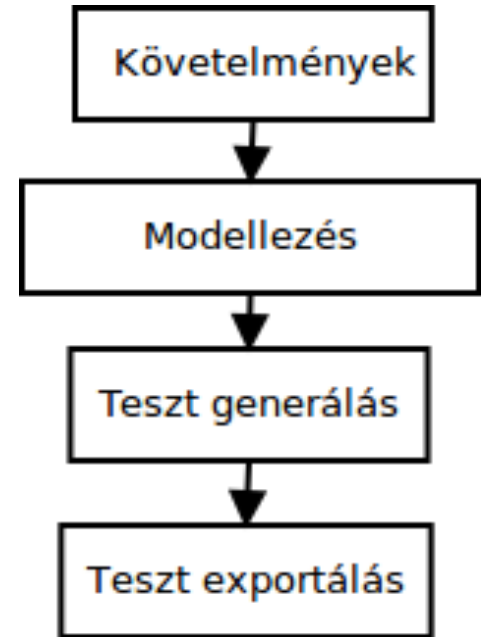
irodalomkutatás + ipari eszközök vizsgálata

Tesztgenerálási lehetőségek

- szimbolikus futtatás
- MBT
- kombinatorikus tesztelés
- véletlenszerű tesztelés
- keresés alapú tesztelés

Conformiq

- ipari MBT eszköz
 - Eclipse plugin / önálló szoftverként
- sok támogatott + beépített saját modellező



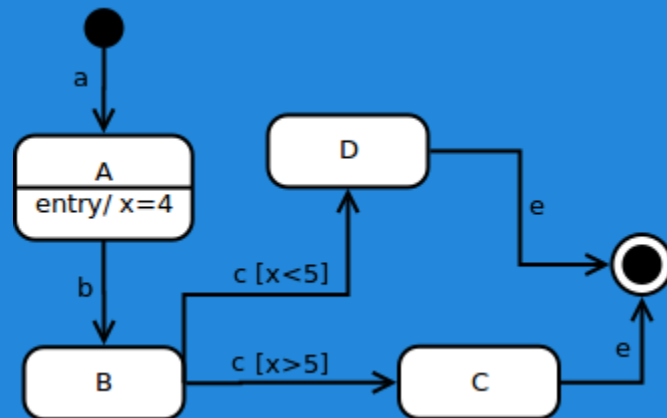
Köztes modell, teszt generálás

Eszköz neve	Modell	Köztes modell	Generálási módszer
GraphWalker	UML	GraphML	keresés alapú (A*, legrövidebb út, minden permutáció), véletlenszerű
PyModel	FSM + Python	gráf	keresés alapú
Conformiq	QML	Lisp (CQλ)	szimbolikus végrehajtás
GOTCHA	EFSM	gráf	keresés alapú (BFS, DFS)
ParTeG	UML + OCL	gráf	keresés alapú (DFS)

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <graphml xmlns="http://graphml.graphdrawing.org/xmlns"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://graphml.graphdrawing.org/xmlns/1.0/graphml.xsd">
5   <key id="d0" for="edge" attr.name="event" attr.type="string"/>
6   <key id="d1" for="edge" attr.name="guard" attr.type="string"/>
7   <key id="d2" for="node" attr.name="action" attr.type="string"/>
8   <graph id="G" edgedefault="directed">
9     <node id="S"/>
10    <node id="A">
11      <data key="d2">x=4</data>
12    </node>
13    <node id="B"/>
14    <node id="C"/>
15    <node id="D"/>
16    <node id="E"/>
17    <edge id="a" source="S" target="A">
18      <data key="d0">a</data>
19    </edge>
20    <edge id="b" source="A" target="B">
21      <data key="d0">b</data>
22    </edge>
23    <edge id="c" source="B" target="C">
24      <data key="d0">c</data>
25      <data key="d1">x>5</data>
26    </edge>
27    <edge id="d" source="B" target="D">
28      <data key="d0">d</data>
29      <data key="d1">x<5</data>
30    </edge>
31    <edge id="e" source="C" target="E">
32      <data key="d0">e</data>
33    </edge>
34    <edge id="e" source="D" target="E">
35      <data key="d0">e</data>
36    </edge>
37  </graph>
38 </graphml>

```



Események, őrfeltételek, akciók

Algoritmusok implementációja

- örfeltételek kipróbálása CPP algoritmusnál
 - → nem használható általánosan
- szélességi keresés
 - rövidebb tesztesetek
- mélységi keresés
 - hosszabb tesztesetek

Értékelés

- hosszútávú célok meghatározása
- köztes modell lehetőségek kutatása
- tesztgenerálási algoritmusok vizsgálata
- tapasztalatok ipari eszköz alapján
- saját implementáció