UNIVERSITÄT KOBLENZ · LANDAU

Business Process Management

Exercise 5

Group 04

Rahul Bhanushali

Davud Ismayilov

Chanchal Gopalakrishnan

Harita Sawant

Vishal Kumar

Michael Agyekum Bremang

TOTAL: 2,5/10

Exercise 5

Task 1 2/2

Design a pattern that represents a function followed by a function without an event in between, even if there are ANY NUMBER OF connectors in-between (e.g., “Function -> XOR -> XOR-> Function” must also be detected, not only “Function -> Function”).

Solution:

```
DirectedPathsNotContainingElements(
                                   ElementsOfType(V, Function) ,
                                   ElementsOfType(V, Function) ,
                                   ElementsOfType(V, Event)
                                  )
```

Task 2 0/3

Design a pattern that represents loops in EPCs containing connectors only (i.e., that start and end with a connector and furthermore contain only connectors on the whole loop).

Solution:

```
Complement(
SelfUnion(
DirectedLoops(
              ElementsOfType(E, XOR_Event)
               )
         ) ,
                Union(
                       Union(
                             Union(ElementsOfType(V,Event),
                                   ElementsOfType(V,Function)
                                    ) ,
                             Union(ElementsOfType(V,XOR),
                                   ElementsOfType(V,AND)
                                    )
                              ) ,
                                   ElementsOfType(V,OR)
                       )
               )
```

This would just return a set of various edges that has nothing to do with connector loops (-3)

Task 3 0/3

Design a pattern that represents connectors having more than one predecessor and more than one successor (i.e., connectors that join and split at the same time).

Solution :

You did not model connectors in your query, but events (-3)

```
Intersection(

                    Complement(
                    ElementsOfType(V, event),
                    Union(
                    SelfUnion(
                    ElementsWithNumberOfPredRelations(ElementOfType(V,event), 0)),
                    SelfUnion(
                    ElementsWithNumberOfPredRelations(ElementOfType(V,event), 1))))
                    ,

                    Complement(
                    ElementsOfType(V, event),
                    Union(
                    SelfUnion(
                    ElementsWithNumberOfSuccRelations(ElementOfType(V,event), 0)),
                    SelfUnion(
                    ElementsWithNumberOfSuccRelations(ElementOfType(V,event), 1))))

                    )
```

Task 4 0,5/2

1. GMQL function and its equivalent CTL formula

The order has to be verified by the boss.

.

CTL : $M, s0 \models \neg EF(n\_verify\_order \land \neg o\_boss)$

GMQL :

```
DirectedPathsContainingElements(
                                        ElementsWithAttributeOfValue(V,label,”Check Order”),
                                        ElementsWithAttributeOfValue(V,label,”Ship Order”),

                                        ElementsDirectlyRelated(
                                           ElementsWithAttributeOfValue(V,label,”Verify Order”),
```

This would already return true if “verify order” was on the path, even if the boss would not be involved (-0,5)

```
                    ElementsWithAttributeOfValue(V,label,”Boss”)
                                    )
          )
```

2. GMQL function that cannot be expressed by a CTL formula

Suppose we want to output our model without the events i.e we want to remove all events in our model. This can be done in GMQL by using COMPLEMENT operator but it is not possible in CTL since there does not exist such operator and CTL only find paths through its queries.

GMQL : Complement(V, ElementOfType(V, event))

CTL : Not possible

COMPLEMENT is not a GMQL function, but a set operator (-1)