REBS 2021 A3: BPC and Jolie

njx155: Lars KL
jsx103: Josiah Grønhøj
hck787: Hu Guo
tnd179: Xingrong Zong

18th of January 2022
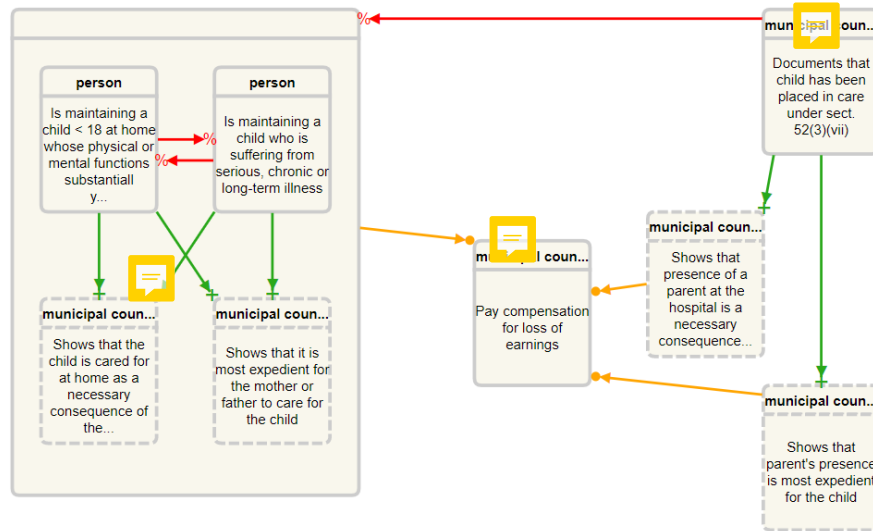
Exercise A.1

Figure 1: Here is the DCR graph of the policies in §42 of the Danish Consolidation Act for Social Services.
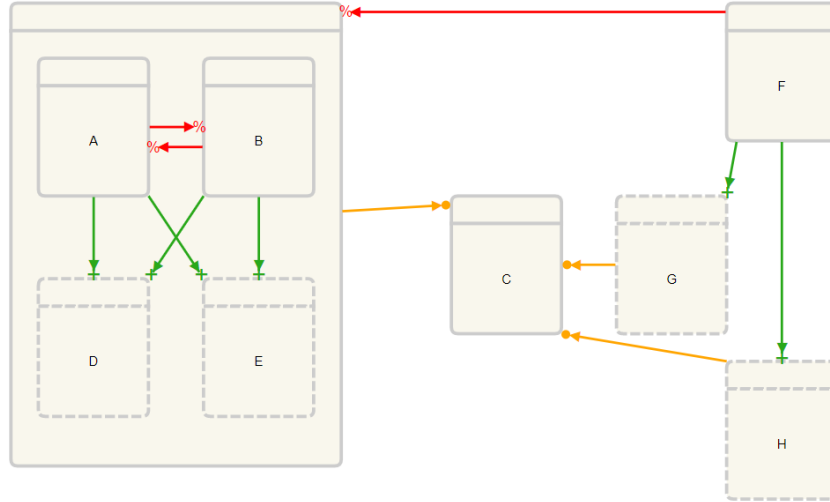
Exercise A.2

Figure 2: Here is the DCR graph where every activity has been labeled with letters from A-J to use for tracing.

We now present our traces where the following indicates:

✓: Compliant
✗: Violated

1. $< A, D, E, C >$ ✓
2. $< B, D, E, C >$ ✓
3. $< F, G, H, C >$ ✓
4. $< A, C >$ ✗
5. $< G, H, C >$ ✗
6. $< B, D, E, F, C >$ ✗

Test for A.2

We generate the traces like a log text file in Assignment 1. When the compliance checker runs, the pattern can be shown in the terminal, and the result of the pattern for that logs will be also shown in the terminal and they are also saved as a text file. Figure 3 shows the results in terminal. The details to run the checker can be referred from README.md.

```
*********The result of pattern 5 is: *************
In 4, Event C is not enabled, so reject.
In 5, Event G is not enabled, so reject.
In 6, Event C is not enabled, so reject.
total amount is: 6
reject amount is: 3
accept amount is: 3



Partially Compliant
```

Figure 3: Test for A.2

Exercise A.4

a)

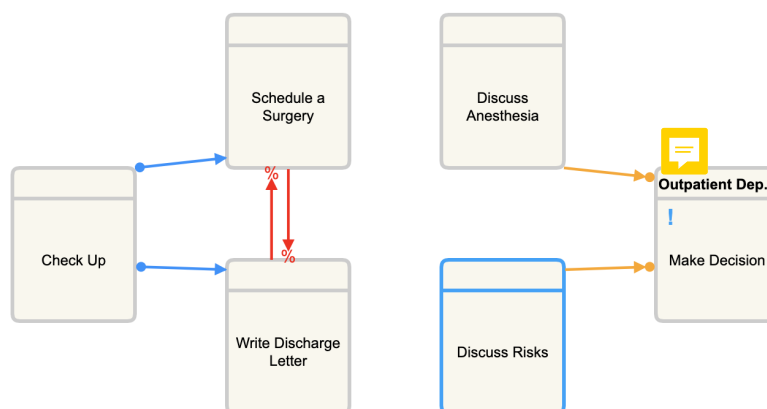Figure 4 shows the DCR compliance rules for a.1 a.3.

Figure 4: DCR Compliance Rules for a.1 a.3

b)

Figure 5 shows the equivalent DCR process model describing the behaviours of the topmost pool. It is compliant with the DCR graph in previous section because: the "Make Decision" is marked as pending, which means it must be executed, and no exclusion relationship is related to it. Also, after "Make Decision", "Schedule Surgery" and "Write Discharge letter" are pending, so a acceptable trace must contain this two activities after "Making Decision" and the condition to "Make Decision" must include "Checkup". So as long as the "Checkup" happens, "SS" and "WDL" must happen. Finally, "Make Decision"'s conditions are "DA" and "DR", which means that a "Make Decision" must follow these two activities.
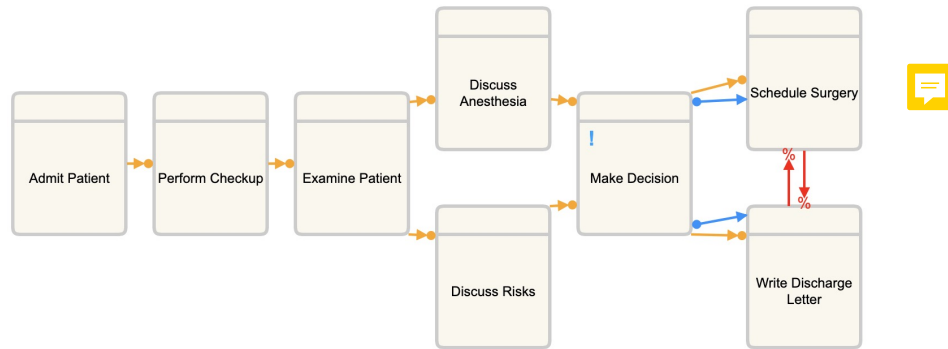

Figure 5: Equivalent DCR graph

Exercise B.1

B.1.1

2.2

The messages received by the two subscribers in two shells are the same. The order of the messages are also the same. For message queueing system, a message is sent from an actor to another. A message can only be consumed once. For a Publish-Subscribe system, a message can be subscribed by multiple consumers.

2.3

Subscriber **Subscriber_InboundCall.ol**: The regular expression for topic **Inbound Call** is "pmcep/Disco Example Log/+/Inbound Call". All the messages received by this subscriber are with topic "Inbound Call".

Subscriber **Subscriber_InboundEmail.ol**:The regular expression for topic **Inbound Email** is "pmcep/Disco Example Log/+/Inbound Email". The test results for the two subscribers are shown in Figure 6 and **inbounemail**.

```
topic : pmcep/Disco Example Log/Case3230/Inbound Call
message : {"event":{"concept:name":"Inbound Call","Customer_ID":"Customer 2707"
l","Product":"iPhone","lifecycle:transition":"complete","Agent_Position":"FL"}
ant 1","variant-index":"1"}}
topic : pmcep/Disco Example Log/Case324/Inbound Call
message : {"event":{"concept:name":"Inbound Call","Customer_ID":"Customer 277"
d Call","Product":"iPhone","lifecycle:transition":"start","Agent_Position":"FL"
iant 1","variant-index":"1"}}
topic : pmcep/Disco Example Log/Case2160/Inbound Call
message : {"event":{"concept:name":"Inbound Call","Customer_ID":"Customer 1813"
 Call","Product":"iPhone","lifecycle:transition":"start","Agent_Position":"FL"
iant 1","variant-index":"1"}}
```

Figure 6: Test 2.3: Subscriber for Topic: Inbound Call

```
topic : pmcep/Disco Example Log/Case1869/Inbound Email
event : {"event":{"concept:name":"Inbound Email","Customer_ID"
il","Product":"MacBook Pro","lifecycle:transition":"start","Ag
ariant 8","variant-index":"8"}}
topic : pmcep/Disco Example Log/Case3751/Inbound Email
event : {"event":{"concept:name":"Inbound Email","Customer_ID"
mail","Product":"MacBook Pro","lifecycle:transition":"start","
"Variant 4","variant-index":"4"}}
topic : pmcep/Disco Example Log/Case143/Inbound Email
event : {"event":{"concept:name":"Inbound Email","Customer_ID"
ail","Product":"iPhone","lifecycle:transition":"start","Agent_
t 108","variant-index":"108"}}
^Ctopic : pmcep/Disco Example Log/Case3267/Inbound Email
```

Figure 7: Test 2.3: Subscriber for Topic: Inbound Email

2.4

Subscriber in **Subscriber_Count.ol** use a "dictionary" to record the topics and the number of times they appear. The dictionary is consisted of two global array variables global.eventNames and global.eventCounts. Whenever a message is received by Subscriber **Subscriber_Count**, if the topic exists in the dictionary, the corresponding number is added, and if it does not exist, a new topic with number 1 is added to the dictionary. Every time a message is received, the distribution of different topics will be shown in the terminal. The test results are shown in Figure 8.

Figure 8: Test 2.4: Subscriber to account the topics