Banki napi zárás egyszerűsített modellje UPPAAL-ban

Formális módszerek feladat

LGM_IN003_1

Készítette:

Hadházi Zsolt

T9TTQQ

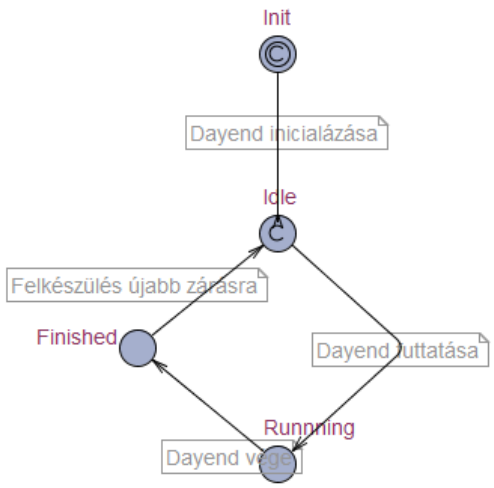
1. Feladat ismertetése:

   A feladatban egy banki számlavezető rendszer napi zárásának egyszerűsített modelljét készítem el.

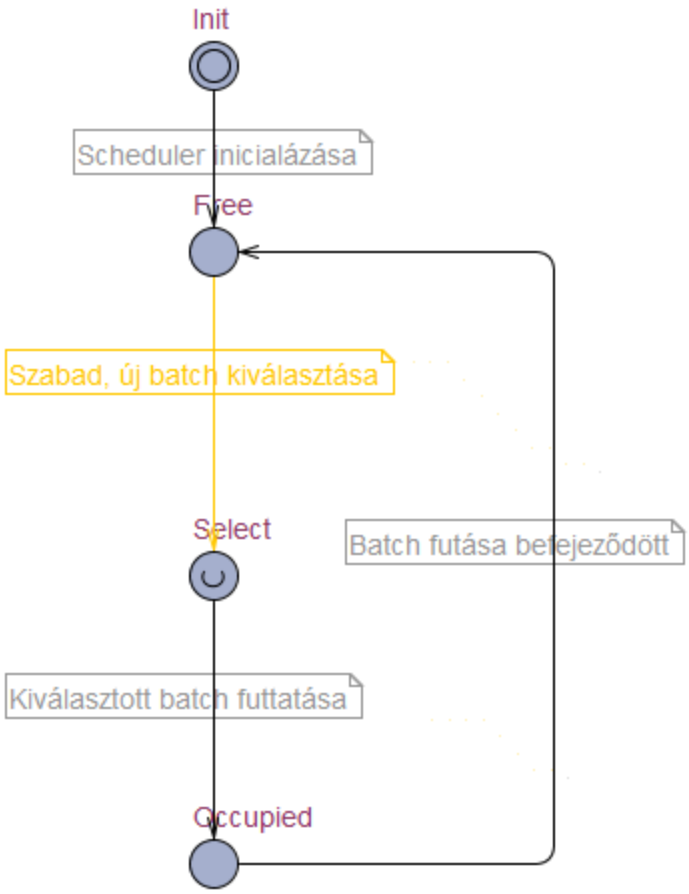
   Az általam ismert számlavezető rendszerek kötegelt feldolgozást végeznek, azaz a zárási folyamat különböző batchekre kerül felosztásra, és ezek egymás után/egymással párhuzasan futnak. A számlavezető rendszer arhitektúrájától függően egy batchbe kerülhetnek azononos típusú termékek feldolgozása, vagy pedig azonos típusú feladatok elvégzése az összes számlára. Például a Batch(0) végzi el a hitelszámláknál a napi kamatokkal kapcsolatos teeendőket, a Batch(1) a betéti számlákkal kapcsolatos kamatakkruálálsokkal foglalkozik, a Batch(3) a kamatváltozásokat vezeti át, stb..

   A batchek sorrendjének szervezése már fejlesztési időben megtörténik, ezért a futási sorrend előre meghatározott.

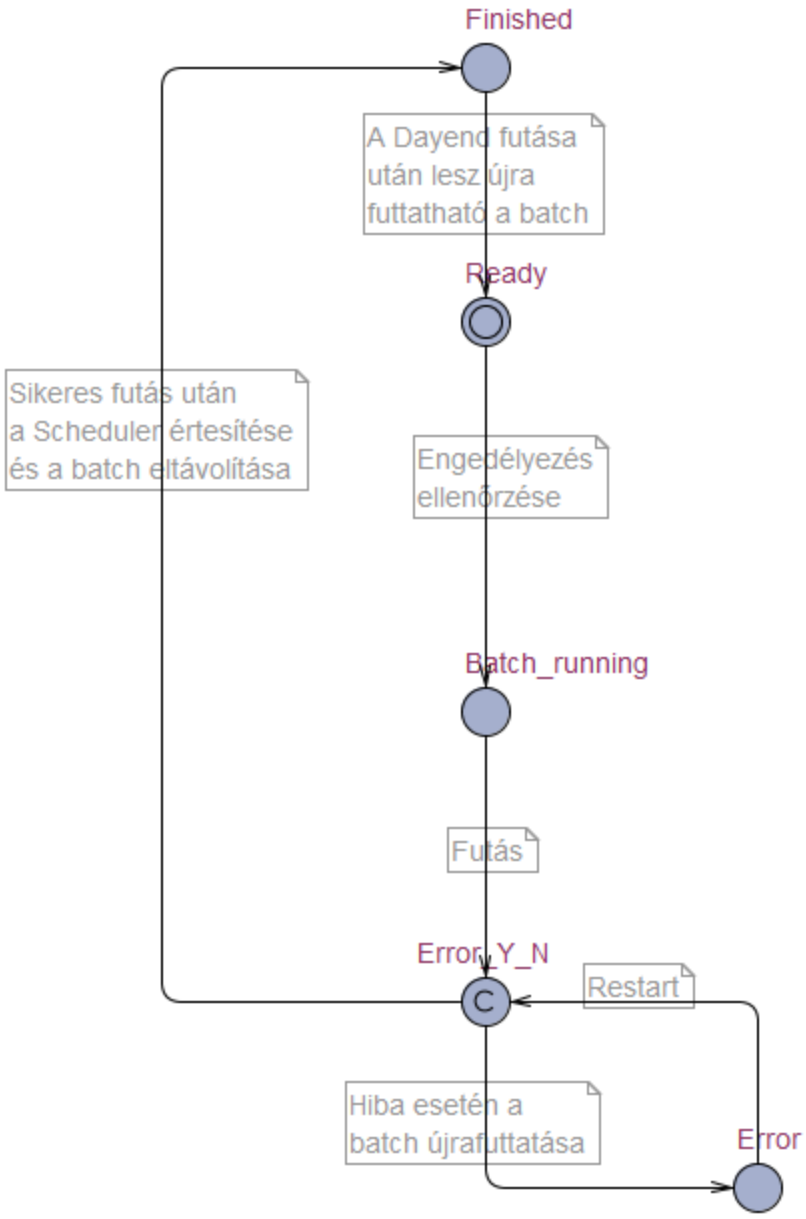
2. Megoldandó feladatok:
   - A futtatandó feladatok összeállítása
   - A futtatandó batch kiválasztása és futtatása
   - A szereplők közötti kommunikáció megoldása
   - Modell ellenőrzés
3. A modell elemei:
   - A napvégi zárást a Dayend állapotváltazásaival tudjuk leírni. A dayend-ből a modellben csak egy pédányra van szükség, ezzel tudjuk modellezni a napvége futását.



   - A Schedulerrel modellezzük a feldolgozandó batch kiválasztását, futási állapotát, illetve lefutása után felkészülünk az új batch futására.



- A Batch-el modellezzük az adott batch futását. A modellben a hibakezelés olyan módon lett egyszerűsítve, hogy amennyiben a batch feldolgozása során hibára fut, akkor még egyszer lefut és ebben az esetben már biztosan nem futhat hibára. Ez az ág azt modellezi, hogy a napvége hibára futása után az üzemeltetők/fejlesztők javítják a hibát és utána folytatják a zárás futását.



- Függvények és struktúrák:
    - A megoldás alapgondolata, hogy a futtatandó batchek sorrendje egy feldolgozási sorba (queue) kerül előkészítésre. Ebben a queue-ban a futtatandó batch-ek id-ja kerül. A futás során a queue elején lévő batch futtatható, a batch futásának végén pedig az elklészült batch-et eltávolítjuk a queue-ból.

    - initialize(): A futás elején itt állítjuk össze a futási sorrendet.
    - remove(): A sor elejéről eltvolítja az elkészült batchet.
    - isEmpty(): Ellenőrzi, hogy van-e futtatható batch.
- Kommunikáció:
    - A dayend egy broadcast channel segítségével értesíti a batcheket az aktuális futás sikerességéről és ezután az összes batch újra futtatható állapotba (Ready) kerül.
    - A Scheduler és a Batch-ek sima channelen keresztül kommunikálnak. A Scheduler a batch_running channellel értesíti a futtatható batchet, a batch a batch_finished csatortnán jelzi, hogy lefutott

4. Modell verifikáció során a következő ellenőrzéseket végeztem:
    - A rendszerben nincsen holtpont.
    - Egyszerre csak egy batch futhat, az összes többi már lefutott, vagy pedig futásra várakozik.
    - Amennyiben a 2. batch fut, akkor a 3. futásra vár.
    - Amennyiben a 1. batch fut, akkor a többi futásra vár.
    - Amennyiben egy batch elkezdi futását, akkor biztosan be is fogja fejezni azt.