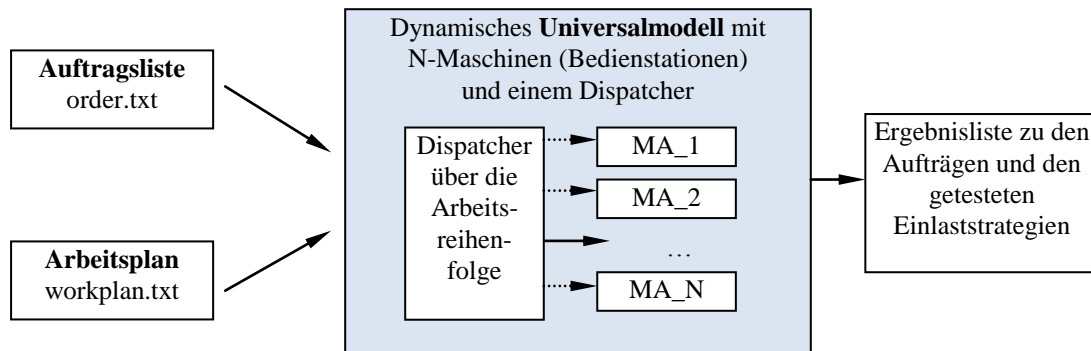


Implementierung eines universellen Simulationsmodells für flexible Bediensysteme mit SLX

1. Verwenden die aktuelle Version des SLX-Compilers aus dem Verzeichnis `i:\prakt\wiedem\simulation\SLX2012\`.
2. Entwickeln Sie ein universell nutzbares Modell für Bediensysteme mit folgenden Leistungsmerkmalen:



- Ausgehend vom Arbeitsplan (genaues Format am Ende der Aufgabenstellung) soll die entsprechend benötigte Anzahl von Maschinen (bzw. Bedienstationen) angelegt werden. Jede Maschine besteht aus einer Bedienzelle und einer vorgelagerten Warteschlange mit ausreichender Größe für die Auftragslose (z.B. 10 Plätze) !
- Ausgehend von der Auftragsliste sollen Maschinen-Aufträge angelegt werden und mit der Bearbeitung (ggf. auch verzögert in Abhängigkeit vom Endtermin) begonnen werden. Jeder Auftrag soll dabei als Gesamtlos (engl. Batch) bearbeitet werden, um unnötige Transport und Rüstkosten zu vermeiden (damit Stückzahlen in Auftragsliste als Multiplikator für Einzelzeiten in Arbeitsplan und nicht als einzelne Objekte generieren!).
- Die Reihenfolge und die notwendige Zeit zur Bearbeitung jedes Auftrags pro Maschine ist in der Arbeitsplandatei workplan.txt abgelegt.
- Bei Existenz mehrerer Aufträge soll der Dispatcher nach mindestens 3 verschiedenen Regeln den nächsten Auftrag auswählen (vgl. Vorlesung). Die Ergebnisse pro Auswahlstrategie – die gesamte Zeit für die Erfüllung aller Aufträge und die Auslastung aller Maschinen sollen ausgegeben werden und eine Strategie soll empfohlen werden.
- Realisieren Sie nach Fertigstellung des Simulationsmodells eine einfache Animation mit PROOF (P3D) in Anlehnung an die obige Skizze. Die einzelnen Aufträge sollten dabei unterscheidbar sein (durch Text oder Farbe).

Bitte bauen Sie das Modell und die Animation mit den ersten kleinen Inputdateien auf, für den finalen Test und die Abnahme werden dann noch größere und realistischere Daten bereitgestellt. Zur Abnahme sind alle Sourcecodedateien und eine kurze Dokumentation der Vorgehensweise (Diagramm zur Modellfunktion) und eine Ergebnisbeschreibung für den „Endkunden“ bereitzustellen. (Abgabetermin ca. Mitte Juni)

Beschreibung der Inputdateien

(zuerst als kleinere Testdateien)

1. Format der Auftragsdatei (order.txt)

AuftragsID	ProduktID	Anzahl_Produkte	Startzeit([Tage]!!)	Liefertermin [Tage]
1	1	20	0	5
2	2	30	0	5
3	3	10	2	9
...				

Diese Datei kann mit SLX wie nachfolgend in ein SET eingelesen werden:

```
open f input name = orderfile; // = "order.txt" in dieser Übung
forever { op = new order; flag=TRUE; // op ist vom Typ Klasse order mit nachf. Attributen

read newline file=f end= loop1 ( op->oid,op->product,op->amount,op->starttime,
                                op->endtime);

// bedingte Debugging-Ausgabe (optional)
if (plevel>2) {print( op->oid,op->product,op->amount,op->starttime,op->endtime)
               "Order_ : Product=_ Amount=_ Start=_ End=_ \n ";}
place op into orders; // setze neue Bestellung (Order) in das Set der Bestellungen
} loop1;;
```

2. Format der Arbeitsplandatei (workplan.txt)

ProduktID	ArbeitsgangNr	Zeit [min]	MaschinenID
1	1	15	1
1	2	5	2
1	3	10	3
2	1	10	2
2	2	15	3
...			