

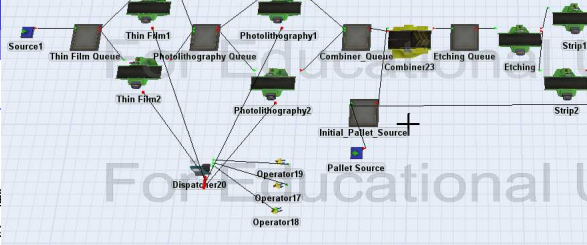
Flexsim: 資料庫系統連結 (Open DataBase Connectivity, ODBC)

W. M. Song 桑慧敏
Tsing Hua Univ. 清華大學

2015.12.09

- 1 系統假設與流程說明
- 2 Access資料庫建立與設定
- 3 Flexsim模型建立與設定

系統假設與流程



- 在 Microsoft Access 建立表以儲存模擬結果回傳至 Access 資料
- 車廠零件商進行零件加工, 有三台沖床機, 其 Process Time 分別為 $\text{uniform}(1,5)$ 、 $\text{uniform}(1,10)$ 及 $\text{uniform}(1,15)$
- Source 之 Inter-Arrival Time 為 $\text{exponential}(0,5)$, 隨機分配至有空的機台
- 求得時間=1000 的時候, 各機台加工量為多少?
- 流程為下圖:

建立Access 資料表單

- 開啓 MS-Office Access 新增一資料庫, 另存為 Access 2002-2003 資料庫 (*.mdb), 此命名為 ODBC.mdb

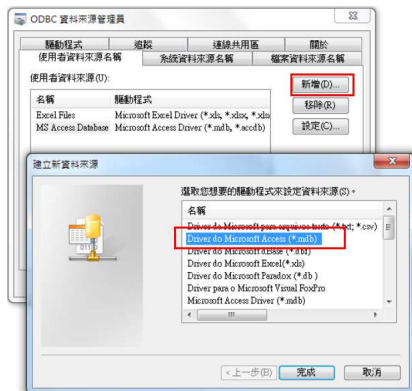


- 新增一表單, 命名為 Table1(可自行命名)
- 表單中 ObjectName 設為文字型態欄位, 其餘設為數值型態欄位



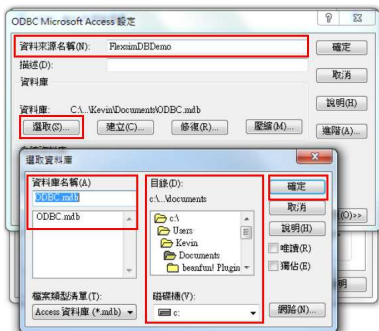
ODBC設定

- 依路徑開啓 C:\Windows\SysWOW64\odbcad32.exe
- 使用者資料來源名稱 → 新增 → Driver do Microsoft Access(*.mdb) → 完成



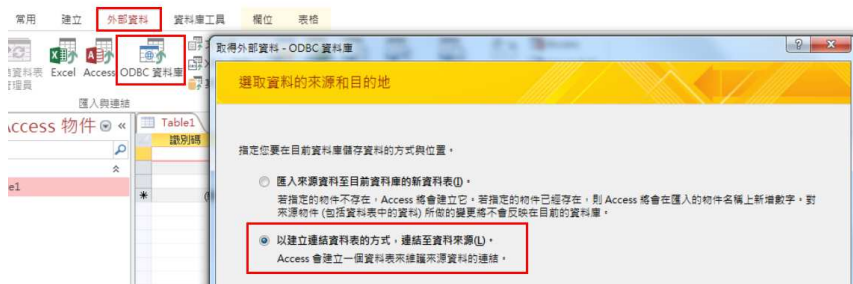
ODBC設定

- 資料來源名稱:FlexsimDBDemo(可自行命名)
- 資料庫 → 選取
- 依照 Access 資料庫儲存的位置依序選取磁碟機及目錄, 在左側會顯示檔案名稱, 選取剛建立的檔案並按確定



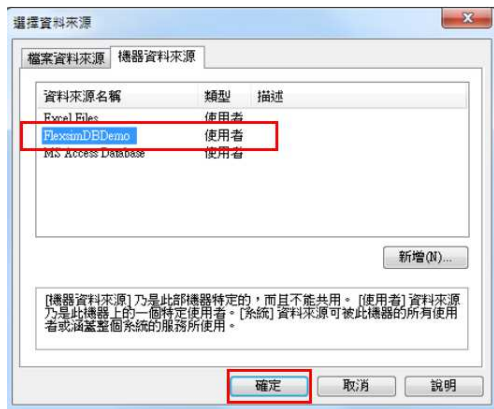
ODBC設定

- 回到 Access 畫面, 點選外部資料 → ODBC 資料庫 → 以建立聯結的方式, 連結至資料來源 → 確定



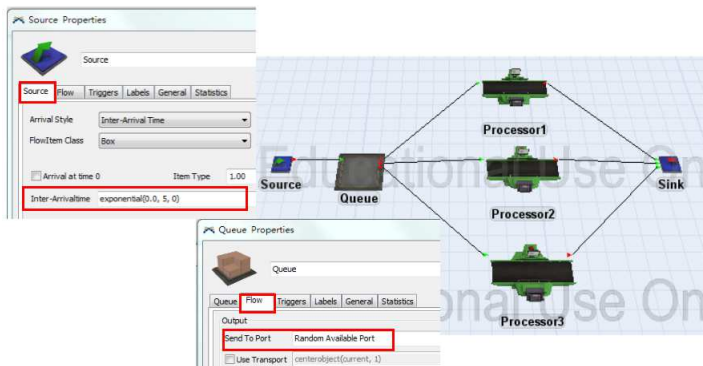
ODBC設定

- 機器資料來源 → FlexsimDBDemo(剛建立的資料來源) → 確定



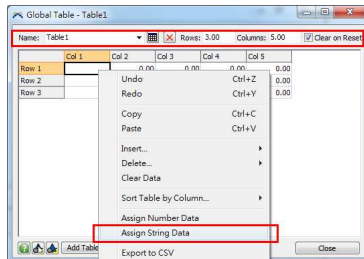
Flexsim模型流程

- Source → Inter-Arrivaltime: exponential(0,5,0)
- Queue → Flow → Send To Port: Random Available Port



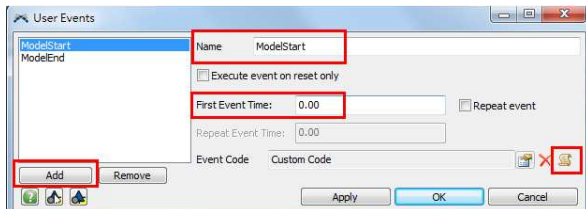
Global Table 設定

- Tools → Global Tables → Add
- Name: Table1(可自行命名) Rows:3 Columns:5(依照 Access 表格建立), 勾選 Clear on Reset
- Col1 按右鍵選取 Assign String Data 將該欄位屬性改為字串模式 (配合 Access 表單)
- 設定為字串模式後將表格中 0.00 刪去, 可以刪除表示屬性更改成功



User Events 設定

- Tools → User Events → Add
- Name:ModelStart(可自行命名)
- First Event Time:0 表示在時間0的時候執行
- Event Code 選取最右方之圖示編輯程式



User Events 設定

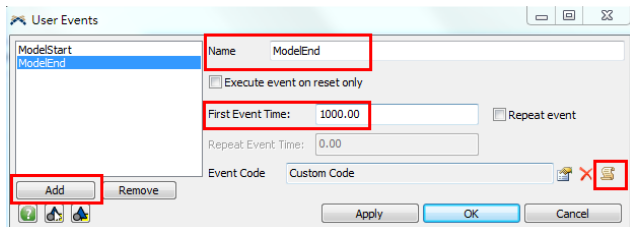
- msg("視窗標題","視窗中的文字", 視窗按鈕類型)
- 視窗按鈕類型=4, 表示按鈕為 Yes(回傳值=1) 及 No(回傳值=0)
- dbopen("ODBC 資料來源名稱","command (SQL string or Tablename), ", 讀取模式)
- 讀取模式=1, 表示以 Table Mode 讀取
- dbimporttable("要匯入資料的 Global Table 名稱", 起始 row, 起始 column,row 的數量,column 的數量),dbgetnumrows() 自動抓取列的數量
- dbclose() 關閉資料庫



```
ModelStart - Event Code*
1 /**Custom Code*/
2 treenode current = ownerobject(c);
3
4 if(msg("Import Data", "要載入資料庫的資料嗎?", 4) ==1)
5 {
6     dbopen("FlexsimDBDemo", "Table1",1);
7
8     dbimporttable("Table1",1,2,dbgetnumrows(),4);
9
10    dbclose();
11
12 }
13 else
14 {
15     return 0;
16 }
```

User Events 設定

- Tools → User Events → Add
- Name:ModelEnd(可自行命名)
- First Event Time:1000 表示在時間 1000 的時候執行
- Event Code 選取最右方之圖示編輯程式



User Events 設定

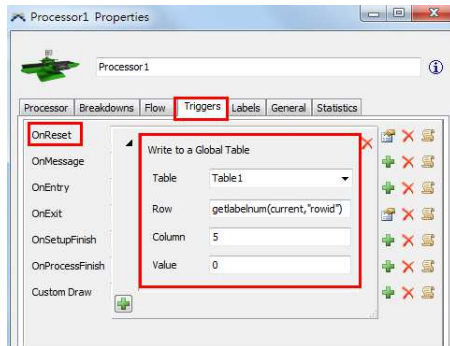
- dbopen("ODBC 資料來源名稱","command (SQL string or Tablename),","讀取模式,")
- 讀取模式=0, 表示以 SQL 模式讀取
- dbsqlquery : 使用 SQL Query
- concat: Combine strings



```
1 /**Custom Code*/
2 treenode current = ownerobject(c);
3
4 if(msg("Export Data", "要將資料寫到資料庫嗎?", 4) == 1)
5 {
6     //SQL Mode
7     dbopen("FlexsimDBDemo", "SELECT 1", 0);
8     for (int row=1; row<=gettablerows("Table1"); row++) {
9         dbsqlquery(concat(
10             "UPDATE Table1 SET Throughput=", // 更新Table1內的Throughput欄位
11             numtostring(gettablenum("Table1", row, 5), 0, 0), // 值
12             " WHERE ObjectName=", // 以ObjectName作為索引
13             gettablestr("Table1", row, 1),
14             ""
15         ));
16     }
17     dbclose();
18 }
19 else
20 {
21     return 0;
22 }
23
```

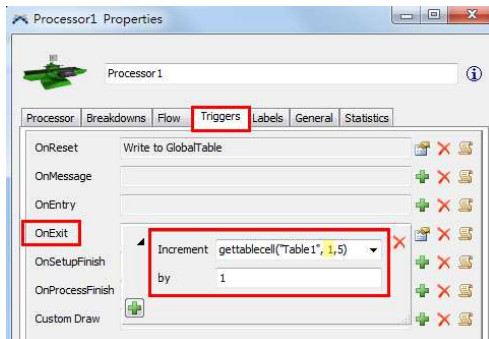
Processor設定

- 設定每次 Reset 的時候將 Global Table 的輸出欄位記錄歸 0
- Processor → Triggers → OnReset → Write to a Global Table



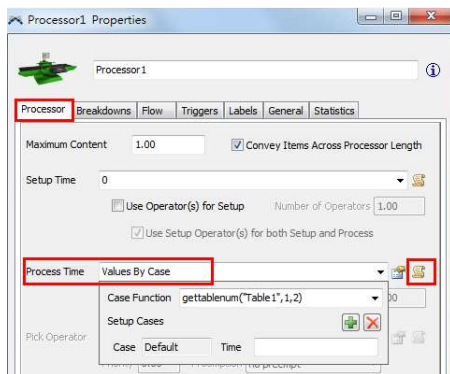
Processor設定

- 設定每次加工完的時候將 Global Table 的輸出欄位記錄加 1
- 黃底部分需依據機台編號而改變
- Processor → Triggers → OnExit → Increment Value



Processor設定

- 設定 Process Time 使其讀取 Global Table 資訊
- 選擇 Value By Case 並按最右方圖示加以修改程式
- Processor → Process Time → Value By Case → 進入修改程式



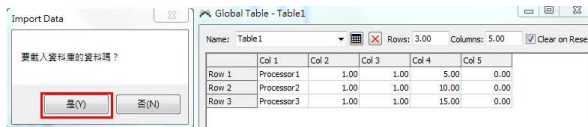
Processor設定

- 依據 Table1 的第 2 欄決定 Process Time Distribution, 第 3、4 欄為 Distribution 之參數
- Table1 的第 2 欄為 1 則用 uniform, 為 2 則用 normal
- 黃底部分需依據機台編號而改變

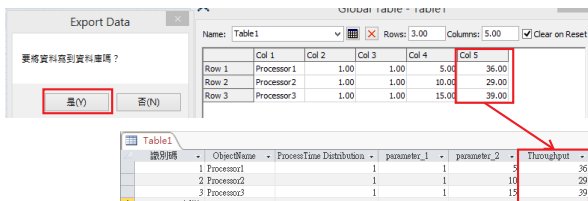
```
Processor1 - Process Time
Processor1 - Process Time
1 treenode current = ownerobject(c);
2 treenode item = parnode(1);
3 /**popup:ValuesByCase:hasitem=1:valuestr=Time*/
4 /**Values By Case*/
5 /** \nCase Function: */
6 int case_val = /**tag:ValueFunc/**/gettablenum("Table1",1,2)/**/;
7
8 switch (case_val)
9 {
10 /**tagex:data*/
11 case 2 : return normal(gettablenum("Table1",1,3),gettablenum("Table1",1,4));
12 default: return uniform(gettablenum("Table1",1,3),gettablenum("Table1",1,4));
13 }
14
15 return 0;
--
```

執行Model

- 按 Run 之後會出現之前設置的視窗, 按是 (Y) 則會看到 Access 的資料匯入至 Table1



- 執行到時間 1000 時, 會出現之前設置的視窗, 按是 (Y) 則會看到 Table1 的資料匯出至 Access



前後差異

以前的做法

- 將 Access 資料庫直接存成附檔名為.mdb 的檔案

正確的做法

- 將 Access 資料庫另存成 Access 2002-2003 資料庫 (*.mdb)