

## Chapter 7

### 運輸與供應鏈模擬

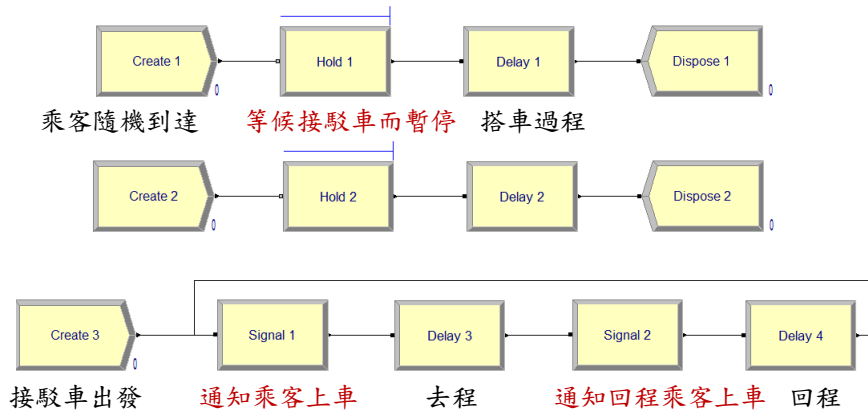
- 利用Hold與Signal模組控制流程暫停與繼續進行
- 使用Set資料模組，讓entity可以由不同類型的資源來處理
- 使用2-dim Variable與Set資料模組合併相似的流程，但是仍然保持各自運作

## 7.1 接駁巴士模擬

- 某城市提供接駁車載運乘客往返於高鐵站與旅遊景點之間，中途沒有停靠站，接駁車有乘坐上限(25)，管理者可以控制接駁車數量與發車間隔，希望以模擬評估對載客率與乘客等候時間的影響。
- 模擬包含三個同時進行的流程：
  - (1) 乘客在高鐵站等候接駁車，並搭車前往旅遊景點。
  - (2) 乘客在旅遊景點等候接駁車，並搭車前往高鐵站。
  - (3) 接駁車往返行駛於兩地之間的過程。

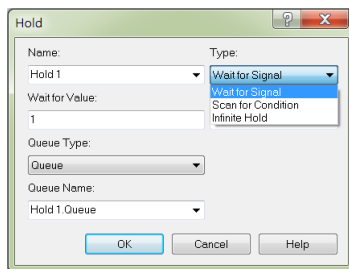


## 接駁巴士的初步流程設計



Question: 如何控制上車人數? 如何讓人車同時到達?

## 乘客流程的Hold模組

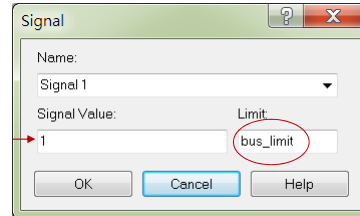
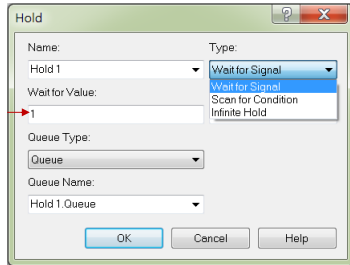


Hold模組內的Type選項有三個，代表流程恢復進行的方式。

- **Wait for Signal**是被動等待其他流程的Signal模組送出信號
- **Scan for Condition**是主動觀察某個條件是否成立。
- **Infinite Hold**終止流程，entity進入休眠狀態，必須要由其他流程的模組在其他位置取出。

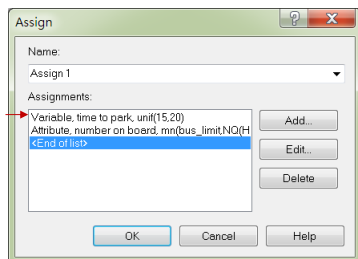
- **Wait for Value**選項用數值來區分不同信號
- Limit欄位空白，表示不限制可接收信號的entities數量
- Hold 1.queue儲存暫停的entities

## 接駁車流程的Signal模組



- Signal Value: 1代表通知高鐵站乘客上車的信號，2代表通知旅遊景點乘客上車的信號。
- Limit欄位設定發送信號的上限，代表接駁車載客上限，可輸入25，或輸入變數bus\_limit。若欄位空白，則無上限。

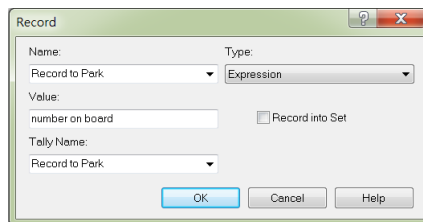
## 接駁車記錄載客人數



加入Assign模組以計算載客數

載客人數number on board等於  
 $\text{mn}(\text{bus\_limit}, \text{NQ}(\text{Hold 1.Queue}))$

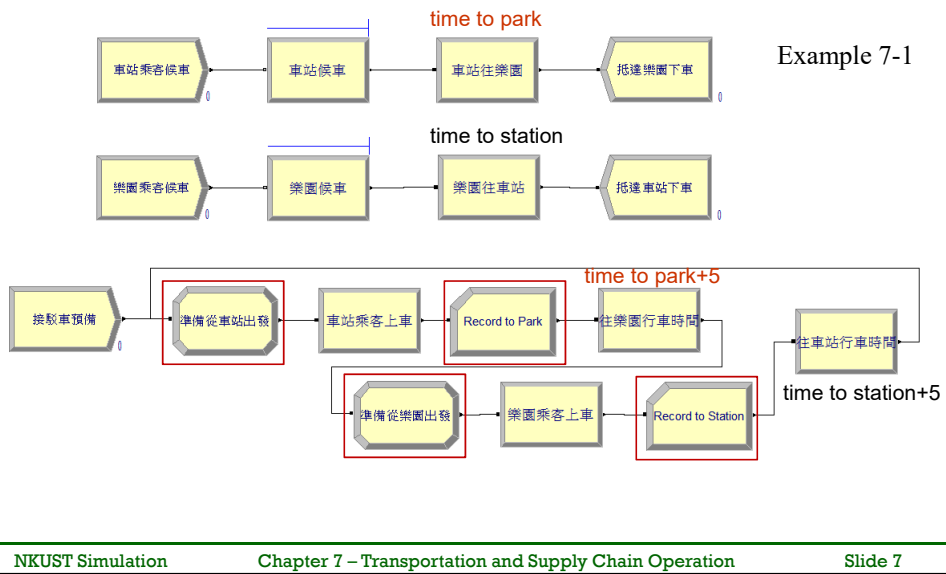
變數time to park控制乘客流程與接駁車流程同步



Record模組紀錄並統計載客數

14.5版視窗畫面

## 接駁巴士的完整流程設計



## Example 7-1 模擬結果

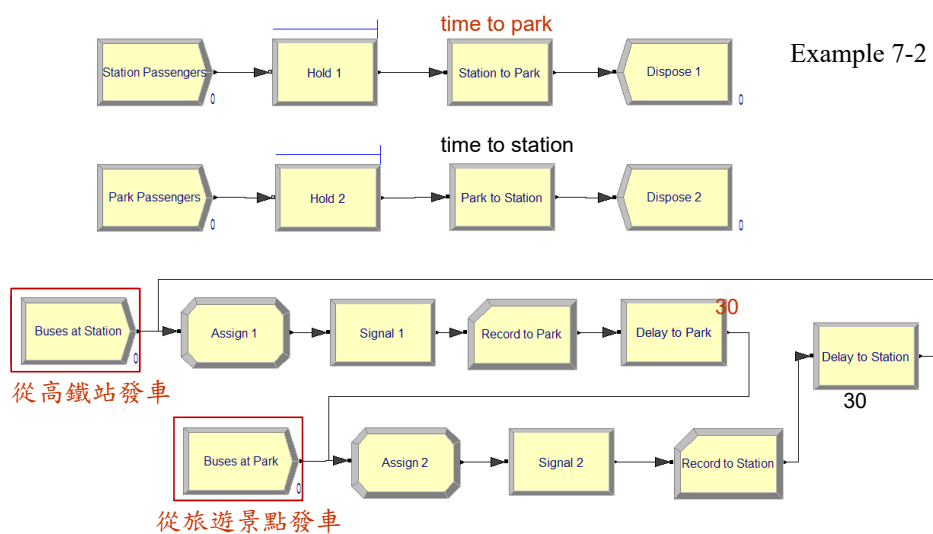
Replication Length = 18 hours

	高鐵站出發	旅遊景點出發	
乘客平均等待時間	8.3852	8.2789	
乘客等待時間最大值	25.4899	26.9825	
排隊人數最大值	37	35	載客上限為25
平均載客數	13.5368	14.4681	

## 改變發車方式的影響

- 原有方式為所有接駁車都從高鐵站發車，起始的間隔為15分鐘，接下來都是到站後停留5分鐘再行駛回程。
- 由於行駛時間變化，容易造成發車間隔不穩定。
- 新的發車方式是各有兩部車從高鐵站與旅遊景點發車，起始的發車間隔仍為15分鐘。規定每班車從任何一站出發後半小時，就從另一站發車回來。
- 降低不確定性：對駕駛員而言，等於每趟的行駛時間與休息時間相加為30分鐘，對旅客而言，等於固定每隔15分鐘就發車。
- 預期班次將減少。

## 改變發車方式後的流程



## 模擬18小時的結果比較

Example 7-1

	高鐵站出發	旅遊景點出發
乘客平均等待時間	8.3852	8.2789
乘客等待時間最大值	25.4899	26.9825
排隊人數最大值	37	35
平均載客數	13.5368	14.4681

Example 7-2

	高鐵站出發	旅遊景點出發
乘客平均等待時間	7.8621	7.5479
乘客等待時間最大值	19.764	15.000
排隊人數最大值	31	25
平均載客數	19.291	17.791



以上的討論僅根據一次replication的模擬結果

pickup, dropoff

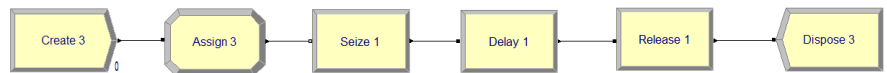
## 7.2 多廠區訂單分配問題

- 水果商不定期釋出product 1, product 2的緊急訂單，分給兩家代工夥伴。A夥伴的工廠為：Kaohsiung(2), Dalian(1), Shanghai(1, 2)，B夥伴的工廠：HsinChu(1), Shenzhen(2)
- 各廠平常處理來自一般客戶的訂單，當接到水果商緊急訂單時，會優先安排上線生產，產線每次只處理一張訂單。
- 緊急訂單的分配原則是有產能可立即生產者優先，如果有兩個廠可選擇時，指派的優先順序為：

priority	Product 1	Product 2
1	Dalian (A)	Shanghai (A)
2	Shanghai (A)	Kaohsiung (A)
3	HsinChu(B)	Shenzhen (B)

## Modeling Issues

- Entity: 五個廠各自的一般訂單，兩種緊急訂單
- Resource: 五個廠的生產線
- 各廠的一般訂單流程相似，可以使用共通的流程模組，以簡化模式架構。



Q1: 如何讓相似流程共用模組，但仍保持差異？

Q2: 如何分配緊急訂單給各廠？

Q3: 如何讓各廠同時處理一般訂單與緊急訂單？

## 以集合來處理多類型與多資源的流程

- 先在Resource資料模組定義五個資源，名稱為Dalian, HsinChu, Kaohsiung, Shanghai, Shenzhen，然後在Set資料模組定義一個集合 **factory set**，類型為Resource，將五個工廠依序納入集合。

Set - Basic Process			
	Name	Type	Members
1	Set P1	Resource	3 rows
2	Set P2	Resource	3 rows
3	factory set	Resource	5 rows
4	Set Entity Type	Entity Type	5 rows

Members	
	Resource Name
1	Dalian
2	Hsinchu
3	Kaohsiung
4	Shanghai
5	Shenzhen

Set - Basic Process			
	Name	Type	Members
1	Set P1	Resource	3 rows
2	Set P2	Resource	3 rows
3	factory set	Resource	5 rows
4	Set Entity Type	Entity Type	5 rows

Members	
	Entity Type
1	Order 1
2	Order 2
3	Order 3
4	Order 4
5	Order 5

Advanced Set - Advanced Process			
	Name	Set Type	Members
1	Set Queue	Queue	5 rows

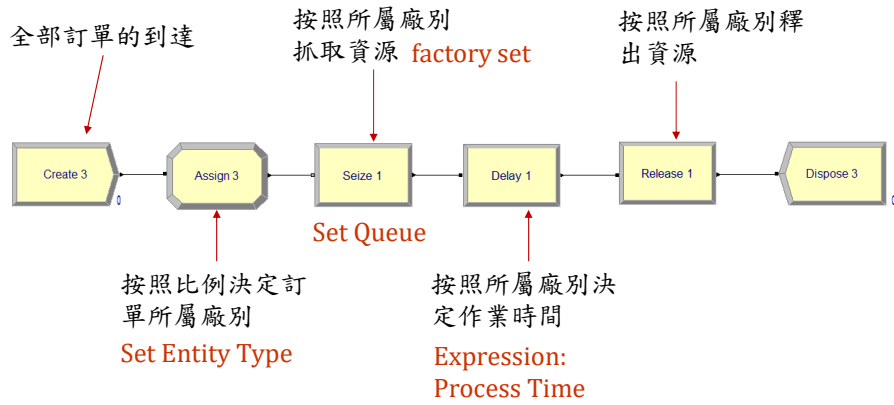
  

Members	
	Queue Name
1	Queue 1
2	Queue 2
3	Queue 3
4	Queue 4
5	Queue 5

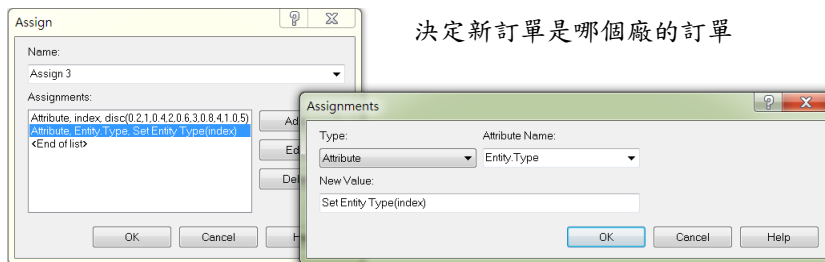
Set Entity Type

Set Queue

## 各廠一般訂單的共用流程

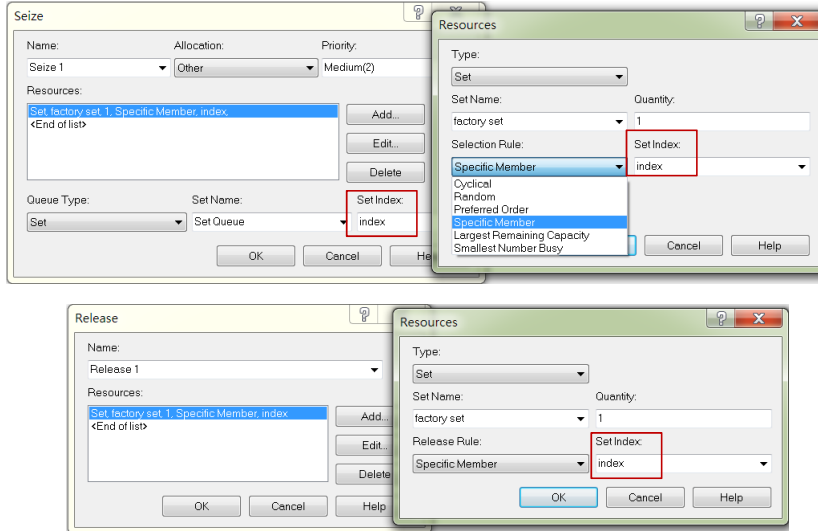


## 屬性index與entity type的對應

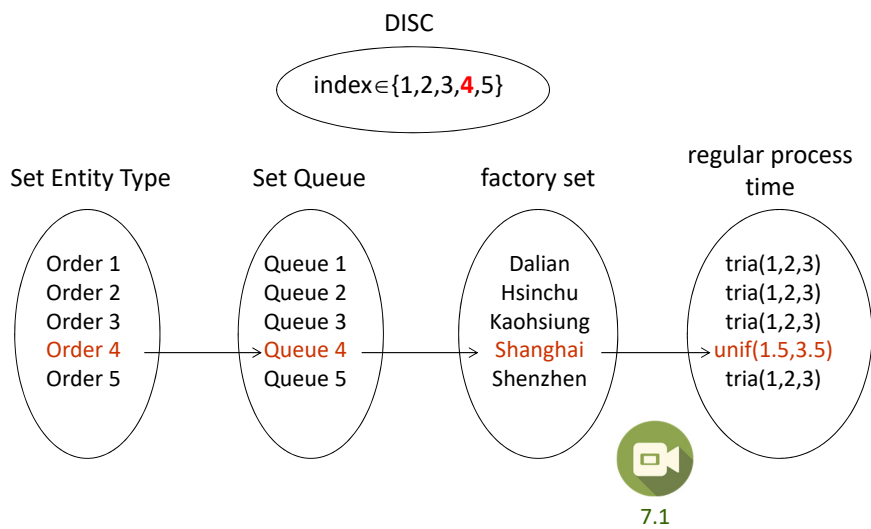




## 以index屬性區分不同廠的生產流程

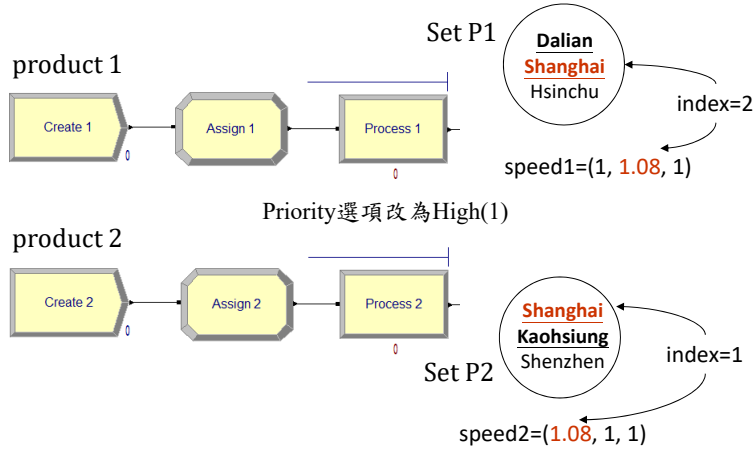


## 集合之間的對應關係



## 緊急訂單流程

Shanghai廠因管理問題，需要比其他廠多花費8%的時間來處理緊急訂單



## 緊急訂單按照優先順序分配

Set - Basic Process			
Name	Type	Members	
1	Set P1	Resource	3 rows
2	Set P2	Resource	3 rows
3	factory set	Resource	5 rows
4	Set Entity Type	Entity Type	5 rows

Members	
Resource Name	
1	Dalian
2	Shanghai
3	Hsinchu

優先順序 ↓

**Process**

Name: Process 1    Type: Standard

Logic: Action: Seize Delay/Release    Priority: High(1)

Resources: Set SetP1. 1 Preferred Order. index

Delay Type: Expression    Units: Days

Expression: order\_size\*speed 1(index)

Report Statistics

**Resources**

Type: Set    Quantity: 1

Set Name: Set P1

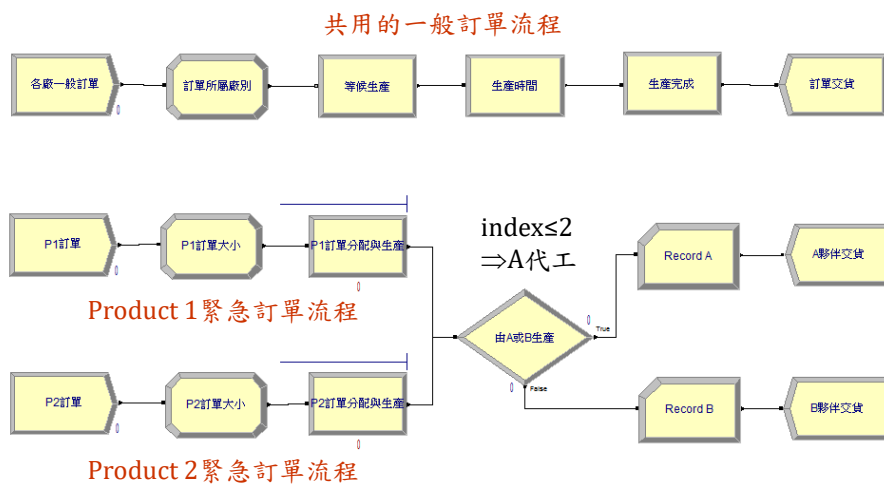
Selection Rule: Preferred Order

Save Attribute: index

## Selection Rule的不同選項

Rule	定義
Cyclical	有兩種或更多的資源可選時，依照集合的編號順序輪流選擇，類似籃球比賽爭球時的球權輪替。
Random	有兩種或更多的資源可選時，隨機選擇，不考慮其他因素。
Preferred Order	有兩種或更多的資源可選時，選擇集合編號順序最前者。
Specific Member	依照Set Index欄位的輸入，指定集合成員，必要時等候，不考慮其他成員。
Largest Remaining Capacity	當各種資源有多個單位時，選擇可用資源單位最多的成員，有兩種或更多數量相同時，選擇集合編號順序最前者。
Smallest Number Busy	當各種資源有多個單位時，選擇使用中的資源單位最少的成員，有兩種或更多數量相同時，選擇集合編號順序最前者。

## 完整的流程模組設計



## Statistics 模組定義績效指標

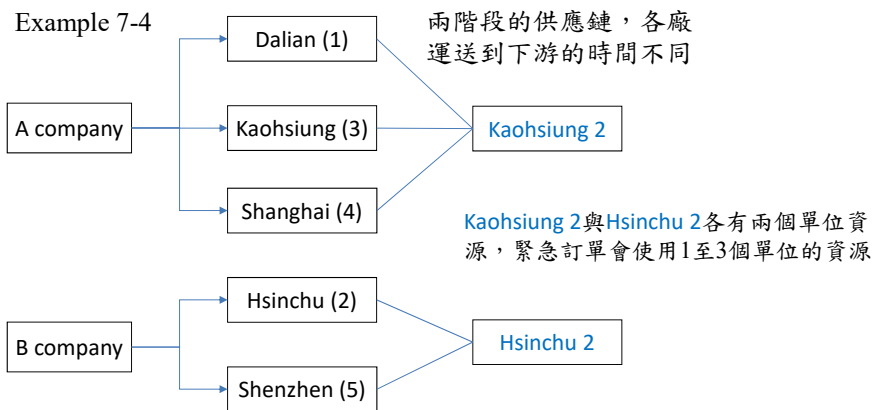
Statistic - Advanced Process					
	Name	Type	Expression	Report Label	Output File
1	product 1 time	Output	TAVG(product 1.TotalTime)	product 1 time	
2	product 2 time	Output	TAVG(product 2.TotalTime)	product 2 time	
3	ratio A	Output	$NC(\text{Record A}) / (NC(\text{Record A}) + NC(\text{Record B}))$	ratio A	
4	avg order 1 time	Output	TAVG(Order 1.TotalTime)	avg order 1 time	
5	avg order 2 time	Output	TAVG(Order 2.TotalTime)	avg order 2 time	
6	avg order 3 time	Output	TAVG(Order 3.TotalTime)	avg order 3 time	
7	avg order 4 time	Output	TAVG(Order 4.TotalTime)	avg order 4 time	
8	avg order 5 time	Output	TAVG(Order 5.TotalTime)	avg order 5 time	

## Example 7-3 模擬結果

- replication length: 300 days
- warm up period: 30 days
- number of replications: 30

	Average	Half-Width
Product 1 Time	2.4684	0.02890
Product 2 Time	3.5267	0.04009
Dalian Utilization	0.92475	0.01699
Hsinchu Utilization	0.92870	0.01628
Kaohsiung Utilization	0.94065	0.01837
Shanghai Utilization	0.99569	0.00402
Shenzhen Utilization	0.90453	0.02060
Order Ratio for A	0.62343	0.01709

## 7.3 Gather and Disperse

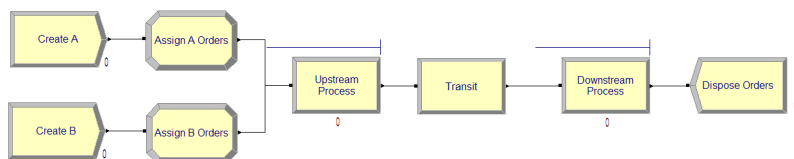


兩階段的供應鏈，各廠運送到下游的時間不同

Kaohsiung 2與Hsinchu 2各有兩個單位資源，緊急訂單會使用1至3個單位的資源

Kaohsiung也能接受product 1的緊急訂單，優先順序在Hsinchu之後，HsinChu也可生產product 2，且被列為最優先。

## 修改後的一般訂單流程

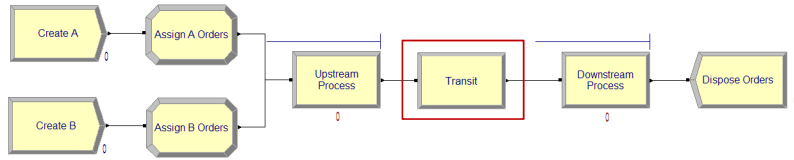


Assign	
Name:	Assign A Orders
Assignments:	Attribute, index, disc(1/3,1,2/3, 3, 1, 4) Attribute, destination, 1 Attribute, Entity Type, Set Entity Type(index) [End of List]

設定A公司的上游廠編號為1、3或4，下游目的地destination為1，以及更改entity type以利Arena的自動統計。

另一個Assign模組設定B公司的上游廠編號為2或5，下游目的地編號為2。

## 設定上游廠到下游廠的運送時間



Expression - Advanced Process							Expression Values			
Name	Comment	Rows	Columns	Data Type	File Name	Expression Values	1	2	3	
1	transit time	5	2	Native		10 rows	1	3	3	
2	transit time 2	5		Native		5 rows	2	0.5	1	
3	regular process time	5		Native		5 rows	3	1	0.5	
							4	3	3	
							5	3	3	

定義名稱為transit time的二維expression，  
Row對應到上游廠的編號index，  
Column對應下游目的地編號destination，

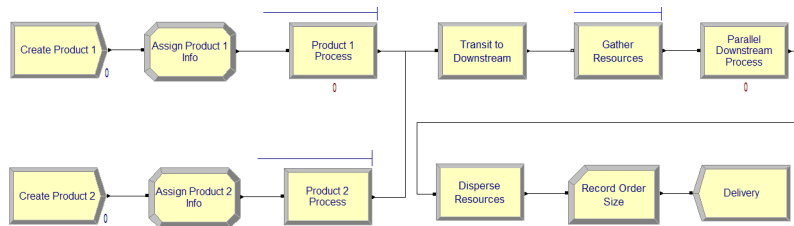
Delay ? X

Name:  Allocation:

Delay Time:  Units:

OK Cancel Help

## 修改後的緊急訂單流程



Variable - Basic Process												
Name	Comment	Rows	Columns	Data Type	Clear Option	File Name	Initial Values					
1	speed 1	4		Real	System		3 rows	Initial Values				
2	speed 2	4		Real	System		3 rows	1	2	3	4	
3	group ID	2	4	Real	System		8 rows	1	1	4	2	3
4	member ID	2	4	Real	System		8 rows	2	2	4	3	5

透過member ID擔任的對照表功能決定運送時間transit time 2(member ID(type, index))

type=2, index=4  $\Rightarrow$  member ID=5

Delay ? X

Name:  Allocation:

Delay Time:  Units:

OK Cancel Help

## Gather與Disperse模組的設定

**Gather**

Name: Gather Resources | Allocation: Other | Priority: High(1)

Set Type: Set | Unit Selection Rule: Largest Remaining Capa | Save Attribute: number

Resource Set Name: downstream set | Resource State: [ ]

Minimum Required Units: 1 | Maximum Desired Units: 3

Queue Type: Queue | Queue Name: Gather Resources.Queue

OK Cancel Help

**Disperse**

Name: Disperse Resources

Type: Set | Units to Disperse: [ ]

Resource Set Name: downstream set | Release Order Rule: Last Seized First Released

OK Cancel Help

Minimum Required Units與Maximum Desired Units  
的欄位則填入最少以及最多需要的資源數量，因此使用的資源可能來自不同地點

## 二維變數對照表與Counter Set

Variable - Basic Process							
Name	Comment	Rows	Columns	Data Type	Clear Option	File Name	Initial Values
1 speed 1		4		Real	System		3 rows
2 speed 2		4		Real	System		3 rows
3 group ID		2	4	Real	System		8 rows
4 member ID		2	4	Real	System		8 rows

Initial Values				
	1	2	3	4
1	1	1	2	1
2	2	2	1	2

Set - Basic Process				
Name	Type	Member Definition Method	Members	
1 Set P1	Resource	Manual List	4 rows	
2 Set P2	Resource	Manual List	4 rows	
3 factory set	Resource	Manual List	5 rows	
4 Set Entity Type	Entity Type	Manual List	5 rows	
5 downstream set	Resource	Manual List	2 rows	
6 Counter Set	Counter	Manual List	2 rows	

Members	
Counter Name	
Record A	
Record B	

由group ID(type, index)判定所屬的公司，並分別記錄  
type=1, index=3 ⇒ group ID=2

**Record**

Name: Record Order Size

Statistic Definition:  
Count (index, type, type, Counter Set, group ID, type, index)  
 Expression, number, No. avg gathering  
 <End of list>

OK Cancel

**Statistic Definition**

Type: Count

Value: order\_size

Counter Set Name: Counter Set | Set Index: group ID(type,index)

Record into Set

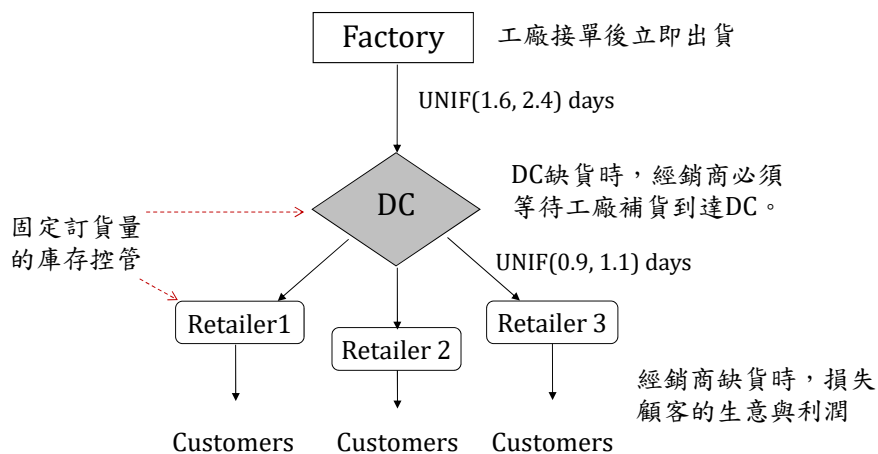
OK Cancel Help

## Example 7-4 模擬結果

與Example 7-3相同實驗設定

	Average	Half-Width
Product 1 Time	7.599	0.085
Product 2 Time	9.264	0.138
ratio A	0.585	0.014
Dalian Utilization	0.929	0.024
Hsinchu Utilization	0.959	0.011
Kaohsiung Utilization	0.967	0.012
Shanghai Utilization	0.992	0.005
Shenzhen Utilization	0.910	0.023
Hsinchu 2 Utilization	0.804	0.014
Kaohsiung 2 Utilization	0.899	0.014
avg gathering	1.367	0.04

## 7.4 三階層供應鏈庫存管理





## 供應鏈庫存管理流程

- 供應鏈的流程可分為三部份：經銷商銷售流程、經銷商向DC進貨的流程、以及DC向工廠進貨的流程。
- 三個流程相互關聯：經銷商銷售後可能因庫存降至訂貨點而啟動向DC補貨的流程，DC出貨給經銷商後可能啟動向工廠訂貨的流程，工廠補貨到達DC後可能啟動出貨給欠貨待補經銷商的流程。
- 模擬的目的是調整各成員的庫存控管參數，期望以較低的整體成本達到較高的供貨水準(fill rate)。

## 績效定義

平均單日的庫存相關成本包含：

- 全體成員的庫存持有成本，包含在途庫存，單位持有成本為\$0.3
- 經銷商的缺貨損失，單位缺貨損失為\$3
- 經銷商的訂貨成本，單次成本為\$64
- DC的訂貨成本，單次成本為\$200

供貨水準 (Fill Rate)

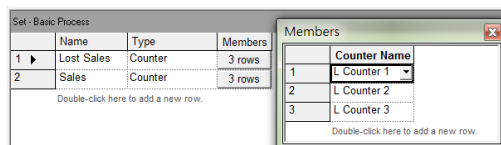
- 全體經銷商可滿足的需求比例

## 供應鏈模擬的變數設定

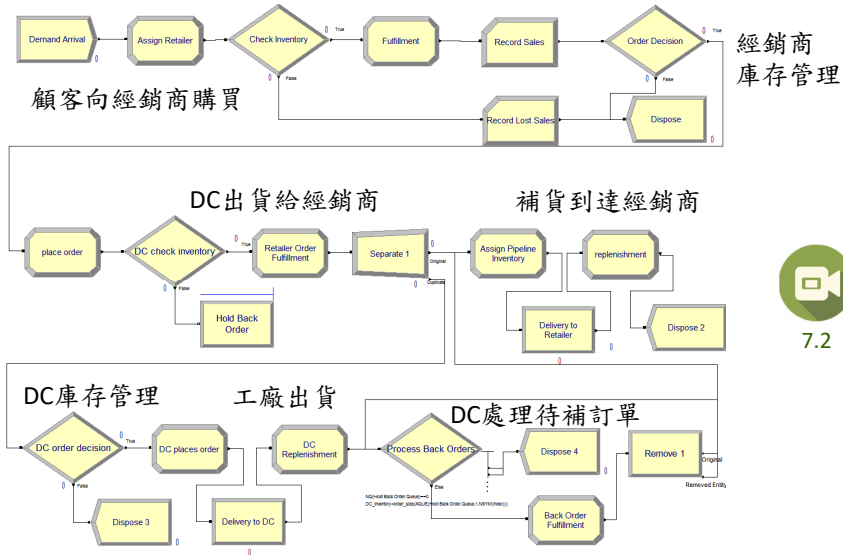
Variable	維度	定義	起始值
inventory	index=1,2,3	經銷商現有庫存量	30, 25, 25
order_size	index=1,2,3	經銷商下單的固定訂貨量	60, 50, 50
reorder_pt	index=1,2,3	經銷商訂貨點	20, 20, 20
on_order	index=1,2,3	尚未送達經銷商之訂購量	
order_count	index=1,2,3	經銷商的訂單次數	
DC_inventory		DC現有庫存量	80
DC_order_size		DC每次下單的訂貨量	200
DC_reorder_pt		DC訂貨點	60
DC_on_order		尚未送達DC之訂購量	
DC_backorder		DC尚欠經銷商之數量	
DC_order_count		DC的訂單次數	
pipeline		DC運往經銷商的在途庫存	

## 以集合方式記錄經銷商的銷售與損失

Counter set	定義
Lost Sales(index)	累計經銷商的生意損失
Sales(index)	累計經銷商的銷售量

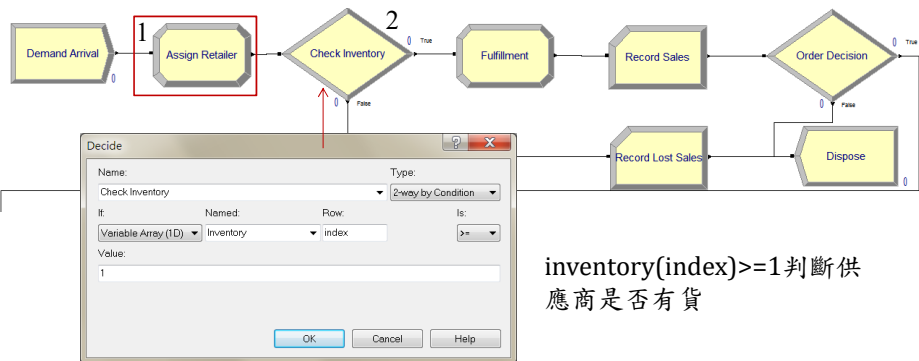


## 三階層供應鍊的流程模組

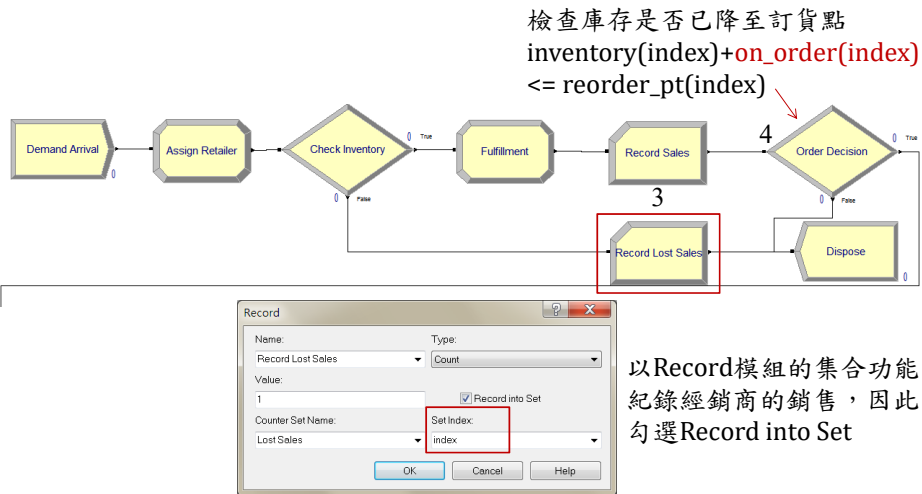


## 顧客需求到達經銷商(1, 2)

經銷商的銷售比例為40% : 35% : 25%，因此在此Assign模組以DISC(0.4,1,0.75,2,1,0,3)隨機決定顧客屬性index。



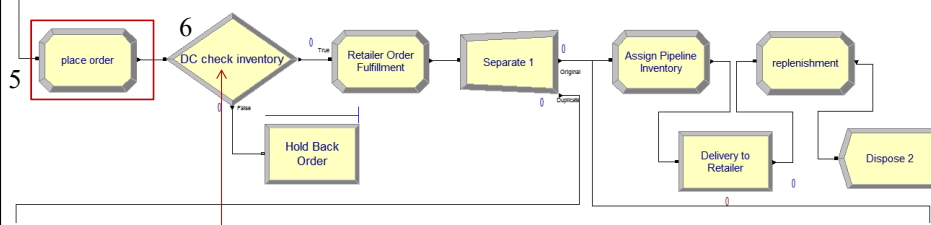
## 經銷商紀錄銷售損失與檢查庫存(3, 4)



## 經銷商向DC發出訂單(5, 6)

entity type=order

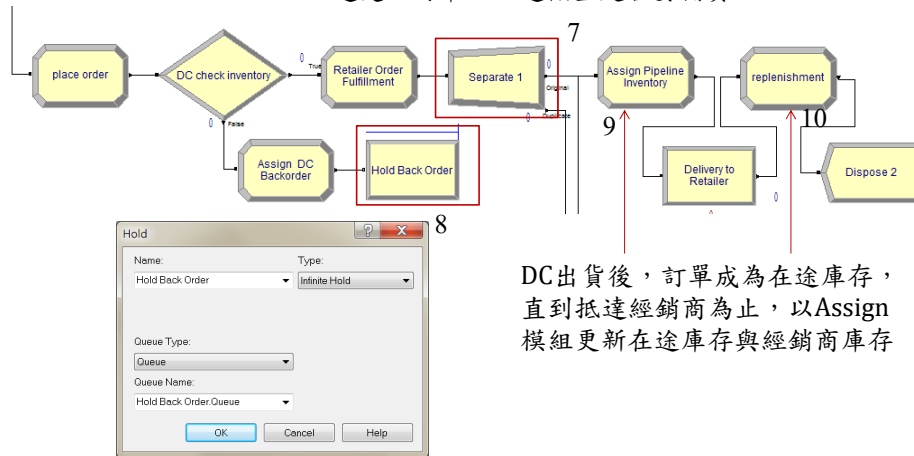
更新訂購次數order\_count(index)、  
 尚未到的已訂購量on\_order(index)、  
 紀錄訂單產生的時間order time



DC檢查是否有足夠庫存可出貨給經銷商，  
 $DC\_inventory(index) \geq order\_size(index)$

## DC處理訂單的流程 (7-10)

Separate模組將流程分為兩邊，一邊處理訂單，一邊檢查是否要補貨

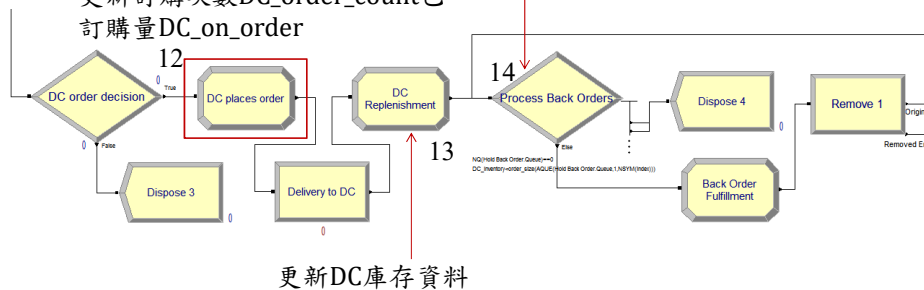


DC出貨後，訂單成為在途庫存，直到抵達經銷商為止，以Assign模組更新在途庫存與經銷商庫存

## DC向工廠補貨的流程 (12-14)

entity type=order  
更新訂購次數DC\_order\_count已  
訂購量DC\_on\_order

檢查有無待補之經銷商訂單，  
 $NQ(\text{Hold Back Order.Queue})=0$



更新DC庫存資料

## DC處理經銷商之欠補訂單(14)

Decide

Name: Process Back Orders

Type: N-way by Condition

Conditions:

Expression: `NO(Hold Back Order Queue) == 0`  
`DC_inventory<order_size>(AQUE[Hold Back Order Queue.1.NSYM(index)])`  
 <End of list>

1. 判斷是否有待補的訂單

2. 判斷DC是否有足夠庫存可供應第一順位的待補訂單

AQUE: 查詢等候線特定排序的特定屬性

NSYM(index): index屬性的代碼

Expression Builder

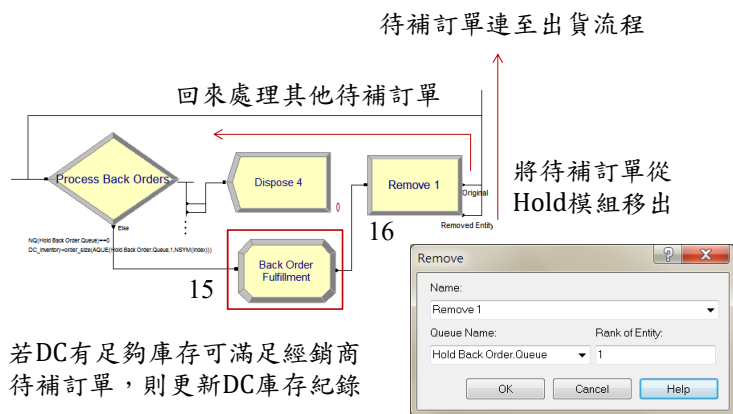
Queue Name: Hold Back Order Queue

Attribute Name: index

Entity Rank In Queue: 1

Current Expression: `DC_inventory<order_size>(AQUE[Hold Back Order Queue.1.NSYM(index)])`

## DC處理待補訂單出貨(15, 16)



## Statistics 模組設定系統績效

	Expression 定義計算方式
<b>Total Inventory</b> 供應鏈總庫存	Delivery to DC.WIP*DC_order_size + pipeline + DC_Inventory+Inventory(1)+Inventory(2)+Inventory(3)

	Expression 定義計算方式
<b>avg fill rate</b> 平均供貨水準	$(NC(S \text{ Counter } 1)+NC(S \text{ Counter } 2)+NC(S \text{ Counter } 3)) / ((NC(S \text{ Counter } 1)+NC(S \text{ Counter } 2)+NC(S \text{ Counter } 3)+NC(L \text{ Counter } 1)+NC(L \text{ Counter } 2)+NC(L \text{ Counter } 3))$
<b>avg unit cost</b> 平均單日成本	$((NC(L \text{ Counter } 1)+NC(L \text{ Counter } 2)+NC(L \text{ Counter } 3))*3 + (order\_count(1)+order\_count(2)+order\_count(3))*64+DC\_order\_count*200)/2970+ DAVG(Total Inventory)*0.3$

## 三層供應鏈模擬結果

使用 Batch Means 的實驗方式：設定 warm up period=30 days，replication length=3000 days。

	Average	Half-width
avg fill rate	.9979	
avg unit cost	145.14	
Total Inventory	306.61	2.9893
dc inv	119.28	2.7886
pipeline inv	72.110	1.1674
inv1	40.044	0.3621
inv2	36.490	0.3164
inv3	38.681	0.4004

Example 7-5