Flow, space and activity relationships I.

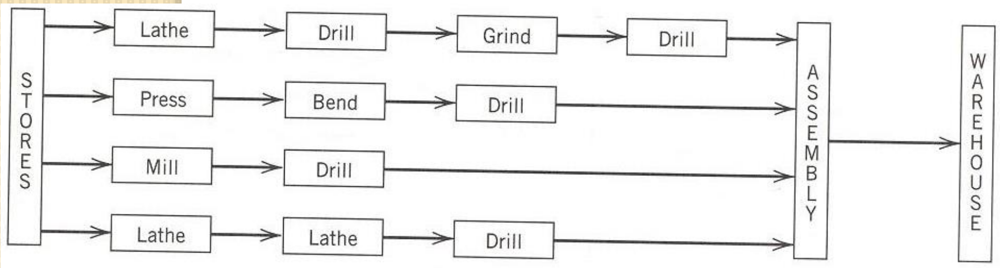
- Departmental Planning
- Manufacturing cells
  - Clustering algorithms for cell formation

Flow, Space and Activity Relationships

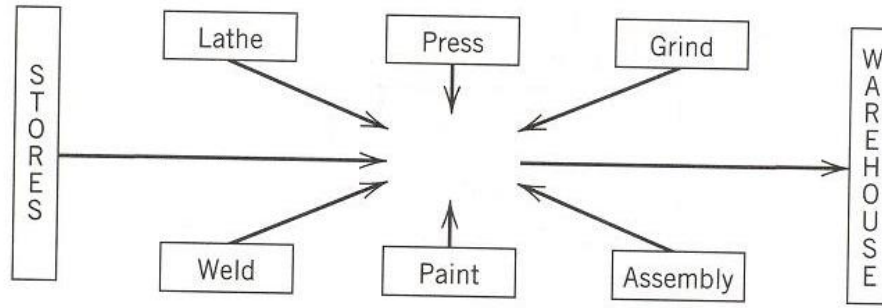
- Flow
  - Flow into, within and from manufacturing facility
  - Flow of materials, people, equipment, information, money, etc.
- Space
  - The amount of space required in the facility
  - Workstation specification, department specification and other space requirements
- Activity relationships
  - Activity relationship is the key input in facilities design
  - Defined by flow relationships, organizational relationships, environmental relationships, process relationships and control relationships

Departmental planning

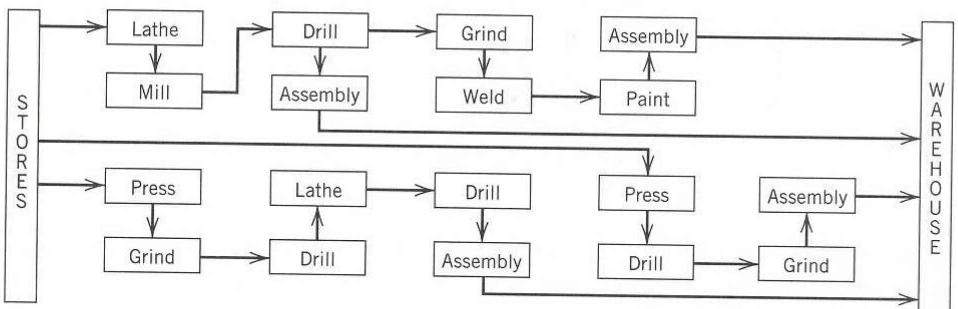
- Production planning departments are collections of workstations to be grouped together during the facilities layout process
- Combining workstations that perform similar functions:
  - Similar products or components
  - Similar processes
- Classification of layouts based on product volume-variety:
  - Product layout (flow shop)
  - Fixed product layout
  - Group layout
  - Process layout (job shop)



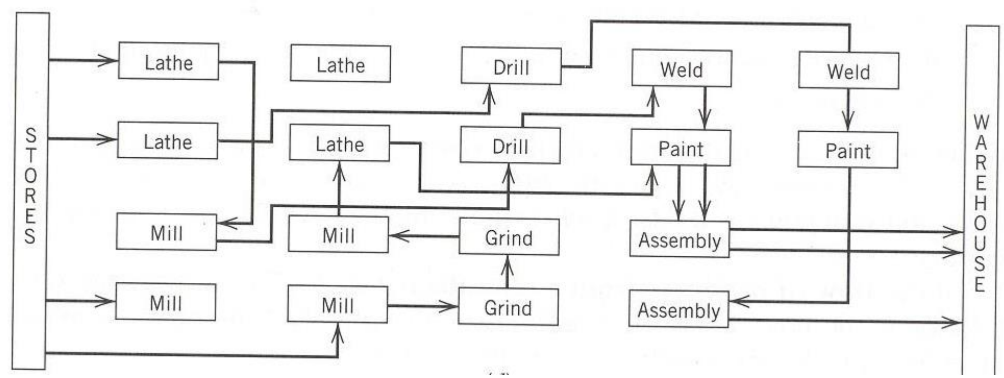
Product layout (Flow shop)



Fixed product layout



Group layout

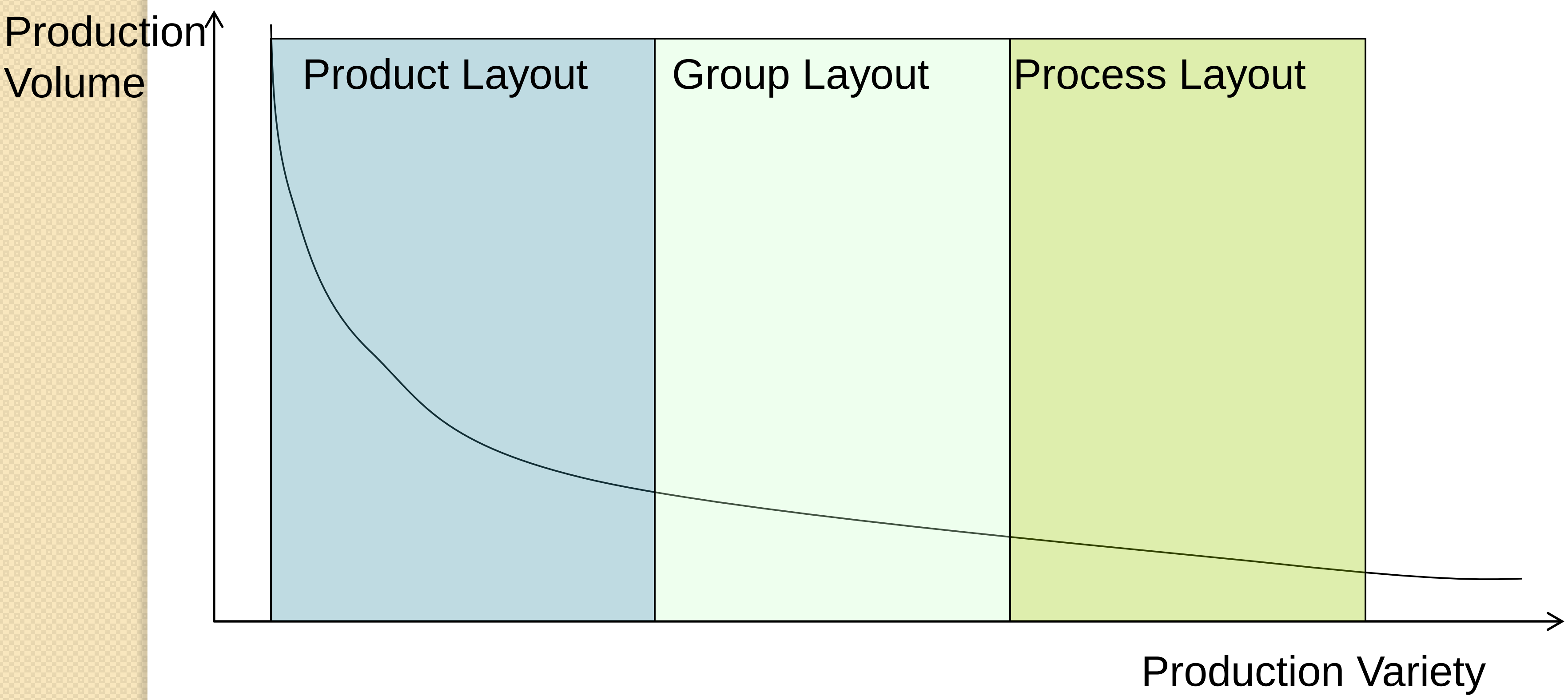


Process layout (Job shop)

Departmental planning

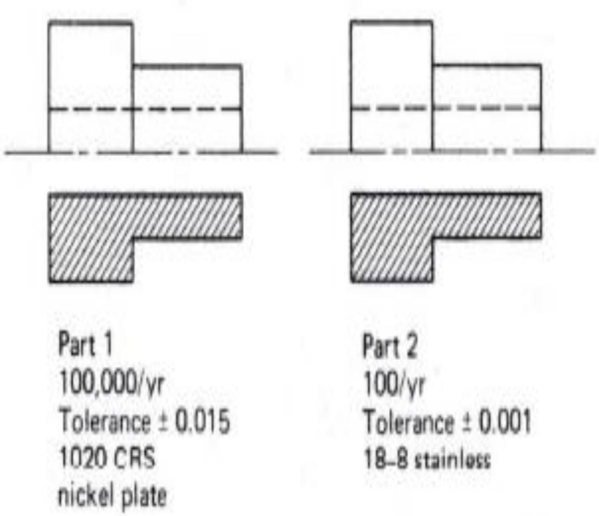
Product	Layout	Combining workstations
•Standardized •Large stable demand	•**Product layout** (flow shop)	•Combine all workstations required to produce the product
•Physically large •Awkward to move •Low sporadic demand	•**Fixed product layout**	•Combine all workstations required to produce the product with the area required for staging the product
•Capable of being grouped into families of similar parts	•**Group layout** (product family layout)	•Combine all workstations required to produce the family of products
•None of the above	•**Process layout** (job shop)	•Combine identical workstations into departments •Combine similar departments

Production volume and product variety determines type of layout

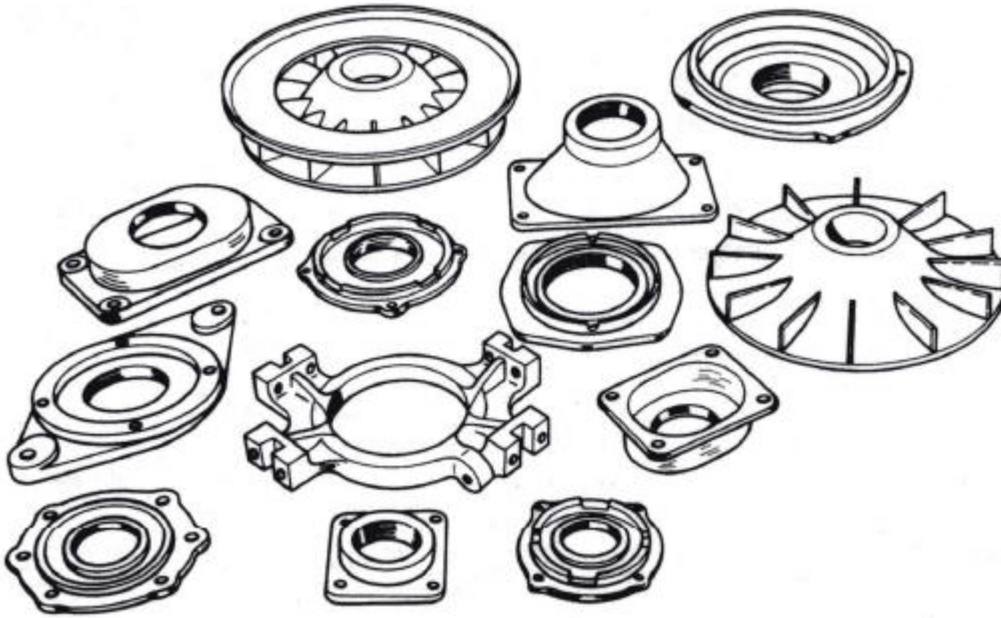


Group Technology – Cellular Manufacturing

- *Group technology (product family) departments* aggregate medium volume-variety parts into families based on **similar manufacturing operations and design attributes.**



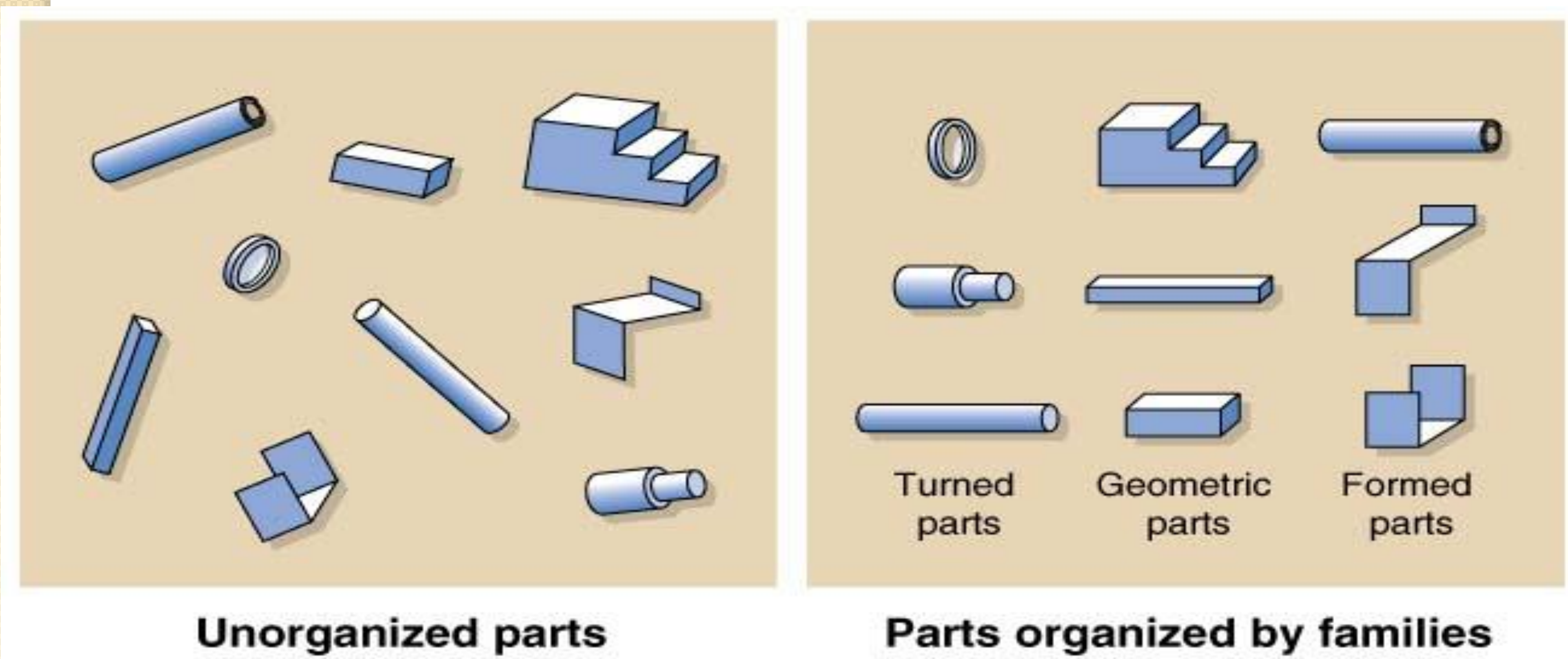
Similar design attributes,
different manufacturing requirements



Different design attributes,
similar manufacturing requirements

Group Technology – Cellular Manufacturing

- *Group technology (product family) departments* aggregate medium volume-variety parts into families based on **similar manufacturing operations and design attributes.**



Group Technology – Cellular Manufacturing

- *Manufacturing cells* group machines, employees, materials, tooling and material handling and storage equipment to produce families of parts.
- Manufacturing cell operation needs minimum external support
- Often designed, controlled and operated using *JIT, TQM* and *TEI*
- Benefits of cell manufacturing:
  - **Reduction:** inventories, space, paperwork, equipment, transportation, etc.
  - **Simplification:** communication, handling, scheduling, etc.
  - **Improvement**: productivity, flexibility, quality, customer satisfaction, etc.

Cellular Manufacturing

- Evaluation of cell design decisions

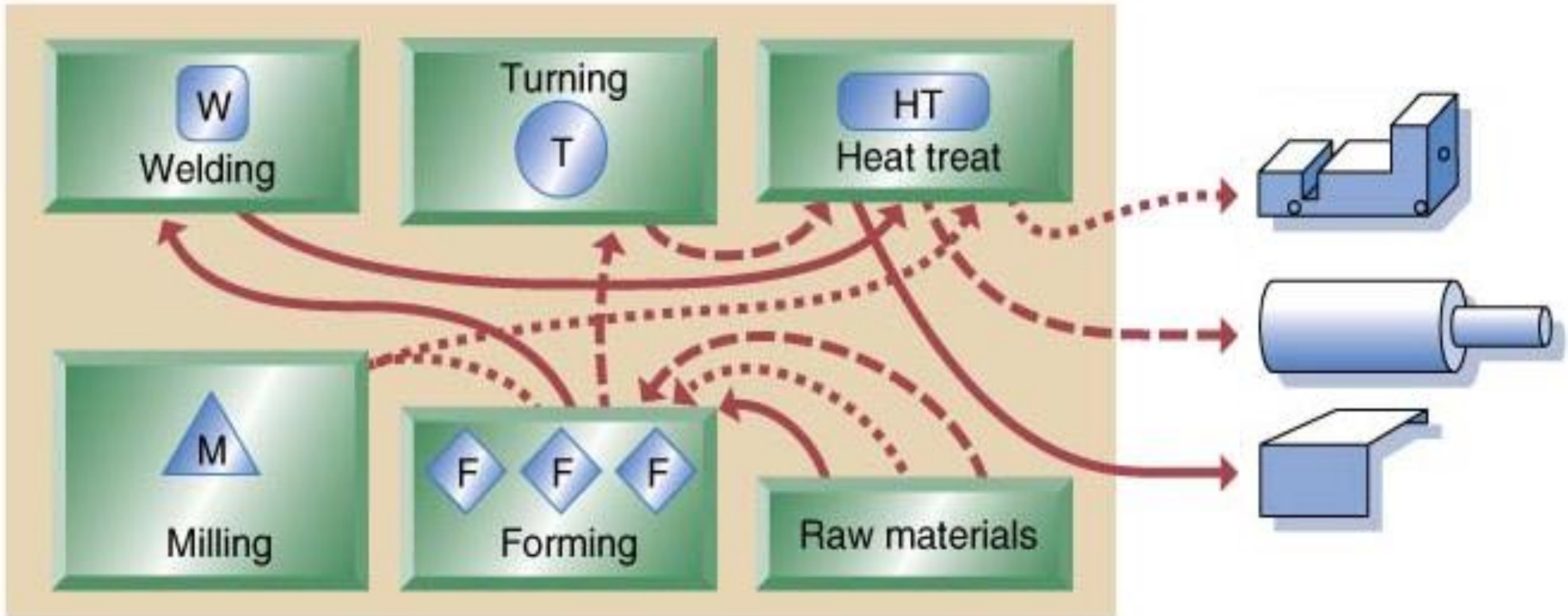
System structure

Equipment and tooling investment	Low
Equipment relocation cost	Low
Inter and intra cell material handling cost	Low
Floor space requirement	Low
Extend to which parts are completed in a cell	High
Flexibility	High

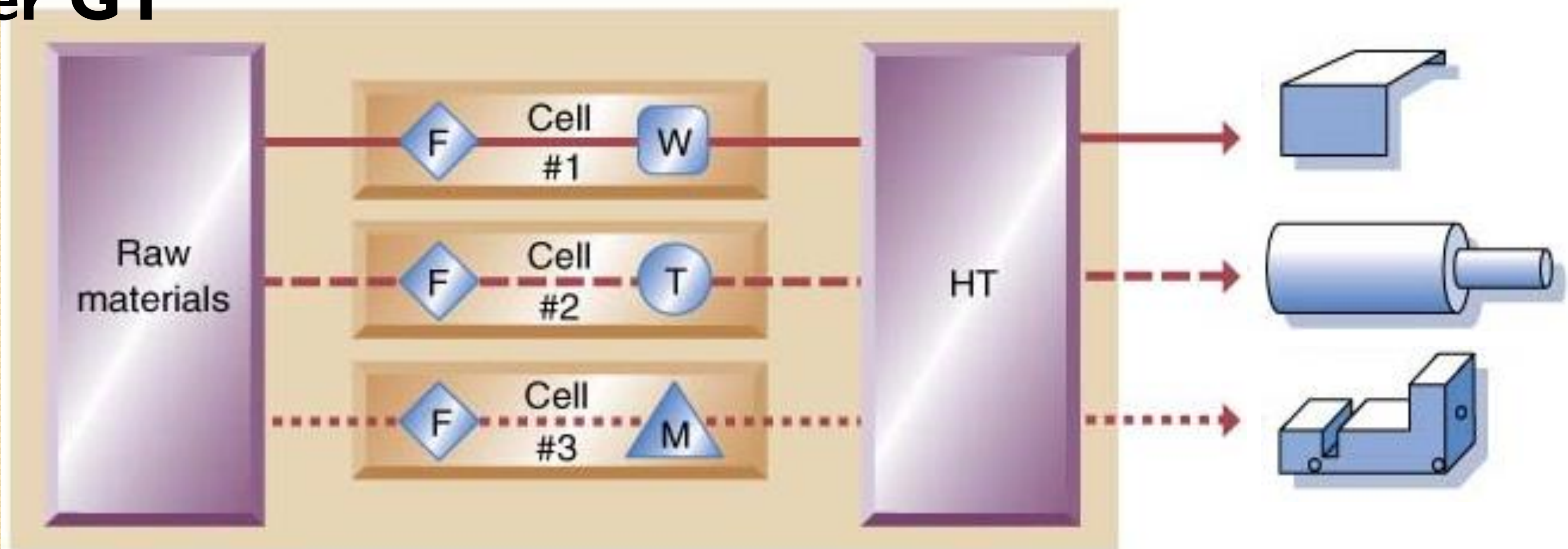
System operation

Equipment utilization	High
Work-in-process inventory	Low
Queue lengths at each work station	Low
Job throughput time	Short
Job lateness	Low

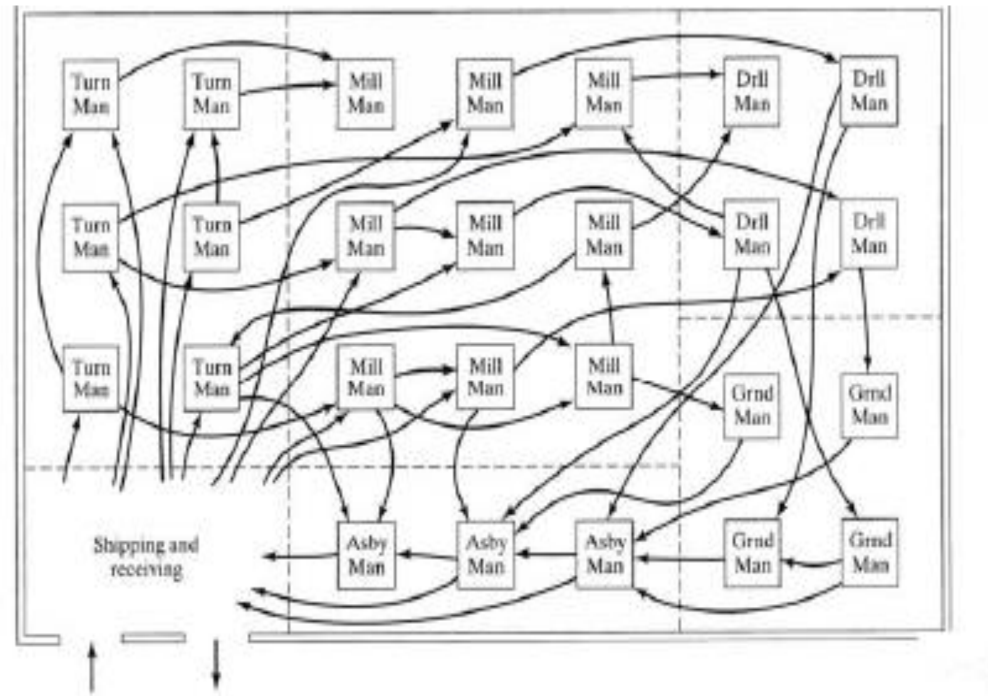
Before GT



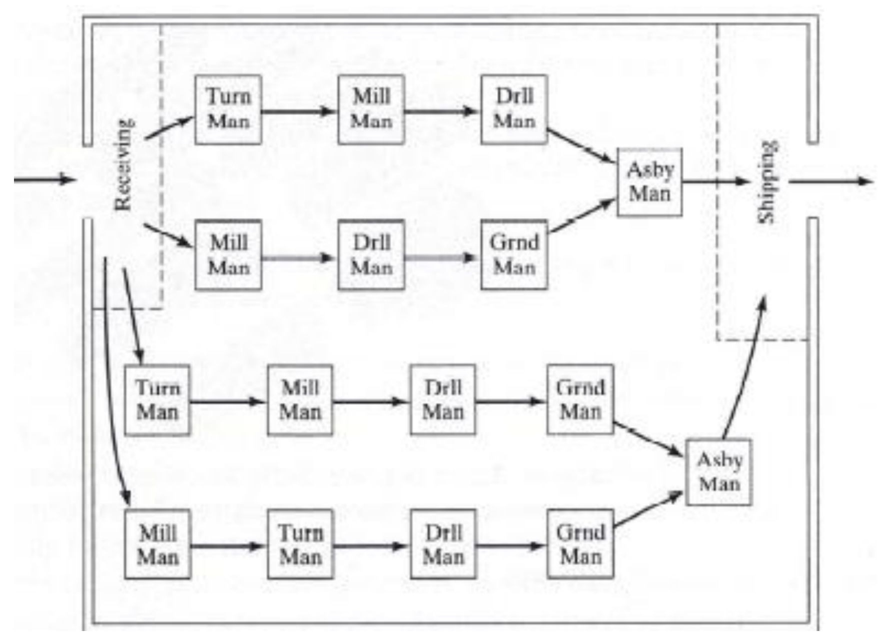
After GT



- Before GT



- After GT



Manufacturing cell forming

- Successful implementation requires addressing selection, cell design, cell operation and cell control issues
- Manufacturing cell forming:
  - Classification
  - Production flow analysis
  - Clustering methodologies
  - Heuristic procedures
  - Mathematical models
- Cell forming is seldom the responsibility of a facility planner

Clustering methodologies

- Group parts together so that they can be processed as a family
- Links parts and machines in

  machine-part matrix

	Machine #				
Part #	1	2	3	4	5
1	1		1		
2	1				
3		1		1	1
4	1		1		
5		1			
6				1	1

Machine-part matrix

Clustering methodologies

- Group parts together so that they can be processed as a family
- Links parts and machines in

  machine-part matrix

	Machine #				
Part #	5	4	2	3	1
3	1	1	1		
6	1	1			
5			1		
4				1	1
1				1	1
2					1

Machine-part matrix

Direct clustering algorithm (DCA)

1. Form machine-part matrix
2. Sum the 1s in each column & row
3. Order the rows in descending order
4. Order the columns in ascending order
5. Sort the columns (1 in the first row moves left, then in the second row, etc.)
6. Sort the rows (1 in the first column moves upward, then in the second column, etc.)
7. Form sells

Direct clustering algorithm

2. Sum the 1s in each column & row

3. Order the rows in descending order
4. Order the columns in ascending order

Part #	Machine # 1	2	3	4	5	#
1	1		1			2
2	1					1
3		1		1	1	3
4	1		1			2
5		1				1
6				1	1	2
# of 1s	3	2	2	2	2	

Part #	Machine # 5	4	3	2	1	# of 1s
3	1	1		1		3
6	1	1				2
4			1		1	2
1			1		1	2
5				1		1
2					1	1
# of 1s	2	2	2	2	3	

Direct clustering algorithm

5. Sort the columns (1 in the first row moves the column to the left, then 1 in the second row, etc.)

Part #	5	4	3	2	1	# of 1s
3	1	1		1		3
6	1	1				2
4			1		1	2
1			1		1	2
5				1		1
2					1	1
# of 1s	2	2	2	2	3	

Figure 3.3 Ordered machine-part mat

Machine #

Part #	5	4	2	3	1	# of 1s
3	1	1	1			3
6	1	1				2
4				1	1	2
1				1	1	2
5			1			1
2					1	1
# of 1s	2	2	2	2	3	

Figure 3.4 Column-sorted machine-part matrix.

Direct clustering algorithm

6. Sort the rows (1 in the first column moves the row upward, then 1 in the second column, etc.)

Machine #

Part #	5	4	2	3	1	# of 1s
3	1	1	1			3
6	1	1				2
4				1	1	2
1				1	1	2
5			1			1
2					1	1
# of 1s	2	2	2	2	3	

Figure 3.4 Column-sorted machine-part matrix.

Machine #

Part #	5	4	2	3	1	# of 1s
3	1	1	1			3
6	1	1				2
5			1			1
4				1	1	2
1				1	1	2
2					1	1
# of 1s	2	2	2	2	3	

Figure 3.5 Row-sorted machine-part matrix.

Direct clustering algorithm

7. Form cells

- Cell #1:
  - Machines 2,4 and 5
- Cell #2:
  - Machines 1and 3

	Machine #				
Part #	5	4	2	3	1
3	1	1	1		
6	1	1			
5			1		
4				1	1
1				1	1
2					1

Direct clustering algorithm
"bottleneck machines"

Part #	Machine # 1	2	3	4	5	# of 1s
1	1		1			2
2	1					1
3		1		1	1	3
4	1		1			2
5		1	1			2
6				1	1	2
# of 1s	3	2	3	2	2	

Figure 3.7 Machine-part matrix.

DCA

Part #	Machine # 5	4	2	3	1	# of 1s
3	1	1	1			3
6	1	1				2
5			1	1		1
4				1	1	2
1				1	1	2
2					1	1
# of 1s	2	2	2	2	3	

Figure 3.8 Ordered machine-part matrix

- Machine 2 which is needed for parts #3 and #5 creates conflicts!

Direct clustering algorithm
“bottleneck machines”

Part #	Machine # 5	4	2	3	1
3	1	1	1		
6	1	1			
5			1	1	
4				1	1
1				1	1
2					1

(a)

Part #	Machine # 5	4	2	3	1
3	1	1	1		
6	1	1			
5			1	1	
4				1	1
1				1	1
2					1

(b)

Part #	Machine # 5	4	2	3	1
3	1	1	1		
6	1	1			
5			1	1	
4				1	1
1				1	1
2					1

(c)

What can we do?

Direct clustering algorithm "bottleneck machines"

- Possible solutions:
  - Locate the bottleneck machines close to each other :
    - in different cells
    - at the boundary between cells
  - Redesign the parts
  - Outsource the parts
  - Duplicate machines

Direct clustering algorithm "bottleneck machines"

- Duplicating machines

Machine #

Part #	5	4	2a	2b	3	1
3	1	1	1			
6	1	1				
5				1	1	
4					1	1
2					1	1
1						1

Duplicated machine 2

Part #	5	4	2	3a	3b	1
3	1	1	1			
6	1	1				
5			1	1		
4					1	1
2					1	1
1						1

Duplicated machine 3

Direct clustering algorithm "bottleneck machines"

Machine #

Part #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	# of 1s
1	1		1		1			1	1		1						1	1				1					9
2	1		1		1			1	1		1						1	1				1					9
3	1		1		1			1	1		1						1	1				1					9
4	1		1		1			1	1		1						1	1				1					9
5	1		1			1		1	1		1						1	1				1					9
6	1		1			1		1	1		1						1	1				1					9
7	1		1			1		1	1		1						1	1				1					9
8		1		1			1			1		1							1	1			1				8
9		1		1			1			1		1							1	1			1				8
10	1		1										1	1	1	1					1			1	1	1	10
11	1		1										1	1	1	1					1			1	1	1	10
12	1		1										1	1	1	1					1			1	1	1	10
13	1		1										1	1	1	1					1			1	1	1	10
# of 1s	11	2	11	2	4	3	2	7	7	2	7	2	4	4	4	4	7	7	2	2	4	7	2	4	4	4	

Figure 3.11 Machine-part matrix.

Machine #

DCA

Part #	3	1	26	25	24	21	16	15	14	13	22	18	17	11	9	8	6	5	23	20	19	12	10	7	4	2	# of 1s
13	1	1	1	1	1	1	1	1	1	1																	10
12	1	1	1	1	1	1	1	1	1	1																	10
11	1	1	1	1	1	1	1	1	1	1																	10
10	1	1	1	1	1	1	1	1	1	1																	10
7	1	1									1	1	1	1	1	1	1										9
6	1	1									1	1	1	1	1	1	1										9
5	1	1									1	1	1	1	1	1	1										9
4	1	1									1	1	1	1	1	1		1									9
3	1	1									1	1	1	1	1	1		1									9
2	1	1									1	1	1	1	1	1		1									9
1	1	1									1	1	1	1	1	1		1									9
9																			1	1	1	1	1	1	1	1	8
8																			1	1	1	1	1	1	1	1	8
# of 1s	11	11	4	4	4	4	4	4	4	4	7	7	7	7	7	7	3	4	2	2	2	2	2	2	2	2	

- Create only 2 cells

Part #	3	1	26	25	24	21	16	15	14	13	22	18	17	11	9	8	6	5	23	20	19	12	10	7	4	2
13	1	1	1	1	1	1	1	1	1	1																
12	1	1	1	1	1	1	1	1	1	1																
11	1	1	1	1	1	1	1	1	1	1																
10	1	1	1	1	1	1	1	1	1	1																
7	1	1									1	1	1	1	1	1	1									
6	1	1									1	1	1	1	1	1	1									
5	1	1									1	1	1	1	1	1	1									
4	1	1									1	1	1	1	1	1		1								
3	1	1									1	1	1	1	1	1		1								
2	1	1									1	1	1	1	1	1		1								
1	1	1									1	1	1	1	1	1		1								
9																			1	1	1	1	1	1	1	1
8																			1	1	1	1	1	1	1	1

Cell A, Cell B

(a)

- Place the bottleneck machines at the boundary of the cells

Machine #

Part #	3	1	26	25	24	21	16	15	14	13	22	18	17	11	9	8	6	5	23	20	19	12	10	7	4	2
13	1	1	1	1	1	1	1	1	1	1																
12	1	1	1	1	1	1	1	1	1	1																
11	1	1	1	1	1	1	1	1	1	1																
10	1	1	1	1	1	1	1	1	1	1																
7	1	1									1	1	1	1	1	1	1									
6	1	1									1	1	1	1	1	1	1									
5	1	1									1	1	1	1	1	1	1									
4	1	1									1	1	1	1	1	1		1								
3	1	1									1	1	1	1	1	1		1								
2	1	1									1	1	1	1	1	1		1								
1	1	1									1	1	1	1	1	1		1								
9																			1	1	1	1	1	1	1	1
8																			1	1	1	1	1	1	1	1

Cell A, Cell B, Cell C

(b)

- Duplicate the bottleneck machines

Machine #

Part #	26	25	24	21	16	15	14	13	3a	1a	22	18	17	11	9	8	3b	1b	6	5	23	20	19	12	10	7	4	2
13	1	1	1	1	1	1	1	1	1	1																		
12	1	1	1	1	1	1	1	1	1	1																		
11	1	1	1	1	1	1	1	1	1	1																		
10	1	1	1	1	1	1	1	1	1	1																		
7											1	1	1	1	1	1	1	1	1									
6											1	1	1	1	1	1	1	1	1									
5											1	1	1	1	1	1	1	1	1									
4											1	1	1	1	1	1	1	1		1								
3											1	1	1	1	1	1	1	1		1								
2											1	1	1	1	1	1	1	1		1								
1											1	1	1	1	1	1	1	1		1								
9																					1	1	1	1	1	1	1	1
8																					1	1	1	1	1	1	1	1

Cell A, Cell B, Cell C

(c)

Can the problem be solved by redesigning or outsourcing?

Binary Ordering Algorithm

- Binary ordering algorithm (Rank ordering algorithm) considers the rows and columns as binary strings
- Procedure:
  1. Compute the decimal equivalent of the binary strings for rows
  2. Reorder the rows in decreasing order of their binary value
  3. Compute the decimal equivalent of the binary strings for columns
  4. Reorder the columns in decreasing order of their binary value
  5. If the machine-part matrix is unchanged, then stop, else repeat

Binary Ordering Algorithm

- Machine-part matrix

TABLE 12.3 **Machine–Component Matrix**

	Components									
Machines	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
M1	1	1	1	1	1		1	1	1	1
M2		1	1	1					1	1
M3	1				1	1	1			
M4		1	1	1				1	1	1
M5	1	1	1	1	1	1	1	1		

Binary Ordering Algorithm

1. Assign binary weights from right to left for components and calculate the decimal equivalent for each row (machine)

TABLE 12.4 **Decimal Equivalents for Each Row**

	Components										
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	
	Binary weight										*decimal*
Machines	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	*equivalent*
M_1	1	1	1	1	1		1	1	1	1	1007
M_2		1	1	1					1	1	451
M_3	1				1	1	1				568
M_4		1	1	1				1	1	1	455
M_5	1	1	1	1	1	1	1	1			1020

Binary Ordering Algorithm

2. Rank the machines according to their decimal equivalent computed in the previous step
3. Assign binary weights from bottom up for machines and calculate the decimal equivalent for each column (part)

TABLE 12.5 **Row Arrangement in Decreasing Order of the Decimal Weights**

		Components									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
	Binary weight	*Binary weight*									
Machines		2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
M_5	2^4	1	1	1	1	1	1	1	1		
M_1	2^3	1	1	1	1	1		1	1	1	1
M_3	2^2	1				1	1	1			
M_4	2^1		1	1	1				1	1	1
M_2	2^0		1	1	1					1	1
Column decimal equivalent		28	27	27	27	28	20	28	26	11	11

Binary Ordering Algorithm

4. Reorder the columns in decreasing order of their binary value
5. If the machine-part matrix is unchanged, then stop, else repeat

TABLE 12.6 **One Solution for the Example Using ROC Algorithm**

		Components										
		1	*5*	*7*	*2*	*3*	*4*	*8*	*6*	*9*	*10*	
	Binary	Binary weight										*Row decimal*
Machines	*weight*	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	*equivalent*
M_5	2^4	1	1	1	1	1	1	1	1			1020
M_1	2^3	1	1	1	1	1	1	1		1	1	1019
M_3	2^2	1	1	1					1			900
M_4	2^1				1	1	1	1		1	1	123
M_2	2^0				1	1	1			1	1	115
Column decimal equivalent		28	28	28	27	27	27	26	20	11	11	

Cluster Identification algorithm

1. Select any row and cross it
2. For each crossed 1 make a vertical line
3. For each crossed 1 make a horizontal line
4. Repeat until all the 1s are crossed by a vertical line or by a horizontal line
5. Form a cell from all the machines and components which were crossed
6. Remove all the crossed elements (machines and components) and start again

Cluster Identification algorithm

1. Select any row and cross it

	P1	P2	P3	P4	P5	P6	P7	P8
M1		1	1		1			
M2	1					1		
M3				1			1	
M4						1		
M5								1
M6				1				
M7			1		1			1

Cluster Identification algorithm

2. For each crossed 1 make a vertical line

	P1	P2	P3	P4	P5	P6	P7	P8
M1		1	1		1			
M2	1					1		
M3				1			1	
M4						1		
M5								1
M6				1				
M7			1		1			1

Cluster Identification algorithm

3. For each crossed 1 make a horizontal line

	P1	P2	P3	P4	P5	P6	P7	P8
M1		1	1		1			
M2	1					1		
M3				1			1	
M4						1		
M5								1
M6				1				
M7			1		1			1

Cluster Identification algorithm

4. Repeat until all the 1s are crossed by a vertical line or by a horizontal line

	P1	P2	P3	P4	P5	P6	P7	P8
M1		I	I		I			
M2	1					1		
M3				1			1	
M4						1		
M5								I
M6				1				
M7			I		I			I

Cluster Identification algorithm

5. Form a cell from all the machines and components which were crossed

		P2	P3		P5			P8
M1		1	1		1			
M5								1
M7			1		1			1

First cell is identified!

Cell #1 will produce parts P2, P3, P5 and P8 with Machines M1, M5 and M7

Cluster Identification algorithm

6. Remove all the crossed elements (machines and components) and start again

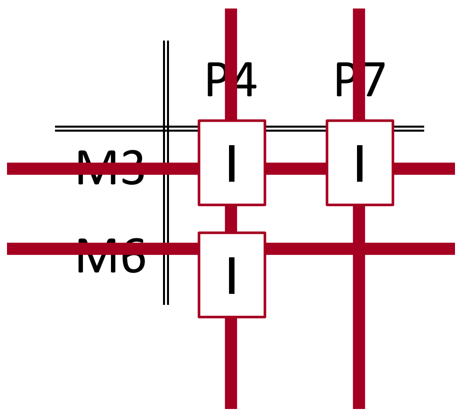
	P1	P4	P6	P7
M2	I		I	
M3		1		1
M4			I	
M6		1		

	P1		P6	
M2	1		1	
M4			1	

Cell #2 will produce parts P1 and P6 with Machines M2 and M4

Cluster Identification algorithm

6. Remove all the crossed elements (machines and components) and start again



	P4	P7
M3	1	1
M6	1	

Cell #3 will produce parts P4 and P7 with Machines M3 and M6

Cluster Identification algorithm

- Three resulting cells:

	P2	P3	P5	P8	P1	P6	P4	P7
M1	1	1	1					
M5				1				
M7		1	1	1				
M2					1	1		
M4						1		
M3							1	1
M6							1	

Cell #1

Cell #2

Cell #3

Cost Analysis Algorithm

- In the real-world cases the solution will rarely be straightforward. The algorithm may determine all the parts to be produced in one cell
- Cost Analysis Algorithm allows to:
  - determine the number of machines
  - consider the cost of subcontracting

Cost Analysis Algorithm

1. Cross a column with the highest cost
2. Make a horizontal line for each crossed 1
3. Form a group of parts which are crossed only by the horizontal lines
   - For each part apply the basic Cluster Identification Algorithm
   - Start always with the part of the highest cost
   - If the max number of machines ends up to be higher, the part is an exception (to be subcontracted)
   - See if you can continue to reach the maximum number of machines
4. Form a cell with the parts and machines which are crossed
5. Eliminate the exceptions and the pieces belonging to the created cell, form a new Machine-part matrix and start again

Cost Analysis Algorithm

1. Cross a column with the highest cost

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		1	1				1				
M2	1				1						1
M3										1	1
M4	1		1			1					
M5					1			1			
M6	1			1				1	1	1	
M7			1	1		1	1		1		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cells can have a **maximum of 4 machines**

Cost Analysis Algorithm

2. Make a horizontal line for each crossed 1

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			1				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Form a group of parts which are crossed only by the horizontal lines and for each part apply the basic algorithm

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			1				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P7 – **P7 is ACCEPTED**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			1				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P2 – **P2 is ACCEPTED**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			1				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P4 – more than 4 machines would be necessary - **P4 is an EXCEPTION**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			I				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P6 – **P6 is ACCEPTED**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	1				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	1			I				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P1 – more than 4 machines would be necessary - **P1 is an EXCEPTION**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	I				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	I			I				1	1	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

3. Apply the basic algorithm for part P9 – more than 4 machines would be necessary – **P9 is an EXCEPTION**

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1		I	I				I				
M2	I				1						1
M3										1	1
M4	I		I			I					
M5					1			1			
M6	I			I				1	I	1	
M7			I	I		I	I		I		
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Cost Analysis Algorithm

4. Form a cell with the parts and machines which are crossed

	P2	P3	P6	P7
M1	1	1		1
M4		1	1	
M7		1	1	1

Cell #1 will produce parts P2, P3, P6 and P7 with Machines M1, M4 and M7

Cost Analysis Algorithm

5. Eliminate the exceptions and the pieces belonging to the created cell, form a new Machine-part matrix and start again

	P5	P8	P10	P11
M2	1			1
M3			1	1
M5	1	1		
M6		1	1	
Cost	15	7	30	4

Cell #2 will produce parts P5, P8, P10 and P11 with Machines M2, M3, M5 and M6

Cost Analysis Algorithm

- Two resulting cells:

Cell #1

Cell #2

Exceptions

	P2	P3	P7	P6	P11	P5	P8	P10	P1	P4	P9
M1	1	1	1								
M7		1	1	1						1	1
M4		1		1					1		
M2					1	1			1		
M5						1	1				
M3					1			1			
M6							1	1	1	1	1
Cost	2.5	8.0	70.0	6.0	15.0	5.0	10.0	7.0	2.0	30.0	4.0

Analysis of the exceptions: Parts P4, P1 and P9 – as discussed previously

Next lecture

- Quiz

Next lecture

- Activity relationships
- Flow patterns
- Flow planning
- Measuring the flow
- Space requirements