MONASH University

FIT2090 BUSINESS INFORMATION SYSTEMS AND PROCESSES

Lecture 7 : Analysing Process Flows

CLAYTON, FACULTY OF INFORMATION TECHNOLOGY
MONASH UNIVERSITY

Reference: Chapter 5, Laguna & Marklund, 2nd Edition, CRC Press

GROUP OF EIGHT AUSTRALIA

Principles

- Businesses need to measure business process performances to provide feedback on their business improvement programs
- By analysing measures such as cycle times of processes, businesses can gain insights into their business improvement programs
- Businesses also need to manage their capacity so that their operations are lean with minimal waste

Objectives

On completion of this lecture, you will be able to:

- Describe the operational variables used to study processes in terms of stock and flow
- Describe the relationship between these operational variables using Little's Law
- Analyse cycle time and capacity

Why should we study/understand – analysing process flows

- An understanding of the operational variables of business processes is a fundamental skill/knowledge of business analysts
- By applying the above knowledge, business analysts are able to analyse the performance of business processes using measures such as cycle times for capacity planning and process design/planning

Contents

- Processes and Flows – Important Concepts
  - Throughput
  - WIP
  - Cycle Time
  - Little’s Formula
- Cycle Time Analysis
- Capacity Analysis
- Managing Cycle Time and Capacity
  - Cycle time reduction
  - Increasing Process Capacity
- Theory of Constraints

Stocks and Flows

- Stocks
  - items on shelves
  - employees
  - financial balance in an account

- Flows
  - rate of sales
  - hiring rate
  - outgoings per week

… in a business process

- Stocks
  - “work-in-progress”
  - “work-in-process” (number of jobs in ‘system’)

- Flows
  - “throughput” (jobs per time)

Business Processes and Flows

A process = A set of activities that transforms inputs to outputs

Two main methods for processing jobs

1. **Discrete processing** – each item is distinct
   - Examples: Cars, cell phones, tax files, etc.
2. **Continuous processing** – no individual items
   - Examples: Gasoline, electricity, consultancy duration etc.

"Job" = work unit

Three main types of flow structures

1. **Divergent** – Several outputs derived from one input
   - Example: Dairy and oil products
2. **Convergent** – Several inputs put together to one output
   - Example: Car manufacturing, general assembly lines
3. **Linear** – One input gives one output
   - Example: Hospital treatment

Example

In manufacturing, material flow names are given according to the shape of the dominant flow:

V-plant

- Process dominant by divergent flows

A-Plant

- A process dominant by converging flows

I-Plant

- A process dominant by linear flows

Flow rate is defined as the number of jobs per unit time

$R_i(t)$ = rate of incoming jobs through all entry points into the process
$R_o(t)$ = rate of outgoing jobs through all exit points from the process

Process Throughput

- Inflow and Outflow rates typically vary over time (see figure on next slide)
  - $R_i(t)$ = Arrival/Inflow rate of jobs at time t
  - $R_o(t)$ = Departure/Outflow rate of finished jobs at time t
  - IN = Average inflow rate over time
  - OUT = Average outflow rate over time

- A stable system must have IN = OUT = λ
  - λ = the process flow rate in
  - λ = the process flow rate out
  - λ = **process throughput**

Process Inflow and Outflow vary over time

Data for L&M, Figure 5.1 Pg 141

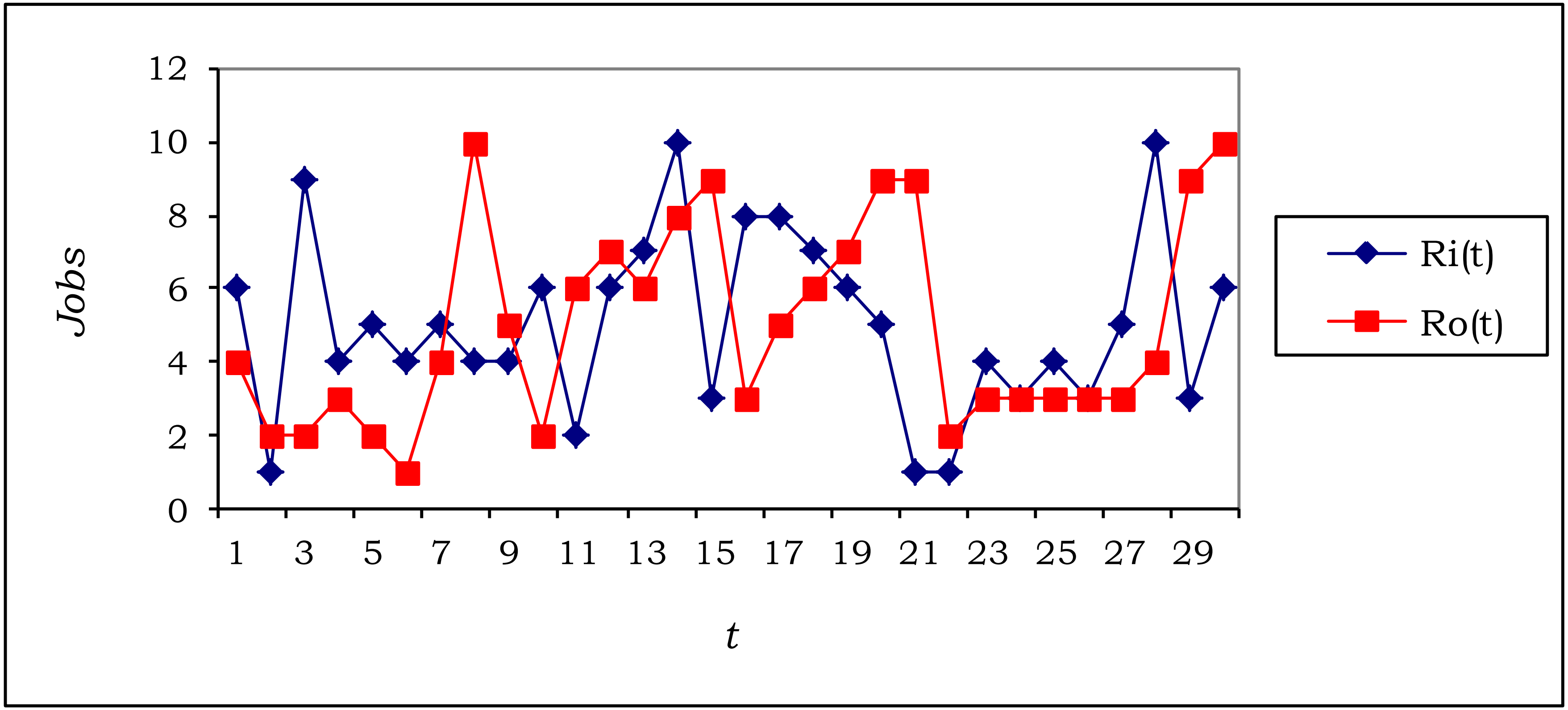
At t=8, outflow is 10 and inflow is 4

Average inflow

Average outflow

$R_i(t)$	$R_o(t)$	$\Sigma R_i(t)$	$\Sigma R_o(t)$	IN	OUT
6	4	6	4	6	4
1	2	7	6	3.5	3
9	2	16	8	5.3	2.7
4	3	20	11	5	2.8
5	2	25	13	5	2.6
4	1	29	14	4.8	2.3
5	4	34	18	4.9	2.6
4	10	38	28	4.8	3.5
4	5	42	33	4.7	3.7
6	2	48	35	4.8	3.5
2	6	50	41	4.6	3.7
6	7	56	48	4.7	4
7	6	63	54	4.8	4.2

Process Inflow and Outflow vary over time



IN = Average of inflow

= Sum of $R_i(t)$ / no. of time periods

OUT = Average of outflow

= Sum of $R_o(t)$ / no. of time periods

Over 30 periods, IN = OUT = approx. 5 jobs

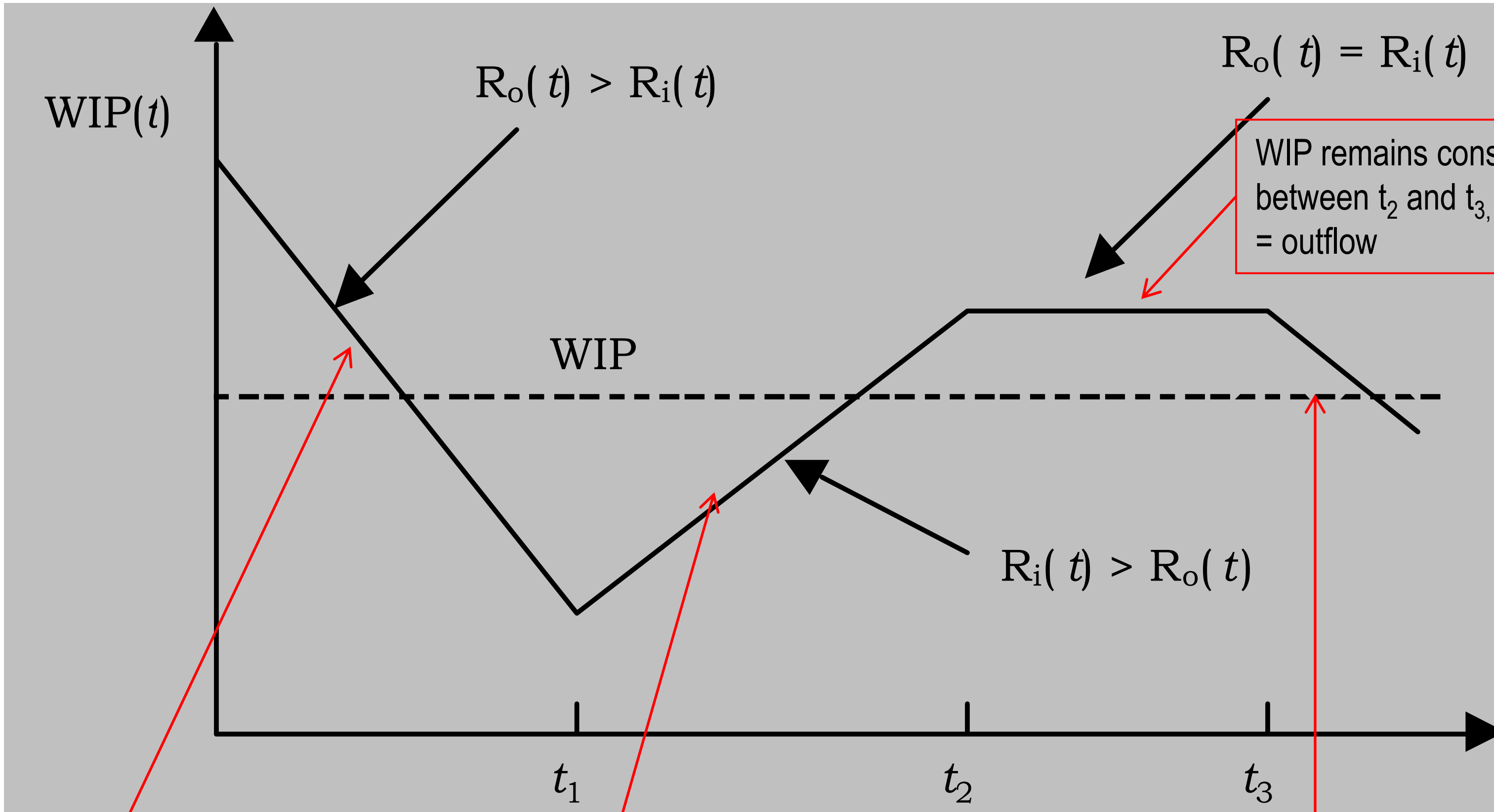
Work-In-Process ("WIP")

- WIP(t) comprises all jobs that have entered the process but not yet left it
  - including jobs waiting for the previous batch to be completed
- WIP(t) = Work in process at time t
  - WIP(t) increases when $R_i(t)>R_o(t)$
  - WIP(t) decreases when $R_i(t)<R_o(t)$
- WIP = Average work in process over time

The Seven Zeros of JIT

- Zero Defects: Quality at the source
- Zero Lot Size: To avoid batching delays
- Zero Setups: To minimize setup delay and allow production in small lots
- Zero Breakdowns: To avoid stopping tightly coupled line
- Zero Handling: To promote flow of parts
- Zero Lead Time: To ensure rapid replenishment of parts
- Zero Surging: *Necessary in system without WIP buffers*.

The WIP Level Varies With Process Inflow and Outflow



WIP remains constant between t_2 and t_3, thus inflow = outflow

WIP decreases at a rate of $R_o(t) – R_i(t)$ until time, t_1

From t_1 to t_2 inflow rate is larger than outflow rate so WIP increases

Average WIP = no. of jobs during each time period/sum of time periods

Process Cycle Time

- The difference between a job's departure time and its arrival time = **cycle time**
  – One of the most important attributes of a process
  – Also referred to as **throughput time**
- The cycle time includes both value adding and non-value adding activity times
  – Processing time
  – Inspection time
  – Transportation time
  – Storage time
  – Waiting time
- Cycle time is a powerful tool for identifying process improvement potential

Little's Formula (Due to J.D.C. Little (1961))

- States a fundamental and very general relationship between the average: WIP, Throughput (= λ) and Cycle time (CT)
  - The cycle time refers to the time the job spends in the system or process

Little's Formula: WIP = $\lambda \cdot$CT

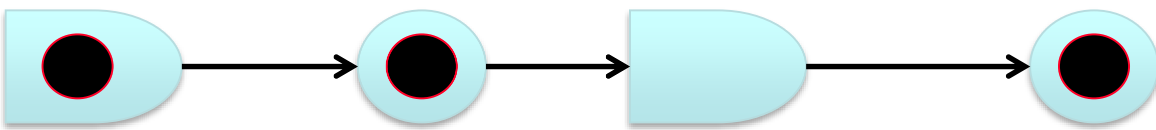
- Implications, everything else equal
  - Shorter cycle time $\Leftrightarrow$ lower WIP
  - If λ increases $\Rightarrow$ to keep WIP at current levels CT must be reduced
- A related measure is (inventory) turnover ratio
  - Indicates how often the WIP is entirely replaced by a new set of jobs

Turnover ratio = 1/CT

Cycle time, Throughput and WIP

Throughput: a job arrives every 6 minutes

= 1/6 jobs per minute



WIP = 3

3 6 3 6 Av: processing time = 18

= "cycle time"

WIP (jobs)
= Cycle time X Throughput (minutes X jobs per minute)
= 18 / 6 (jobs)
= 3 jobs

Throughput (jobs per minute)
= WIP / Cycle Time (jobs / minutes)
= 3/18 (jobs per minute)
= 1/6 of a job per minute

Exercise

Insurance company processes an average of 12,000 claims/yr. On average at any one time, there are 600 applications at various stages of processing. If there are 50 working weeks/yr, how many weeks (on average) does processing a claim take?

λ = 12,000 claims/year

WIP = 600 jobs
WIP = $\lambda \cdot CT$

CT = WIP/ λ = 600/12,000
= 1/20 years
= (1/20)*50 = 2.5 working weeks

CT = WIP/ λ
= (**300**/12,000)*50
= 1.25 working weeks

How can cycle time be reduced?

Either reduce WIP or increase the throughput rate.

Redesign process by reducing the average WIP to **300**, thus CT is reduced to 1.25 weeks

Overview

- Processes and Flows – Important Concepts
  - Throughput
  - WIP
  - Cycle Time
  - Little’s Formula
- **Cycle Time Analysis**
- Capacity Analysis
- Managing Cycle Time and Capacity
  - Cycle time reduction
  - Increasing Process Capacity

Cycle Time Analysis

- The task of calculating the ***average* cycle time** for an entire process or process segment
  - Assumes that the average activity times for all involved activities are available
- In the simplest case a process consists of a sequence of activities on a single path
  - The average cycle time is just the ***sum of the average activity times*** involved
- … but in general we must be able to account for
  - Rework
  - Multiple paths
  - Parallel activities

Rework

- Many processes include control or inspection points where if the job does not conform it will be sent back for rework
  - The rework will directly affect the average cycle time!
- Definitions
  - T = sum of activity times in the rework loop
  - r = percentage of jobs requiring rework (rejection rate)
- Assuming a job is never reworked more than once

$$\Rightarrow \boxed{CT = (1+r)T}$$

- Assuming a reworked job is no different than a regular job

$$\Rightarrow \boxed{CT = T/(1-r)}$$

Some Beautiful Mathematics

- Not for examination…

Repeated reworking:

Cycle Time (CT) =

$T + r*T + r*(r*T) + r*r*r*T + \ldots + r^n * T + \ldots$

Therefore:

$r*CT = r*T + r*(r*T) + r*r*r*T + \ldots + r^n * T + \ldots$

$= CT – T$

Therefore:

$T = CT(1 – r)$

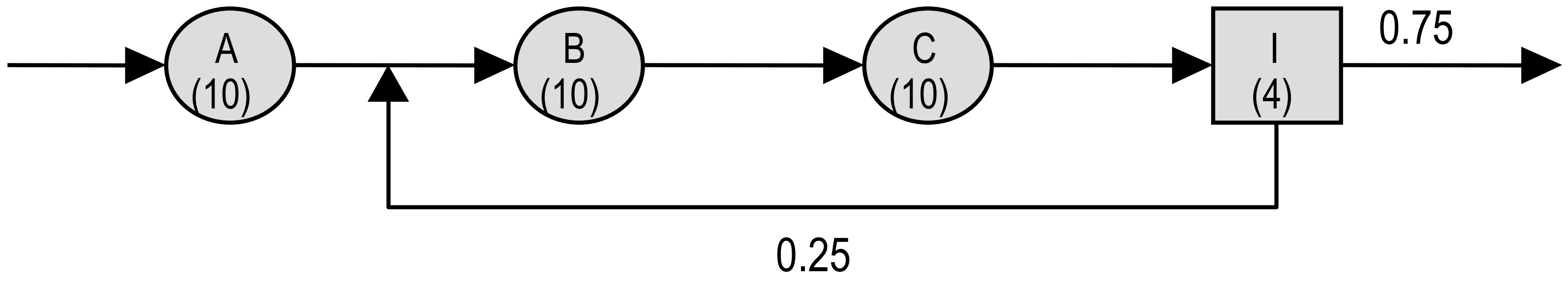
$CT = T/(1-r)$

MONASH University

Example – Rework effects on the average cycle time

Consider a process consisting of

- Three activities, A, B & C taking on average 10 min. each
- One inspection activity (I) taking 4 minutes to complete.
- X% of the jobs are rejected at inspection and sent for rework



What is the average cycle time?

a) If no jobs are rejected and sent for rework.

b) If 25% of the jobs need rework but never more than once.

c) If 25% of the jobs need rework but reworked jobs are no different in quality than ordinary jobs.

a) If no jobs are rejected and sent for rework.

$$10+ (10+10+4) = 34 \text{ mins}$$

A B C I

b) If 25% of the jobs need rework but never more than once.

$$CT = (1+r)T$$

$$10 + (1+0.25)*(10+10+4) = 40 \text{ mins}$$

A B C I

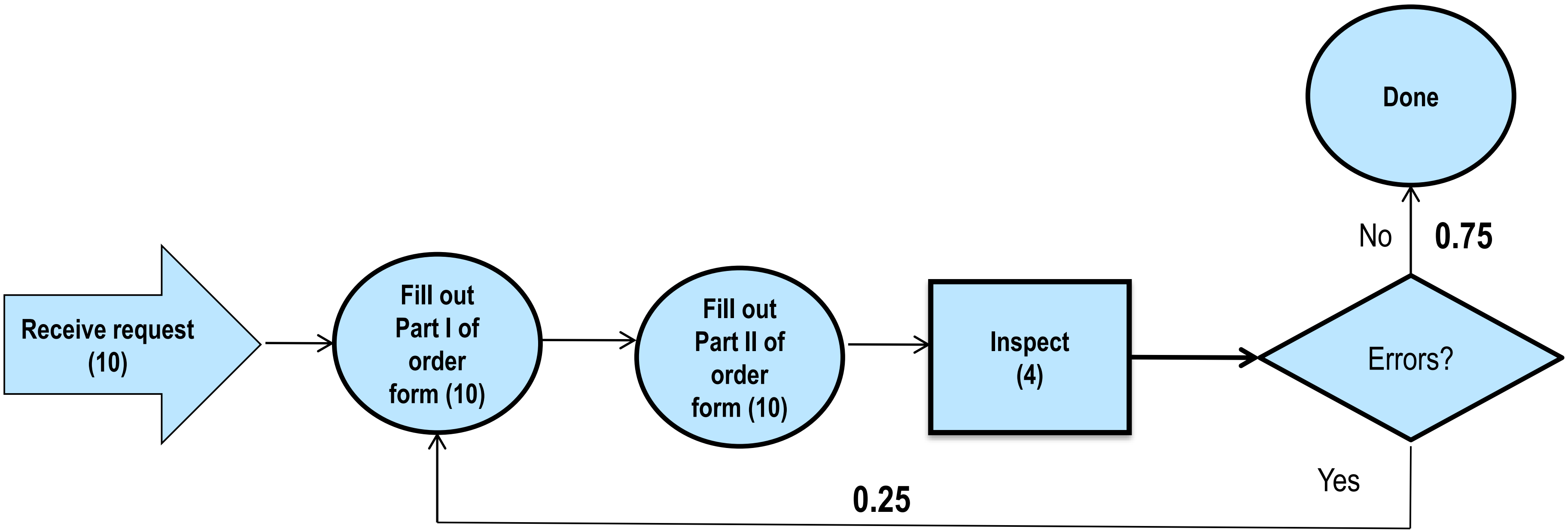
a) If 25% of the jobs need rework but reworked jobs are no different in quality than ordinary jobs.

$$CT = T/(1-r)$$

$$CT = 10 + (10+10+4)/(1-0.25) = 42 \text{ mins}$$

A B C I

Example



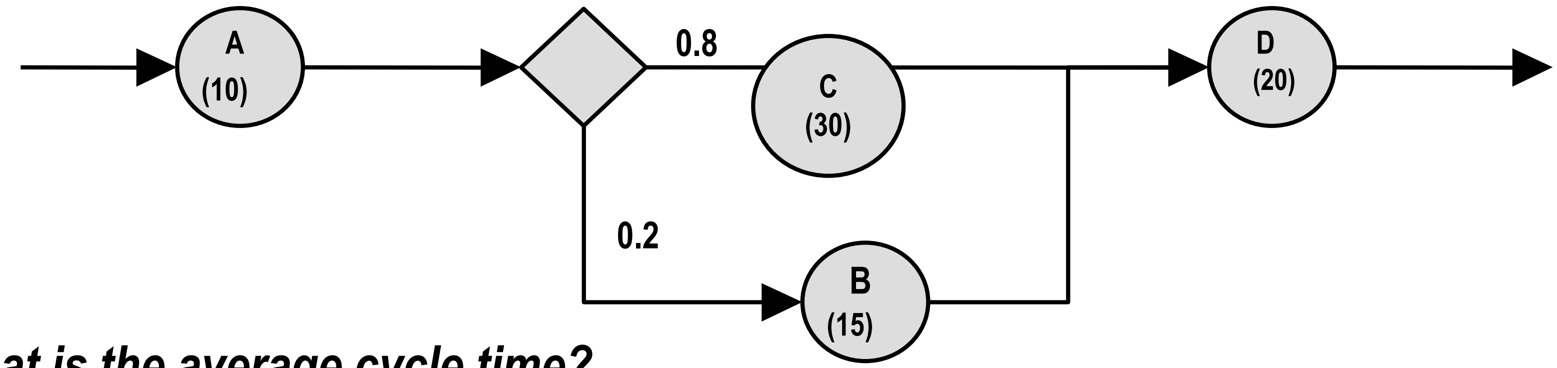
What is the average cycle time for this process?

CT = T/(1-r)

Cycle Time (CT) = 10 + (10+10+4)/0.75 = 42 minutes

Example – Processes with Multiple Paths

- Consider a process segment consisting of 4 activities A, B, C and D with activity times 10,15, 30 & 20 minutes respectively
- On average 20% of the jobs are routed via B and 80% go straight to activity C.



➢ ***What is the average cycle time?***

For 100 jobs:

80 take 10+30+20 = 60 minutes ➔ 4800 total

20 take 10+15+20 = 45 minutes ➔ 900 total

Average = 4800+900 / 100 = 57 minutes

Multiple Paths

- It is common that there are alternative routes through the process
  - For example: jobs can be split in "fast track" and normal jobs

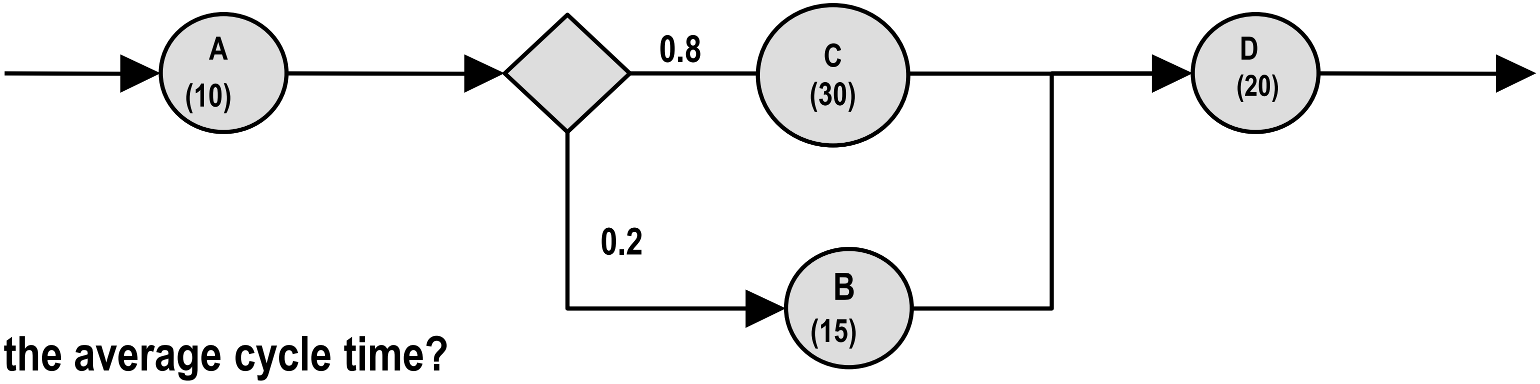
- Assume that **m** different paths originate from a decision point
  - p_i = The probability that a job is routed to path i
  - T_i = The time to go down path i

$$CT = p_1T_1+p_2T_2+\ldots+p_mT_m = \sum_{i=1}^{m} p_iT_i$$

Example – Processes with Multiple Paths

Consider a process segment consisting of 4 activities A, B, C and D with activity times 10,15, 30 & 20 minutes respectively

On average 20% of the jobs are routed via B and 80% go straight to activity C.



What is the average cycle time?

10 + (average time in C and B) + 20

10 + (80*30 + 20*15) / 100 + 20 **(as an average)**

10 + ((0.8 * 30) + (0.2 * 15)) + 20 **(as a probability)**

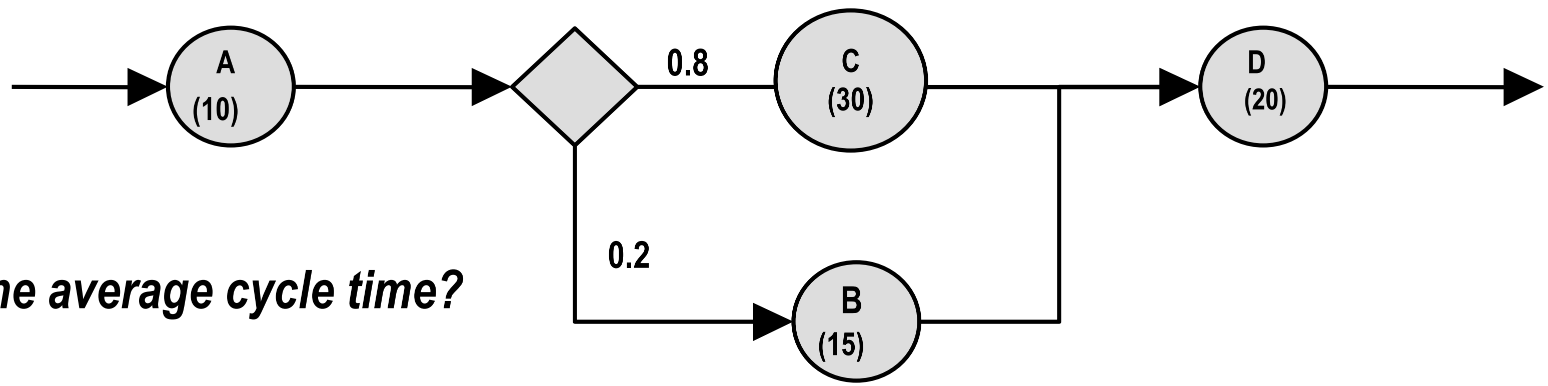
= 10 + 27 + 20

= 57 minutes

Example – Processes with Multiple Paths

Consider a process segment consisting of 4 activities A, B, C and D with activity times 10,15, 30 & 20 minutes respectively

On average 20% of the jobs are routed via B and 80% go straight to activity C.



What is the average cycle time?

$$CT = p_1T_1+p_2T_2+\ldots+p_mT_m= \sum_{i=1}^{m} p_i T_i$$

CT = 10 + (0.8*30) + (0.2*15) + 20 = 57 minutes

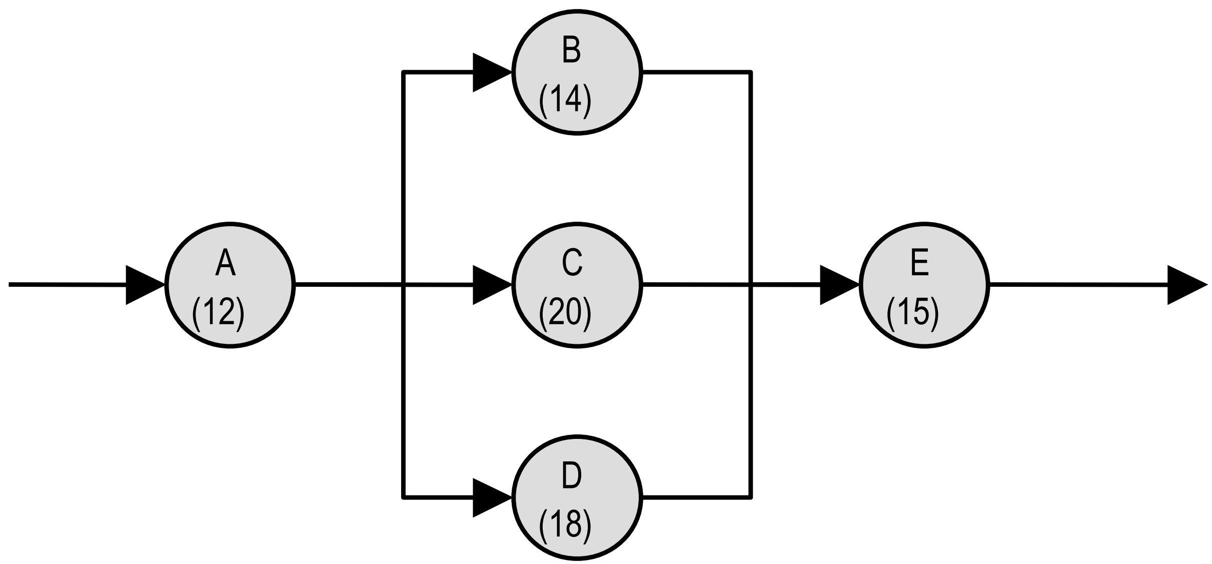
Processes with Parallel Activities

- If two activities related to the same job are done in parallel the contribution to the cycle time for the job is the maximum of the two activity times.
- Assuming
  - M process segments in parallel
  - T_i = Average process time for process segment i to be completed

$$CT_{parallel} = \mathrm{Max}\{T_1, T_2, \ldots, T_M\}$$

Example – Cycle Time Analysis of Parallel Activities

Consider a process segment with 5 activities A, B, C, D & E with average activity times: 12, 14, 20, 18 & 15 minutes



What is the average cycle time for the process segment?

12 + Max{14, 20, 18} + 15

A *B* *C* *D* *E*

= 12 + 20 +15 = 47 minutes

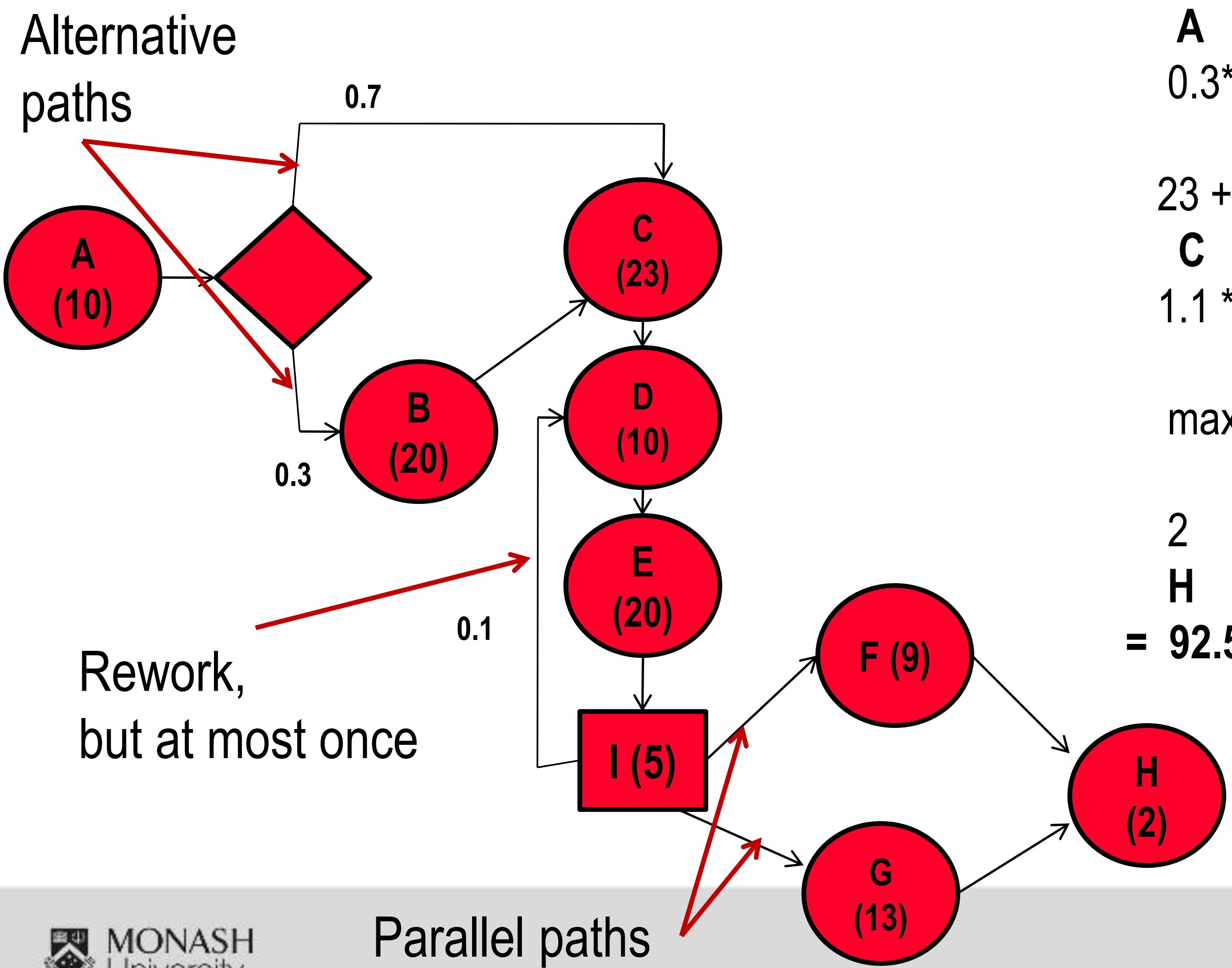
Cycle Time Efficiency

- Measured as the percentage of the total cycle time spent on value adding activities.

$$\text{Cycle Time Efficiency} = \frac{\text{Theoretical Cycle Time}}{\text{CT}}$$

- Theoretical Cycle Time = the cycle time which we would have if only value adding activities were performed
  – That is if the activity times, which include waiting times, are replaced by the processing times

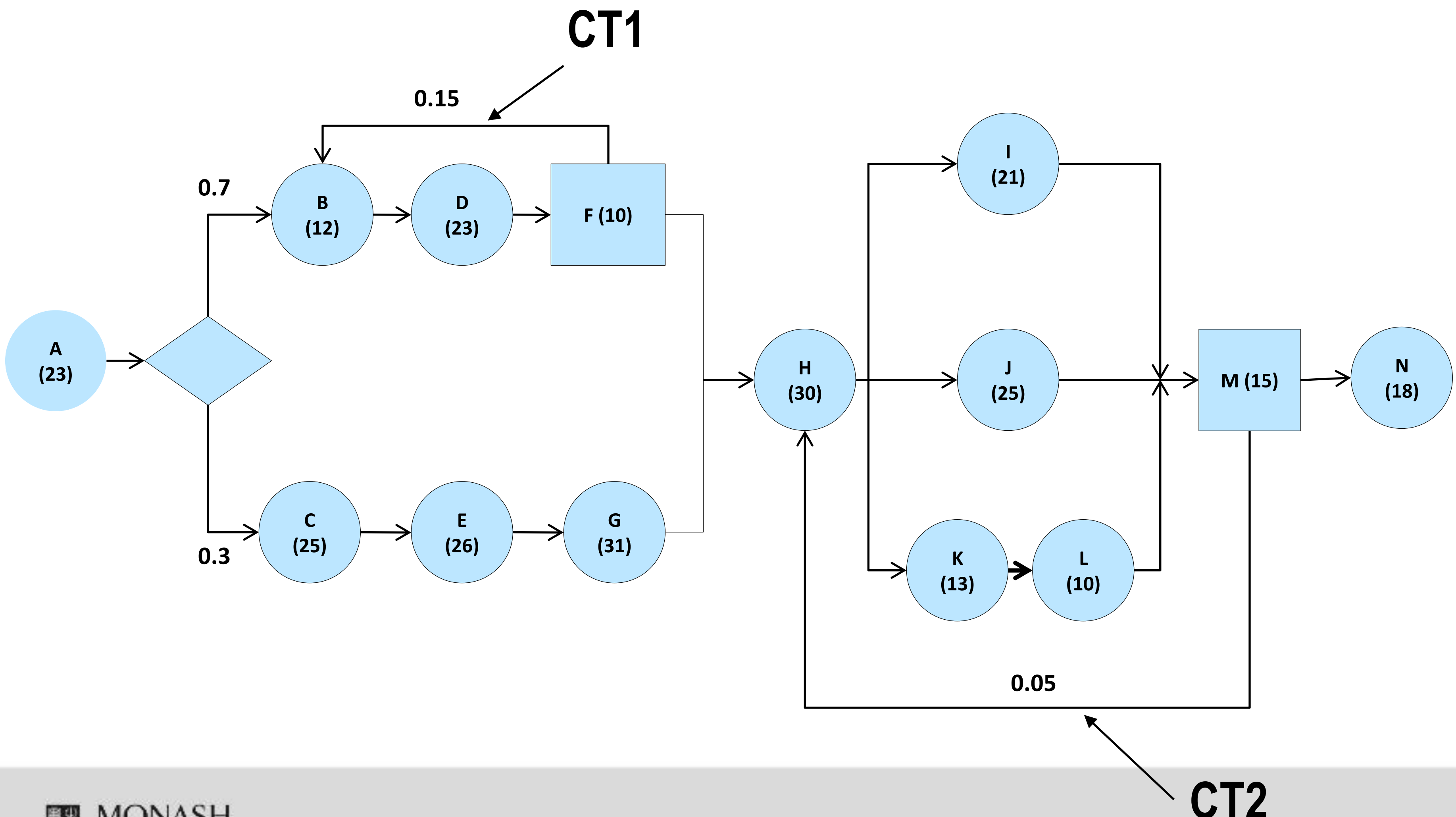
Flowchart Cycle Time Example
L&M Fig. 5.6 p.158



CT = 10 +
A
0.3*20 + 0.7*0 +
B
23 +
C
1.1 * (10+20+5) +
D E I
max{ 9,13 } +
F G
2
H
= 92.5 minutes

Cycle time analysis Exercise 5.9, Pg 176

Flow-chart
Pg 177 Figure 5.12



Exercise

Assuming a job is never reworked more than once in the same rework loop.

Define

T_i =Activity time for activity i (i=A,B,C,D,E,F,G,H,I,J,K,L,M,N)

CT =Average cycle time for the process in question

Let CT_1 represent the average cycle time for the rework loop consisting of activities B, D and F.

Let CT_2 represent the average cycle time of the rework loop consisting of activities H, I, J, K, L, and M.

$CT_1 = 1.15 * (T_B+T_D+T_F) = 1.15 * (12+23+10) = 1.15 * 45$

$= 51.75$ minutes

$CT_2 = 1.05 * (T_H + \max\{T_I, T_J, T_K+T_L\} + T_M)$

$= 1.05 * (30 + \max\{21, 25,13+10\} + 15)$

$= 1.05 * 70 = 73.5$ minutes

$CT = T_A + (0.7 * CT_1) + 0.3 * (T_C+T_E+T_G) + CT_2 + T_N$

$= 23 + (0.7 * 51.75) + 0.3 * (25+26+31) + 73.5 + 18$

= **175.325** minutes

Overview

- Processes and Flows – Important Concepts
  - Throughput
  - WIP
  - Cycle Time
  - Little’s Formula
- Cycle Time Analysis
- **Capacity Analysis**
- Managing Cycle Time and Capacity
  - Cycle time reduction
  - Increasing Process Capacity

Capacity Analysis

- Focus on assessing the capacity needs and resource utilization in the process
  1. Determine the **number of jobs** flowing through different process segments
  2. Determine **capacity requirements** and **utilization** based on the flows obtained in 1.
- The capacity requirements are directly affected by the process configuration
  - ⇒ Flowcharts are valuable tools
  - ⇒ Special features to watch out for
    - Rework
    - Multiple Paths
    - Parallel Activities
- Complements the cycle time analysis!

The Effect of Rework on Process Flows

- A rework loop implies an increase of the flow rate for that process segment
- Definitions
  - N = Number of jobs flowing through the rework loop
  - n = Number of jobs arriving to the rework loop from other parts of the process
  - r = Probability that a job needs rework
- Assuming a job is never reworked more than once

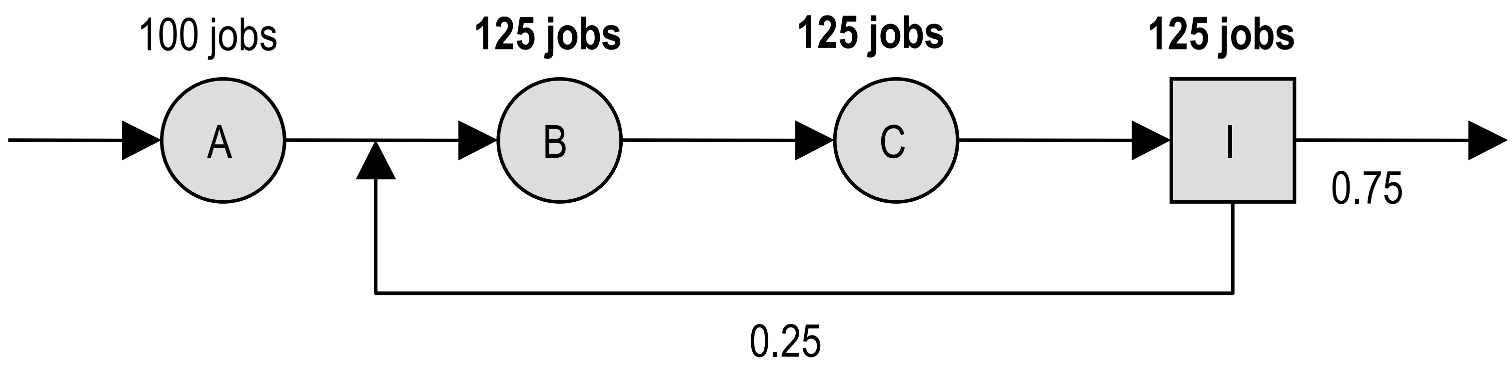
$$\Rightarrow \quad \boxed{N = (1+r)n}$$

- Assuming a reworked job is no different than a regular job

$$\Rightarrow \quad \boxed{N = n/(1-r)}$$

Example – Capacity Analysis with Rework

$N = (1+r)n = (1+0.25)100 = 125$



Multiple Paths and Parallel Activities

Multiple Paths and process flows

- The flow along a certain path depends on
  - The number of jobs entering the process as a whole (n)
  - The probability for a job to go along a certain path
- Defining
  - N_i = number of jobs taking path i
  - p_i = Probability that a job goes along path i

$$\Rightarrow \boxed{N_i = n \cdot p_i}$$

Parallel Activities and process flows

- All jobs still have to go through all activities
  - if they are in parallel or sequential does not affect the number of jobs flowing through a particular activity

Analyzing Capacity Needs and Utilization (I)

- Need to know
  - Processing times for all activities
  - The type of resource required to perform the activity
  - The number of jobs flowing through each activity
  - The number of available resources of each type

Step 1 – Calculate unit load for each resource

- The total resource time required to process one job
  - N_i = Number of jobs flowing through activity i for every new job entering the process
  - T_i = The processing time for activity i in the current resource
  - M = Total number of activities using the resource

$$\text{Unit load for Resource} = \sum_{i=1}^{M} N_i \cdot T_i$$

No. of Jobs

Unit Load for R1 = 2*1 + 5*0.3 + 2*1 = 5.5 min

Process Time

Activity	Processing Time (T)	Resource	No. of Jobs (N)
1	2 min	R1	1
2	5 min	R1	0.3
3	2 min	R1	1
4	3 min	R2	1.1
5	4 min	R2	1.1

Analyzing Capacity Needs and Utilization (II)

Step 2 – Calculate the unit capacity

- The number of jobs per time unit that can be processed

Unit capacity for resource j = 1 / Unit load for resource j

Step 3 – Determine the resource pool capacity

- A resource pool is a set of identical resources available for use
- Pool capacity is the number of jobs per time unit that can be processed
  - Let M = Number of resources in the pool

Pool capacity = M*Unit capacity = M / Unit load

Let’s say available resource for R1 is 2

In other words we have two people (or two machines) that can do tasks assigned to R1

Unit Load	**5.5 minutes**	
Unit Capacity	**1/ 5.5 jobs per min**	
Pool Capacity	**2 x 1/5.5 jobs per min**	**0.36 jobs per min**

Analyzing Capacity Needs and Utilization (III)

Capacity is related to resources not to activities!

- The process capacity is determined by the **bottleneck**
  - The bottleneck is the resource or resource pool with the **smallest capacity** (the **slowest** resource in terms of jobs/time unit)
  - The slowest resource will limit the process throughput

Capacity Utilization

- The theoretical process capacity is obtained by focusing on processing times as opposed to activity times
  - Delays and waiting times are disregarded

  $\Rightarrow$ ***The actual process throughput $\leq$ The theoretical capacity***!

$$\text{Capacity Utilization} = \frac{\text{Actual Throughput}}{\text{Theoretical Process Capacity}}$$

Example

Resource type	Pool Capacity (jobs/min) x 60 =	Pool Capacity (jobs/hour)
R1	0.36	21.6
R2	0.13	7.8
R3	0.17	10.2

So, R2 is the bottleneck and the Process Capacity is 7.8 jobs/hour
Let’s say in reality, the actual throughput is only 6 jobs/hour
Then Capacity Utilisation is (6/7.8) x 100% = 76.9%

Overview

- Processes and Flows – Important Concepts
  - Throughput
  - WIP
  - Cycle Time
  - Little's Formula
- Cycle Time Analysis
- Capacity Analysis
- **Managing Cycle Time and Capacity**
  - Cycle time reduction
  - Increasing Process Capacity

Cycle time Reduction

Cycle time and capacity analysis provide valuable information about process performance

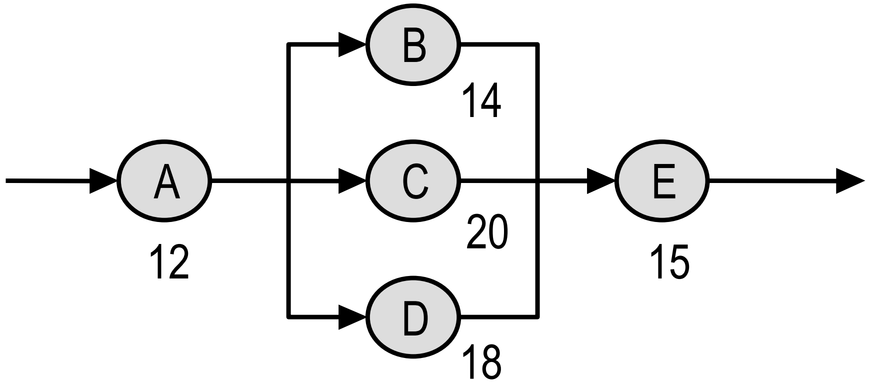
- Helps identify problems
- Increases process understanding
- Useful for assessing the effect of design changes

Ways of reducing cycle times through process redesign

1. Eliminate activities
2. Reduce waiting and processing time
3. Eliminate rework
4. Perform activities in parallel
5. Move processing time to activities not on the critical path
6. Reduce setup times and enable batch size reduction

Example – Critical Activity Reduction

Consider a process with three sequences or paths



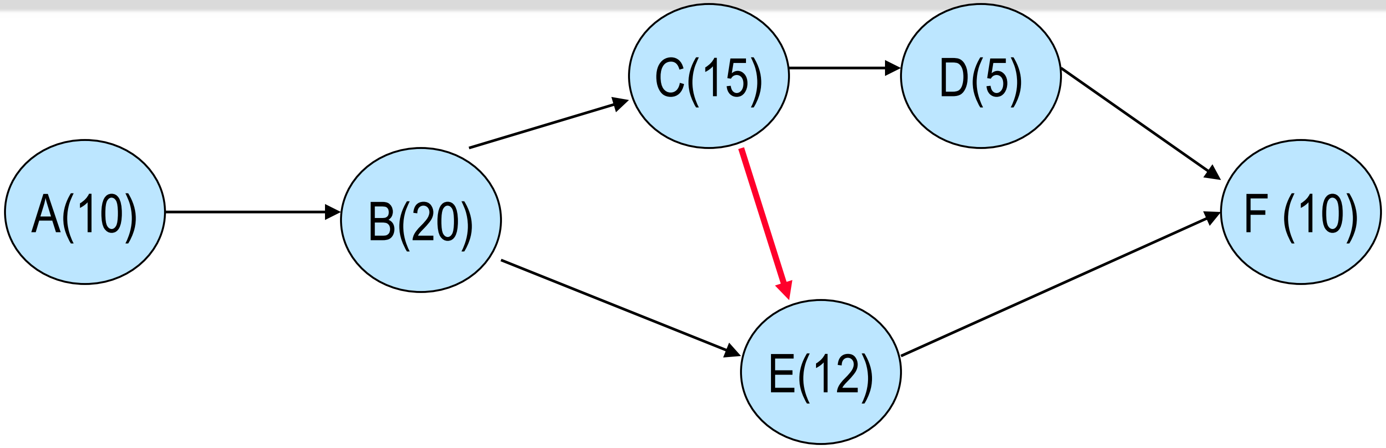
Sequence (Path)	Time required (minutes)
1. A→B →E	12+14+15 = 41
2. A→C →E	12+20+15 = 47 = ***CT***
3. A →D →E	12+18+15 = 45

Critical path

⇒ By moving 2 minutes of activity time from path 2 to path 1 the cycle time is reduced by 2 minutes to CT=45 minutes

⇒ i.e. when we move some work content from the critical path to a non critical path we can decrease the cycle time

Example



Path	Length
A->B->C->D->F	10+20+15+5+10 = 60 min
A->B->E->F	10+20+12+10=52 min

To reduce cycle time, we can redesign the process by moving some work from the critical path to a non critical activity, such as E in this case.

Suppose we move 4 min from C to E, this reduces the CT for the critical path to 56 min and the other path also becomes a critical path, i.e. its CT is also 56 min

Increasing Process Capacity

- Two fundamental ways of increasing process capacity
  1. **Add resource capacity at the bottleneck**
     - Additional equipment, labor or overtime
     - Automation
  2. **Reduce bottleneck workload**
     - Process redesign
       - Shifting activities from the bottleneck to other resources
       - Reducing activity time for bottleneck jobs
- When the goal is to reduce cycle time and increase capacity careful attention must be given to
  - The resource availability
  - The assignment of activities to resources

Theory of Constraints (TOC) (I)

- An approach for identifying and managing bottlenecks
  – To increase process flow and thereby process efficiency
- TOC is focusing on improving the bottom line through
  – Increasing throughput
  – Reducing inventory
  – Reducing operating costs

  $\Rightarrow$ ***Need operating policies that move the variables in the right directions without violating the given constraints***
- Three broad constraint categories
  1. Resource constraints
  2. Market constraints
  3. Policy constraints

Theory of Constraints (TOC) (II)

TOC Methodology

1. **Identify the system's constraints**
2. **Determine how to exploit the constraints**
   – Choose decision/ranking rules for processing jobs in bottleneck
3. **Subordinate everything to the decisions in step 2**
4. **Elevate the constraints to improve performance**
   – For example, increasing bottleneck capacity through investments in new equipment or labor
5. **If the current constraints are eliminated return to step 1**
   – Don't loose inertia, continuous improvement is necessary!

- *See example 5.18 , Chapter 5 in Laguna & Marklund*

Example - Applying the TOC Methodology

Consider a process with 9 activities and 3 resource types (X,Y,Z). Activities 1, 2 & 3 require 10 minutes of processing and the other activities 5 minutes each.

There are 3 jobs, following different paths being processed

Job	Routing	Demand (Units/week)	Profit Margin
A	4, 8, and 9	50	$20
B	1, 2, 3, 5, 6, 7, and 8	100	$75
C	2, 3, 4, 5, 6, 7, 8, and 9	60	$60

- Activities 1, 2 & 3 utilize resource **X**, activities 4, 5, & 6 resource **Y** and activities 7, 8 & 9 resource **Z**. Each resource have **2400** minutes of weekly processing time available

MONASH University

Step 1. Identify system constraints

Resource Utilisation Calculations

Resource	Requirements (min/week)	Utilisation
X	(30X100) + (20X60)=4,200 Job B Job C	4,200/2,400=175%
Y	(5X50)+(10X100)+(15X60) = 2,150 Job A Job B Job C	2,150/2,400 = 90%
Z	(10X50)+(10X100)+(15X60)=2,400 Job A Job B Job C	2,400/2,400=100%

- Resource X is the bottleneck in this problem
- Resource X required over 100% utilisation, so the process is constrained by Resource X

Step 2: Determine how to exploit the system's constraint

Consider 3 rules to process jobs and calculate total weekly profit for each rule.

2.1 Rank jobs based on profit margins

-> **B,C, A**

2.2 Rank jobs based on their profits per direct labour hour:

e.g. Job A has \$1.33 (\$20/15) per direct labour hour, Job B has \$1.50 and C has \$1.20

-> **B,A,C**

2.3 Rank jobs based on their contribution per minute of the constraint, i.e. ratio of profit and direct labour in Resource X (the bottleneck)

Job B has \$2.50 per direct labour per minute in Resource X

Profit/Labour in Resource = \$75/30 = \$2.50

Job C has \$3.00

Job A – its contribution is irrelevant because A jobs are not routed to Resource X

-> **C,B, A** (where A is a 'free' job w.r.t. Resource X

How can Resource X be utilised more effectively?

Step 3. Subordinate everything to the decisions in step 2

Calculate the no. of jobs of each type to be processed, utilisation of each resource, total weekly profit – depends on ranking rule in step 2

3.1 Max no. of B jobs = 80 per week, i.e. maximum that X can complete (2400/30=80)

- If 80 B jobs are processed then no Job C is processed because no more capacity left in X
- Job A can be processed because it does not use Resource X.
- Resource Y is not a constraint because its max. utilisation is 90%
- After Job B, 1,600 minutes left (2400-800) are left in Resource Z
- -> 160 (i.e.1600/10 = 160) Job A can be processed
- Demand for Job A is 50 per week so this can be satisfied
- See resource utilisation table below:

Resource	Requirements (min/week)	Utilisation
X	80 jobs X 30 mins/each job = 2400 mins	2400/2400=100%
Y	(10 X 80)+(5 X 50)=1,050	1,050/2400=44%
Z	(10X80)+(10X50)=1,300	1,300/2400=54%

Total profit of this processing plan (i.e. B,C,A) is \$75X80 + \$20X50 = **\$7,000**

Step 3. Subordinate everything to the decisions in step 2

3.2 From rule 2.2, ranking is B,A,C,

Job A does not require Resource X so processing plan is the same as for rule 2.1 (see previous slide),

Total profit is also **$7,000**

3.3 (**C,B,A**) Calculate max number of Job C that can be processed through bottleneck

120 Job C can be processed each week (2,400/20)

-> Entire demand for Job C (120) can be met.

Subtract capacity from bottleneck and calculate max number of Job B that can be processed with remaining capacity

->40 Job B (1,200/30)

Job A requires Resource Z. Updated capacity of Resource Z is 1,300 min (after subtracting times for Jobs B (40 X 10) and C (60 X 15))

-> Entire demand for Job A can be met

Utilisation for ranking rule 2.3

Resource	Requirements (min/week)	Utilisation
X	(30x40) + (20x60)=2,400	2400/2400 = 100%
Y	(5x50)+(10x40)+(15x60)=1,550	1550/2400=64%
Z	(10x50)+(10x40)+(15x60)=1,800	1,800/2400=75%

Total profit of plan 2.3 is $20X50 + $75X40+$60X60=**$7,600**

Rule 2.3 gives better results in constrained processes as shown in this example where the goal is selecting the **mix of products or services that maximizes total profit.**

Maximising Profit in a Restaurant

- Restaurant has 28 tables
- Historically there are usually tables reserved for which customers fail to arrive

# No Shows	Probability
0	0.1
1	0.25
2	0.3
3	0.2
4	0.15

Maximising Profit in a Restaurant

Restaurant decides to "overbook"

Profit per table: $60
Loss of goodwill if booked table unavailable: $30

How many overbookings per night yields the most profit?

Maximising Profit in a Restaurant

Expected profit with one overbooking (29 reservations accepted)

#Shows	29	28	27	26	25
Prob.	0.1	0.25	0.3	0.2	0.15
Covers	28	28	27	26	25
Profit	1,680	1,680	1,620	1,560	1,500
Turned away	1	0	0	0	0
Cost	30	0	0	0	0
Net Profit	1,650	1,680	1,620	1,560	1,500
Expected Profit (Net Profit X Prob.)	165	420	486	312	225
Overall Expected Profit			**$1,608**		

Maximising Profit in a Restaurant

Which number of overbookings is best?

Number of overbookings	Overall expected Profit
0	1,557
1	1,608
2	1,636.5
3	1,638
4	1.621.5

3 overbookings is best

Expected profit is just $1.50 more
for three overbookings than two overbookings

Example

	Product A	Product B
Profit	$80	$50
Demand	100 units	200 units
Resource	0.4 hours/unit	0.2 hours/unit
Resource available	60 hours	

How should the resources be allocated to maximize profit?

Contd.

	Product A	Product B
Profit	$80	$50
Demand	100 units	200 units
Resource	0.4 hours/unit	0.2 hours/unit
Resource available	60 hours	
Profit	$80/0.4 = $200 per hour	$50/0.2 = $250 per hour

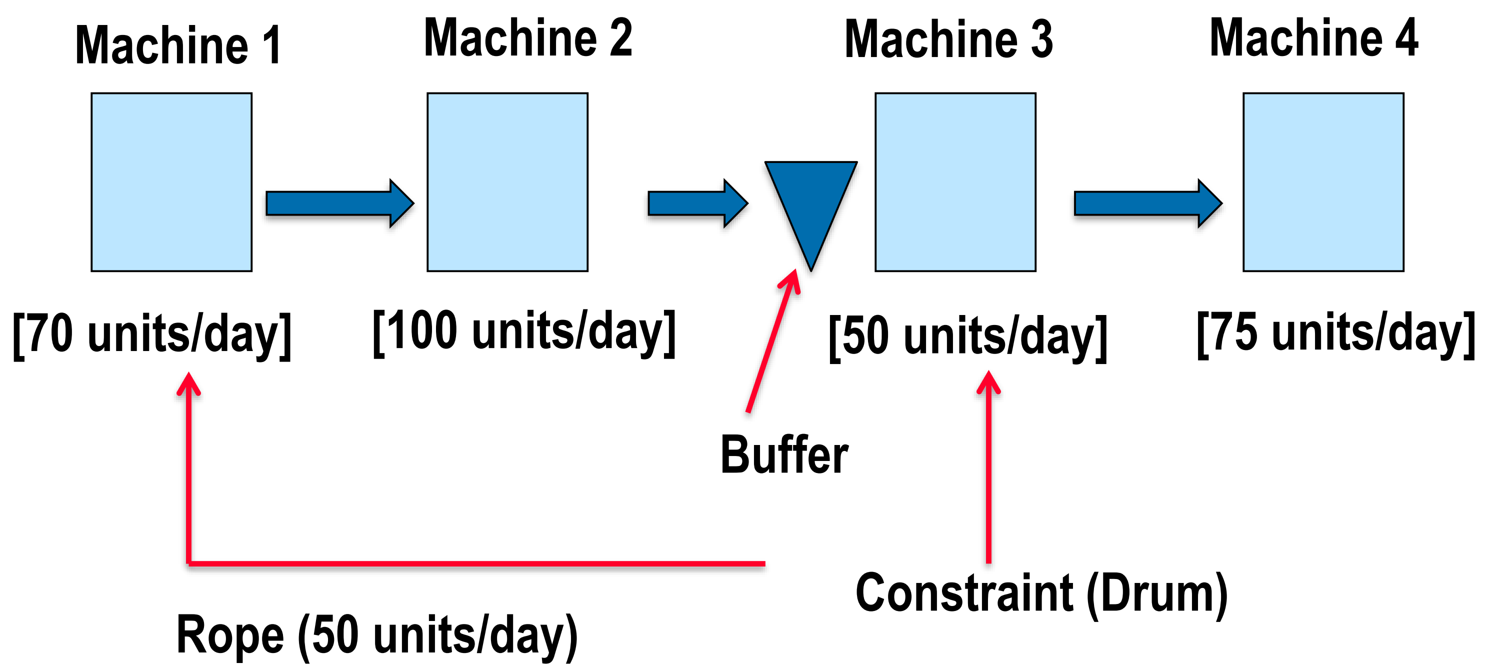
Profit ➔ only produce product B

Constraint ➔ the demand for product B is only 200

Solution:

- Use 200*0.2 = 40 hours meeting all the demand for product B
- Use 20 hours to produce 20/0.4 = 50 units of product A
- **Profit** = 50*200 + 80*50 = 10000 + 4000 **= $14,000**

Drum, Buffer, Rope (DBR) Concept



Theory of Constraints applied to Supply Chains

- End customer demand drives the pace of a supply chain
- One supplier in the chain may be more constrained than the other suppliers
- Aim of supply chain is to maximise its capability to meet demand by maintaining a buffer of the key component to protect against stockouts downstream of this key component manufacturer
- Set inventory levels at other suppliers of the chain according to the demand and buffer levels to minimise total supply chain inventory costs

Summary

- Three operational variables in a process i.e. **throughput, work-in-process** and **process cycle**
- The relationship between these operational variables using **Little's Law**
- Analysis of process performance: **process cycle time, capacity**
- **Theory of Constraints**