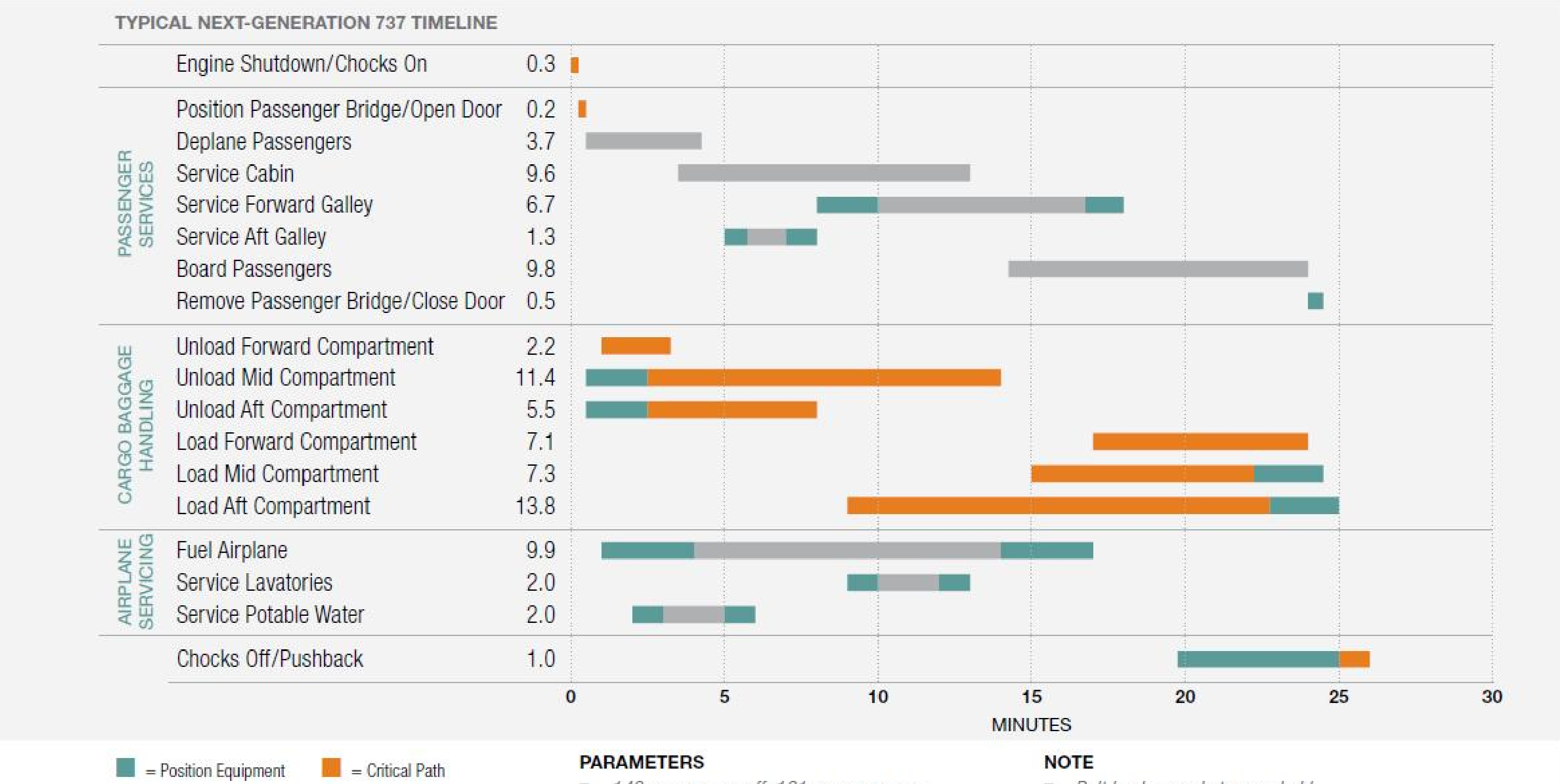
Chapter 12. Project Management

Gantt Chart

A **Gantt chart** is a type of bar chart that illustrates a project schedule.

- Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.
- Terminal elements and summary elements comprise the work breakdown structure of the project.
- Modern Gantt charts also show the dependency relationships between activities.

Gantt Chart : Example



PARAMETERS

- *142 passengers off, 121 passengers on*
- *2 doors used to deplane and enplane*
- *1 galley service truck*
- *1 lavatory service truck*
- *1 potable water service truck*

NOTE

- *Belt loader used at cargo hold*
- *Aft galley, potable water, and lavatory service complete before passenger boarding*

Gantt Chart : Example

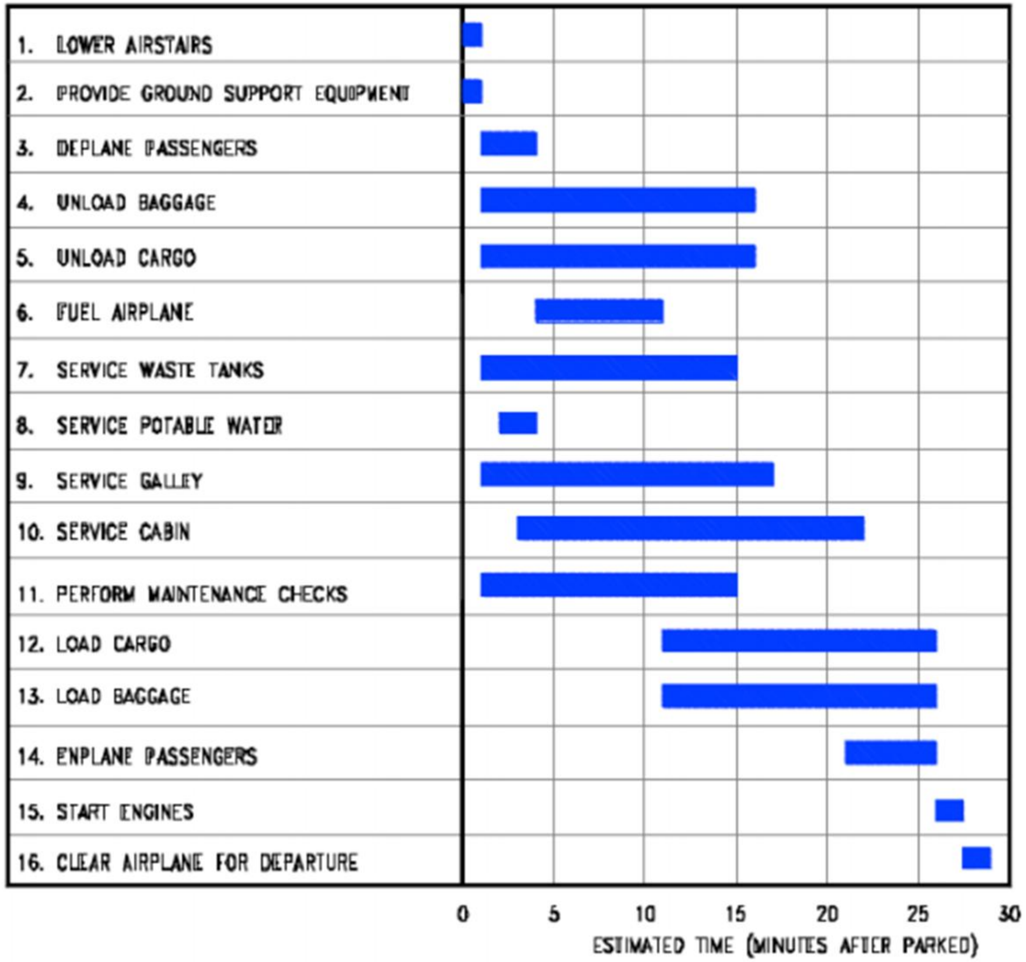


Figure 2.5 Typical turnaround Gantt chart 737-900, -900ER **(Boeing 2005)**

Network Modeling : Example

Example : Marriage

A. Propose to a girlfriend.
B. Approval from my parents.
C. Approval from her parents.
D. Select a place to live after wedding.
E. Choose the wedding date.
F. Prepare wedding gifts.
G. Select a place to perform the wedding ceremony.
H. Select a honeymoon travel place.
I. Send invitations.
J. Wedding.

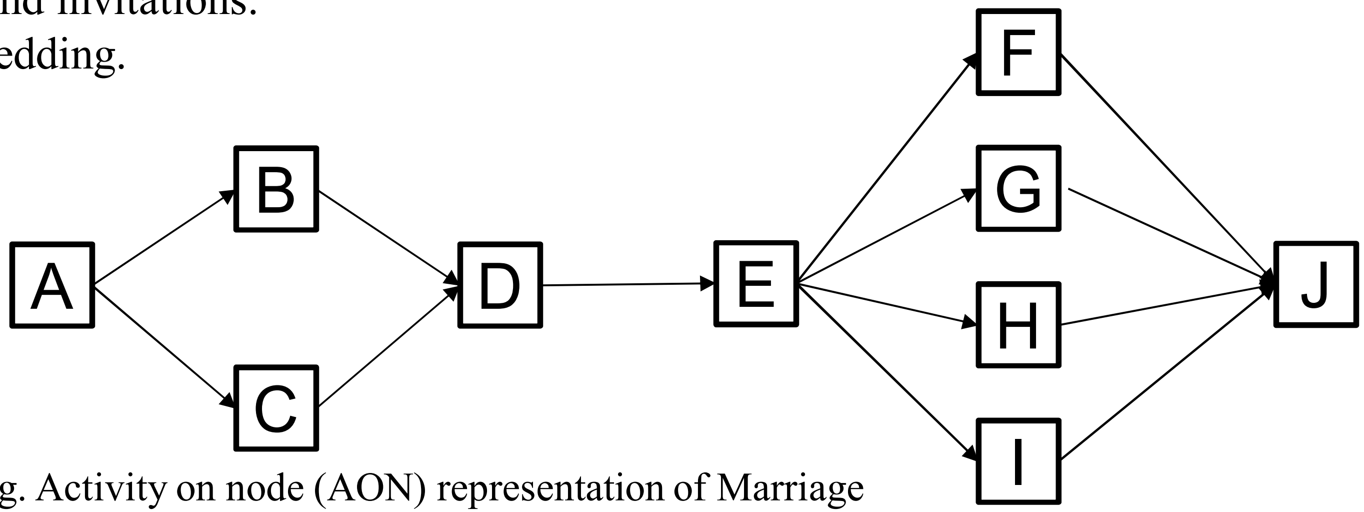


Fig. Activity on node (AON) representation of Marriage

Figure 12.1 UAV (Unmanned Aerial Vehicle)

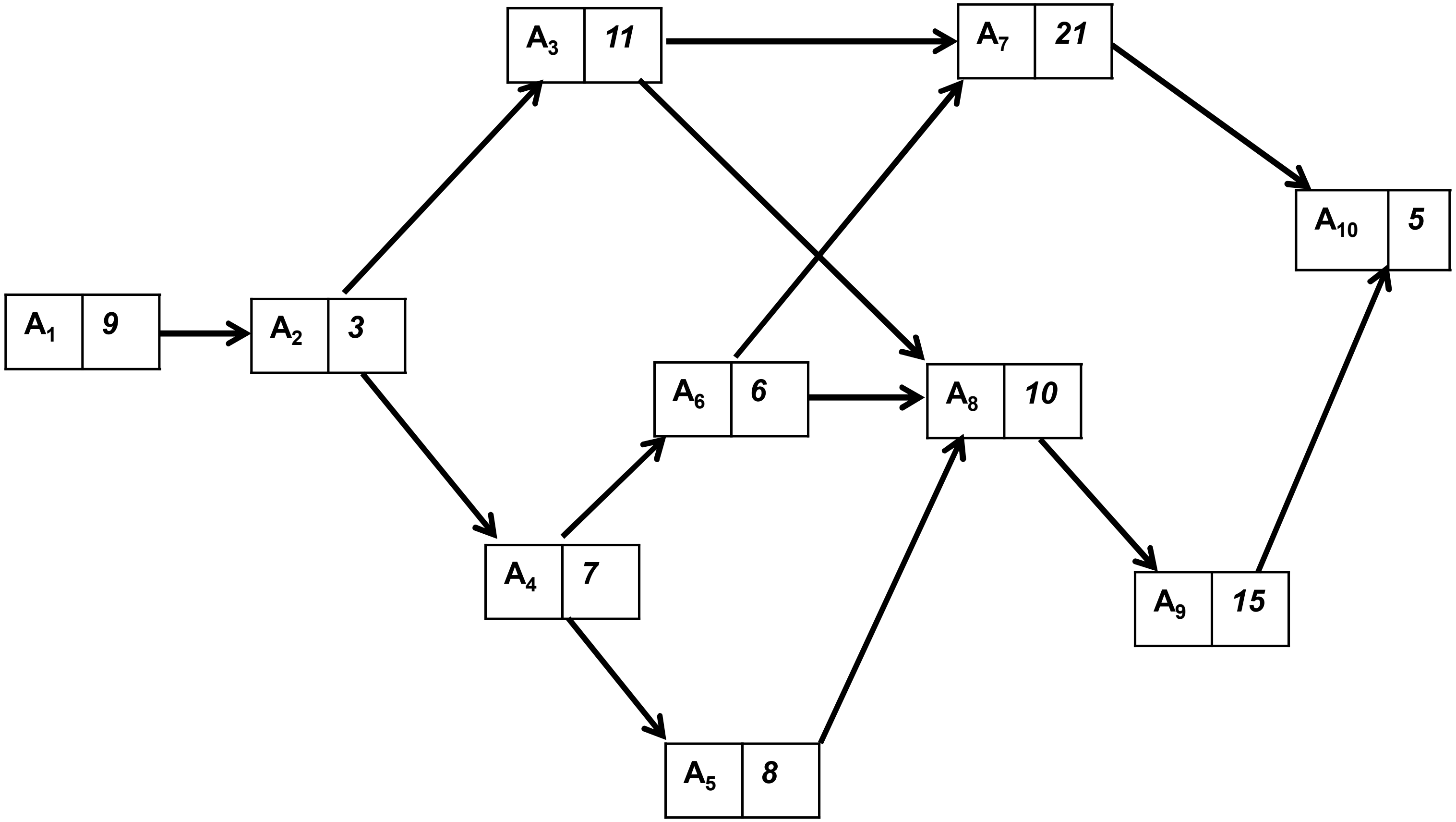
Table 12.1 Activities for the UAV Proposal Development

Activity	**Description**	**Expected duration (days)**
A_1	Prepare preliminary functional and ~	9
A_2	Prepare and discuss surface models	3
A_3	Perform aerodynamics analysis and evaluation	11
A_4	Create initial structural geometry	7
A_5	Develop structural design conditions	8
A_6	Perform weights and inertia analyses	6
A_7	Perform structure and compatibility analyses	21
A_8	Develop balanced free-body diagrams	10
A_9	Establish internal load distributions	15
A_{10}	Prepare proposal	5

Table 12.2 Dependency Matrix for the UAV

UAV dependency matrix		Information providing activity (upstream)									
		A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
Information receiving activity (downstream)	A_1										
	A_2	X									
	A_3		X								
	A_4		X								
	A_5				X						
	A_6				X						
	A_7			X			X				
	A_8			X		X	X				
	A_9								X		
	A_{10}							x		X	

Figure 12.2 Activity on Node(AON) representation of the UAV project



How to find the critical path? (Naive Method)

Critical path = path with the longest duration

How many paths are there in the example?
How many paths are there in the real project?

This duration is equal to the **duration of the overall project!**

Note that A_7 is not on the critical path.

Finding the critical path (IE Method)

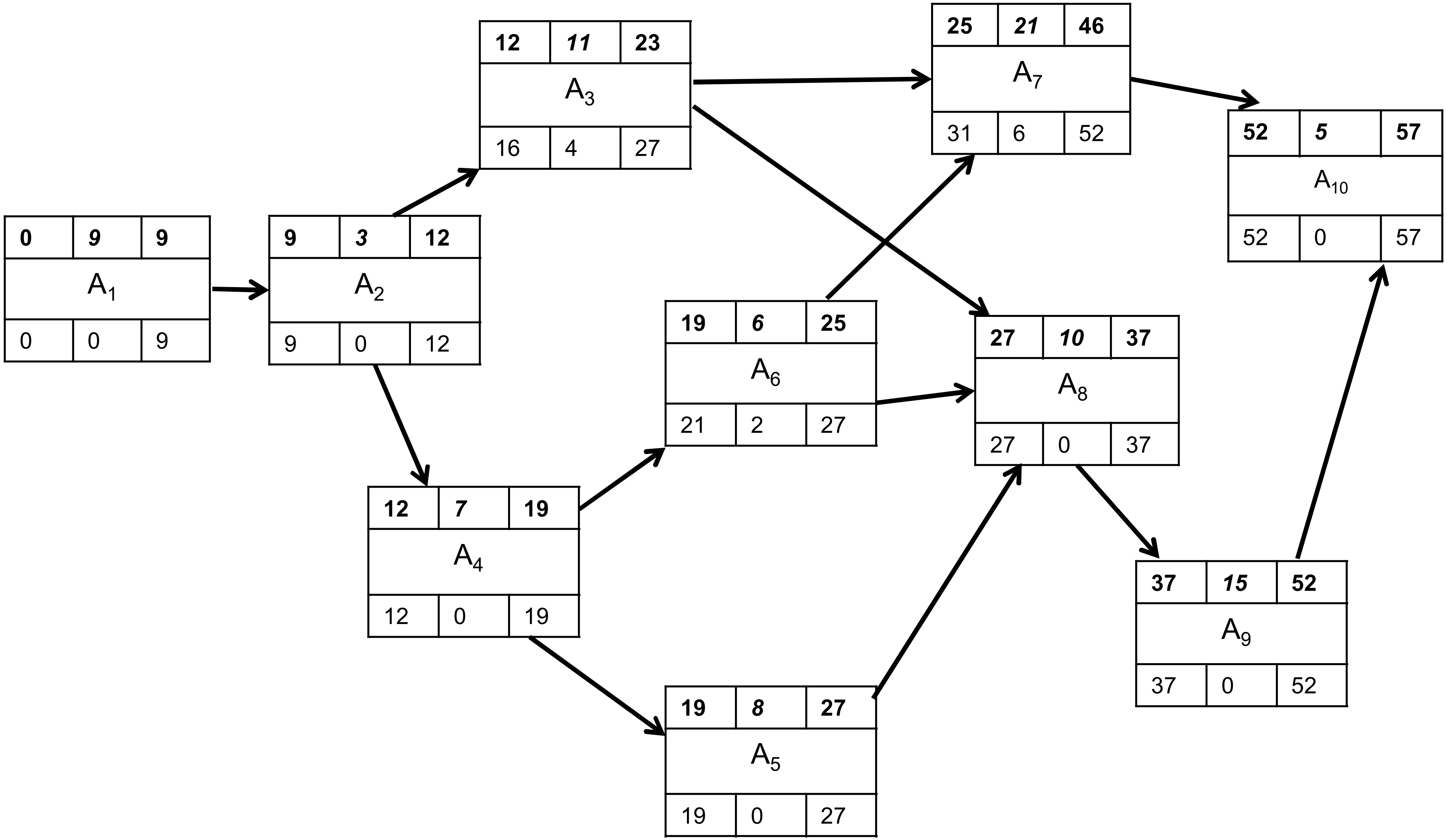
Activity	EST	duration (days)	ECT
A_1	0	9	9
A_2	ECT(A1)=9	3	12
A_3	ECT(A2)=12	11	23
A_4	ECT(A2)=12	7	19
A_5	ECT(A4)=19	8	27
A_6	ECT(A4)=19	6	25
A_7	**Max{ECT(A3),ECT(A6)} = Max{23,25}=25**	21	46
A_8	**Max{ECT(A3), ECT(A5),ECT(A6)} =Max{23, 27, 25}=27**	10	37
A_9	ECT(A8)=37	15	52
A_{10}	ECT(A9)=52	5	57

Slack Time = Latest Start Time – Earliest Start Time

Activity	EST	Duration	ECT	LCT	LST=LCT-Duration	Slack=LCT-ECT
A_1	0	9	9	$LST(A_2)=9$	9-9=0	0
A_2	9	3	12	$Min\{LST(A_3),LST(A_4)\} = Min\{16,12\}=12$	12-3=9	0
A_3	12	11	23	$Min\{LST(A_7),LST(A_8)\} = Min\{31,27\}=27$	27-11=16	27-23=4
A_4	12	7	19	$Min\{LST(A_5),LST(A_6)\} = Min\{19,21\}=19$	19-7=12	0
A_5	19	8	27	$LST(A_8)=27$	27-8=19	0
A_6	19	6	25	$Min\{LST(A_7),LST(A_8)\} = Min\{31,27\}=27$	27-6=21	27-25=2
A_7	25	21	46	$LST(A_{10})=52$	52-21=31	52-46=6
A_8	27	10	37	$LST(A_9)=37$	37-10=27	0
A_9	37	15	52	$LST(A_{10})=52$	52-15=37	0
A_{10}	52	5	57	57	57-5=52	0

$$LCT_i = Min_{\forall j \text{ which is successor of } i} \{LST_j\} \Rightarrow LST \text{ (backtracking)}$$

Critical path= set of activities with their slack=0!


0 9 9
A1
0 0 9
9 3 12
A2
9 0 12
12 11 23
A3
16 4 27
12 7 19
A4
12 0 19
19 8 27
A5
19 0 27
19 6 25
A6
21 2 27
25 21 46
A7
31 6 52
27 10 37
A8
27 0 37
37 15 52
A9
37 0 52
52 5 57
A10
52 0 57

Dealing with Uncertainty

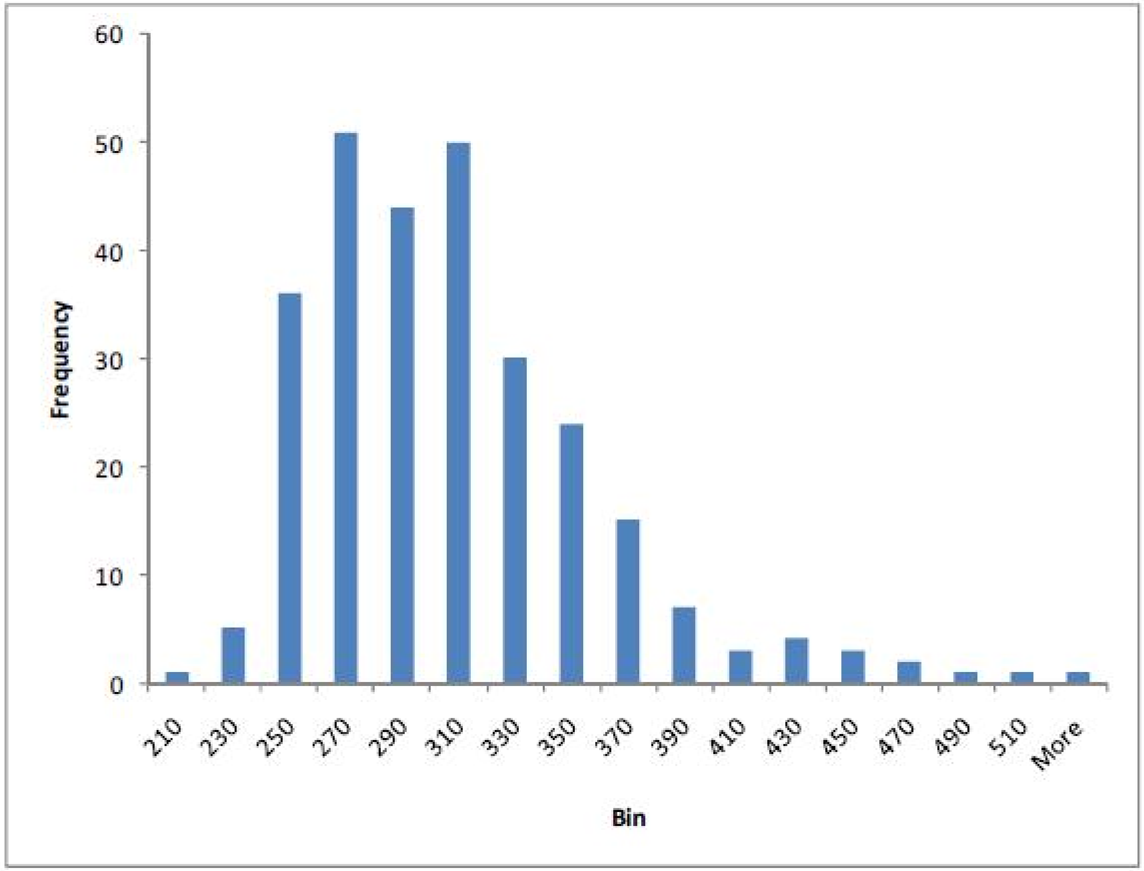
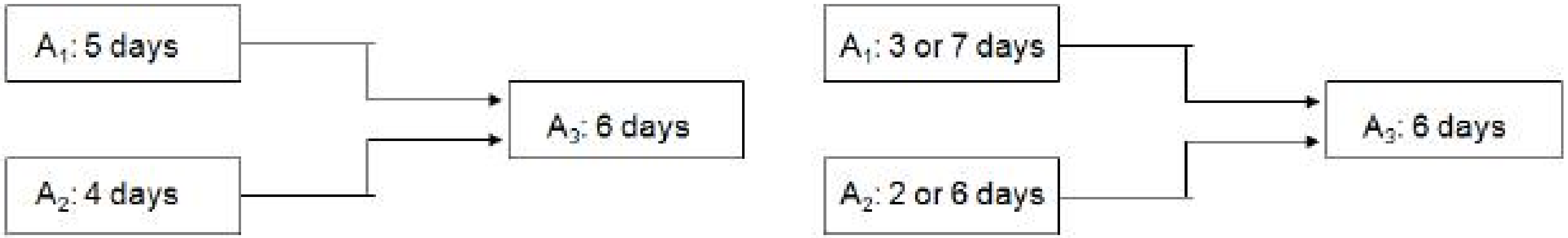


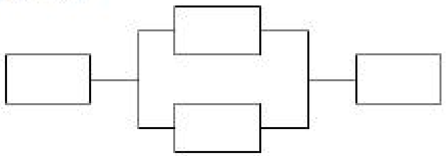
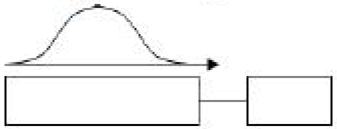
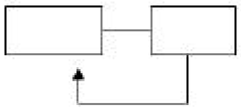
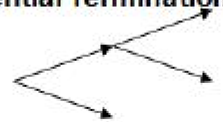
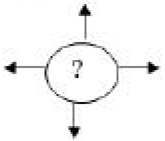
Figure 12.6 Simple example of a project with uncertainty



Scenario	Probability	Explanation	Start of A3	Completion
A1 late and A2 late	0.25	A1 would take 7 days (during which time the 6 days of A2 will also be completed)	7	13
A1 early, A2 late	0.25	A2 would take 6 days (during which time the 3 days of A1 would also be completed)	6	12
A1 late, A2 early	0.25	A1 would take 7 days (during which time the 2 days of A2 would also be completed)	7	13
A1 early and A2 early	0.25	A1 would take 3 days (during which time the 2 days of A2 would also be completed)	3	9

Expected completion time=0.25* (13+12+13+9)=**11.75days!**

We will be running later with **75%** probability!

Certainty 	Low uncertainty project such as construction projects or routine development projects	Calculate critical path Use slack to optimize timing
Uncertainty in activity duration 	Projects with minor uncertainties about activity durations and or resource availability	Monte Carlo Analysis – watch for changes in critical path
Potential iteration 	Potentially iterative projects that include one or multiple rework loops	Design Structure Matrix
Potential Termination 	Multiple scenarios exist, one or more of them require termination of the project	Scenario analysis and decision trees
Unk-Unks 	High levels of uncertainty and a dynamic environment; chaos	Discovery driven planning

How to accelerate projects?

Project completion time vs Budget vs Project quality
(cf. outsourcing some activities)

1. Start the project early! (cf. term project syndrome)

2. Manage the project scope! (cf. 설계 변경)

3. Crash activities!

4. Overlap critical path activities!

How to accelerate projects? (Crashing)

Crashing requires Money!

Marines Forever! Once a Marine, Always a Marine!

CP Forever! 한 번 CP에 속하면, 영원히 CP에 속한다!

Crashing

Direct Costs and Times

- Normal Time (T_N)
- Normal Cost (C_N)
- Crash Time (T_C)
- Crash Cost (C_C)

$$\text{Cost Coefficient (CC)} = \frac{C_C - C_N}{T_N - T_C}$$

(cf. Crashing involves a time-cost tradeoff)

Assumption : costs increase linearly as activity time is reduced from its normal time.

Crashing

Crashing the critical path is shortening the durations of critical path activities by adding resources.

Systematic Crashing Method

(Step 1) Identify the critical activities.

(Step 2) Choose the critical activity with the least CC

(Step 3) Crash the activity until

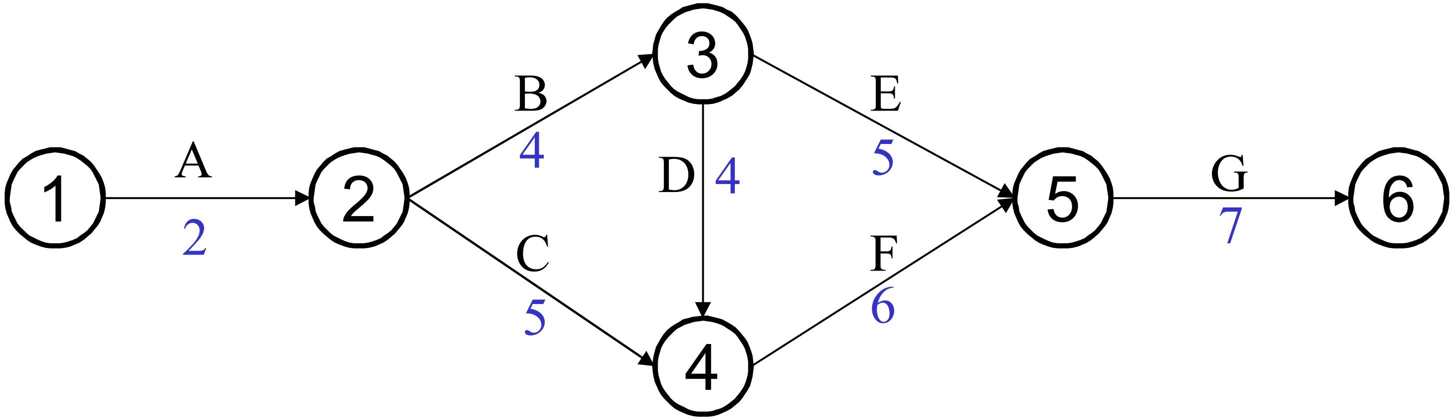
(i) no further reduction is possible → GO TO Step 2

(ii) another path become critical → GO TO Step 2

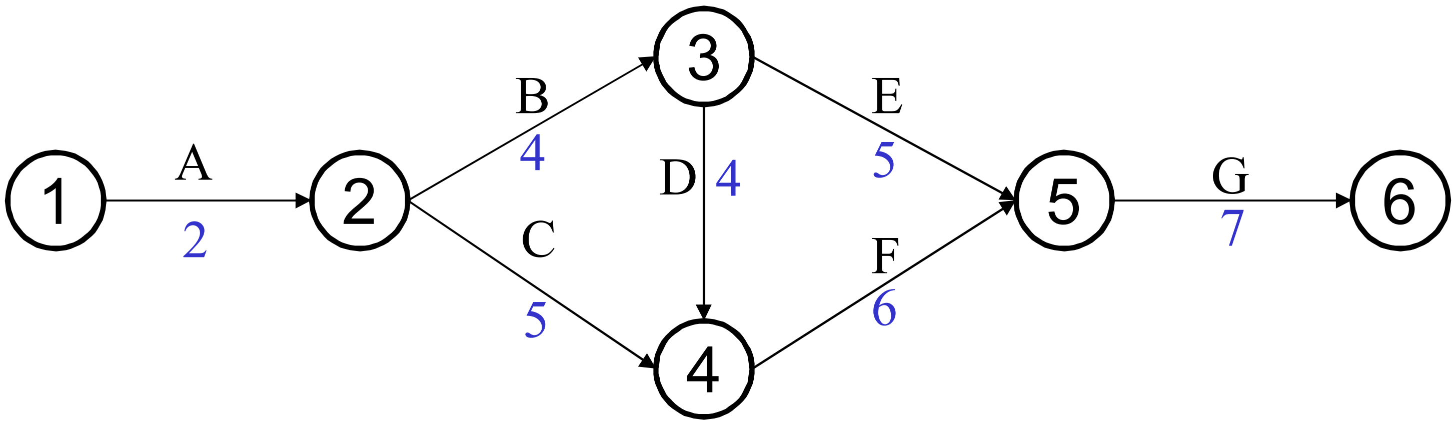
(iii) the increase in direct cost exceeds the savings that result from crashing the project → STOP

Crashing: Example 1

Activity	Predec.	T_N	C_N	T_C	C_C	CC
A	-	2	50 (000)	-	-	-
B	A	4	100	2	140	20
C	A	5	110	-	-	-
D	B	4	90	3	99	9
E	B	5	130	1	218	22
F	C,D	6	100	2	196	24
G	E,F	7	106	-	-	-



Crashing: Example 1



If all activities are crashed,

Critical Path : (i) A → B → D → F → G (ii) A → C → F → G

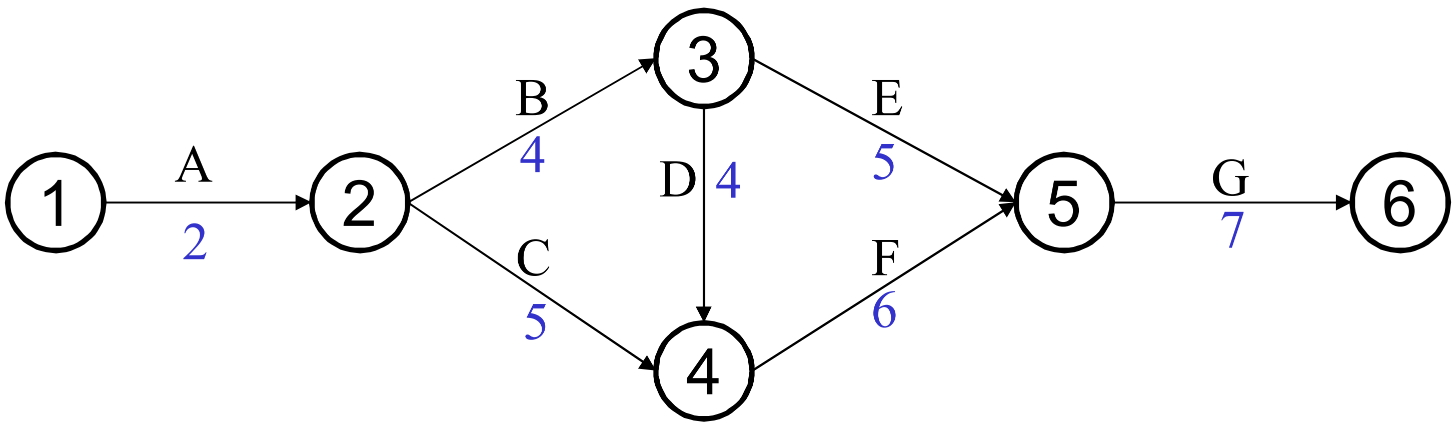
Project Completion Time : 16 weeks

Project Completion Cost : $919,000

(Question) Do we really need to crash all activities to complete project in 16 weeks?

(Answer) Use Systematic Crashing Method

Crashing: Example 1

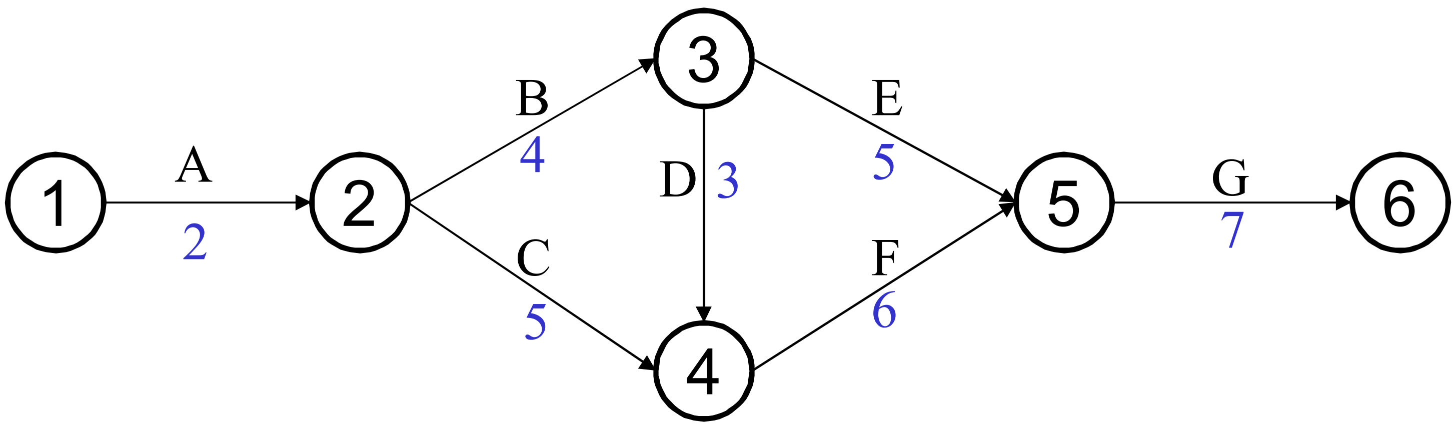


Critical Path : A → B → D → F → G

Project Completion Time : 23 weeks

Project Completion Cost : $686,000

Crashing: Example 1



(*iteration 1*)

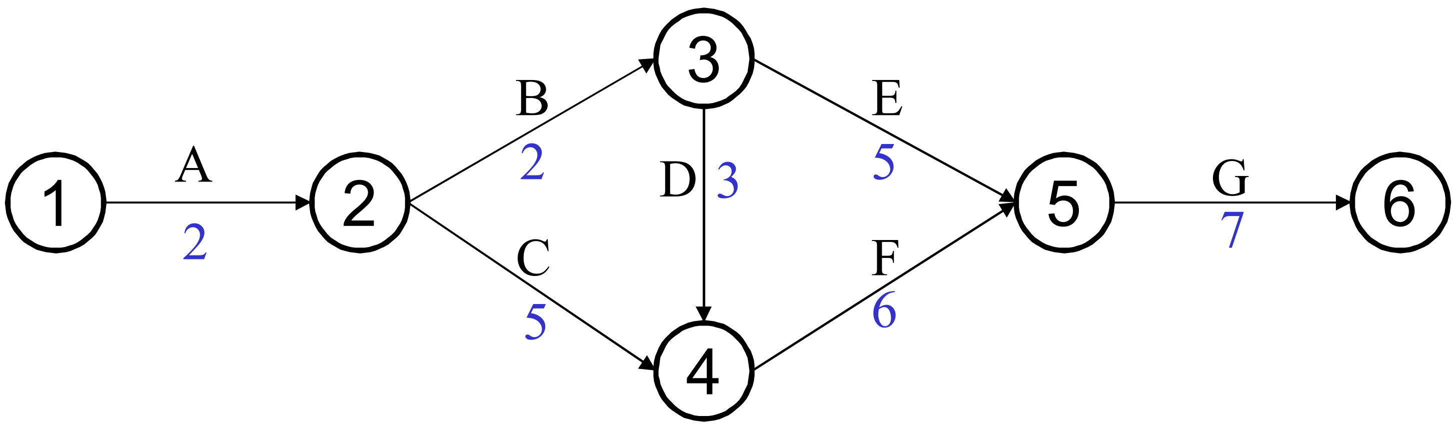
Activity D has minimum CC : 9 We crash D by 1 week.

Critical Path : A → B → D → F → G

Project Completion Time : 22 weeks

Project Completion Cost : \$686,000 + \$9,000 = \$695,000

Crashing: Example 1



(*iteration 2*)

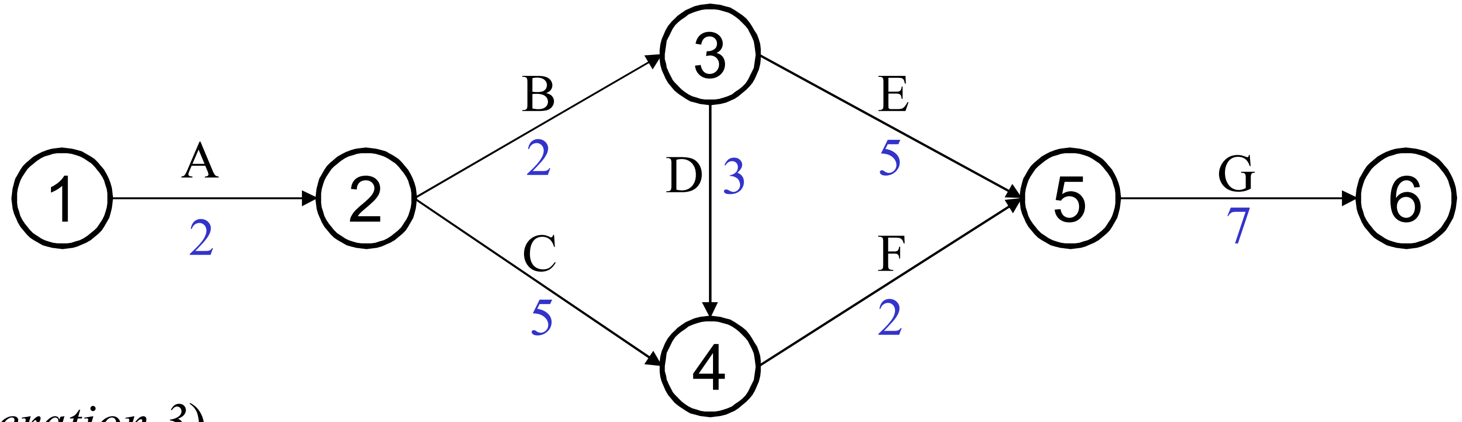
Activity B has minimum CC : 20 We crash B by 2 weeks.

Critical Path : (i) A → B → D → F → G (ii) A → C → F → G

Project Completion Time : 20 weeks

Project Completion Cost : \$695,000 + $2 \times CC_B$ = \$735,000

Crashing: Example 1



(*iteration 3*)

Now we have to consider both critical paths.
If we crash F by 1 week, what will be the project completion time?

Activity F has minimum CC : 24 We crash F by 4 weeks.
Critical Path : (i) A → B → D → F → G (ii) A → C → F → G (iii) A → B → E → G
Project Completion Time : 16 weeks
Project Completion Cost : \$735,000 + 4 × CC_F = \$831,000
(cf.) Systematic Crashing has resulted in a saving of \$919,000 - \$831,000 = \$88,000 compared with the all-crashing case.

Crashing : Example 2

Assume that you have a project in which only one activity can be crashed. The activity can be crashed from 13 days to 11 days at an additional cost of $2000. The current daily overhead cost for this project are $1200. Should this activity be crashed? If it should, how much should it be crashed?

Crashing Cost = $1000/day,   Overhead Cost = $1200/day

(*Step 1*) Is the activity on the CP?

If Yes, Go to *Step 2*. Else, stop and do not crash.

Crashing : Example 2

(*Step 2*) Is there more than 1 CP?

If Yes, Go to *Step 3*.

Else, (i) Crash 2 days if no other path becomes critical, and stop.

(ii) Crash 1 day if another path becomes critical, and Ask.

Is the activity a common activity on all CPs?

If Yes, crash one more day, and stop.

Else, do not crash, and stop.

(*Step 3*) Is the activity a common activity on all CPs?

If No, do not crash, and stop.

Else, crash 1 day, and repeat *Step 3*.

What is PERT

- PERT (Program evaluation and review technique)

✓ 1959년, Booz, Allen, and Hamilton Company와 미 해군이 폴라리스 미사일 개발 중에 프로젝트 관리 도구로 개발함

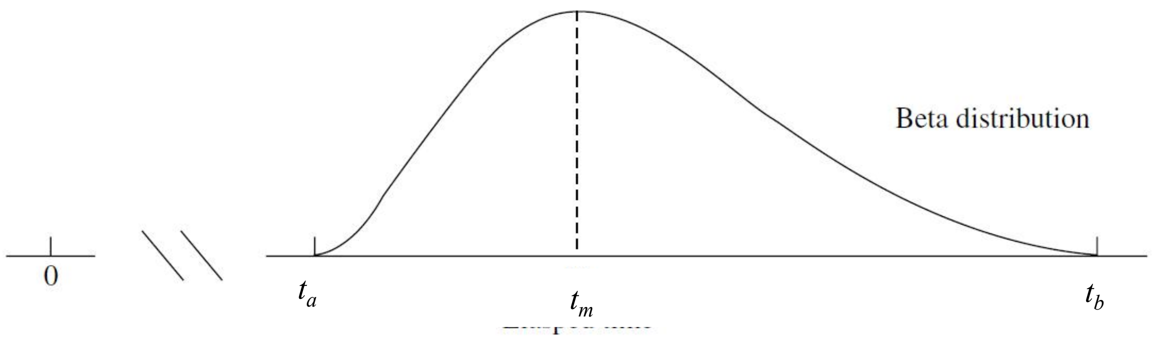
✓ PERT의 기본적인 요소는 각 활동의 소요시간을 확률로 예측하는 것

What is PERT

- CPM (Critical Path Method)는 각 활동에 확정적인 시간이 소요된다고 가정함
- 이에 비해, PERT는 이를 확률변수로 취급함
- 각 활동의 소요시간에 대해, 다음 세 가지 데이터가 필요함

> t_a (낙관적)
> t_b (비관적)
> t_m (most likely)

$\boxed{(t_a \le t_m \le t_b)}$



How does PERT work

- 활동 시간에 대한 확률 분포
  - 활동 시간의 기댓값 계산

$$t_e = \frac{t_a + 4t_m + t_b}{6}$$

CPM (Critical Path Method) 을 통하여 critical path (주경로 또는 위급경로)를 구함

  - 활동 시간의 표준편차 계산

$$\sigma = \frac{t_b - t_a}{6}$$

$t_c = \sum_{\substack{critical\\path}} t_e$ 와 $\sigma_c^2 = \sum_{\substack{critical\\path}} \sigma^2$ 가 각각 프로젝트 완료 시간의 기댓값과 분산이 됨

베타 분포: Clark (1962)

PERT Example

Activity	t_a	t_m	t_b	t_e	σ^2
A	1	2	3	2	$\frac{1}{9}$
B	2	$3\frac{1}{2}$	8	4	1
C	6	9	18	10	4
D	4	$5\frac{1}{2}$	10	6	1
E	1	$4\frac{1}{2}$	5	4	$\frac{4}{9}$
F	4	4	10	5	1
G	5	$6\frac{1}{2}$	11	7	1
H	5	8	17	9	4
I	3	$7\frac{1}{2}$	9	7	1
J	3	9	9	8	1
K	4	4	4	4	0
L	1	$5\frac{1}{2}$	7	5	1
M	1	2	3	2	$\frac{1}{9}$
N	5	$5\frac{1}{2}$	9	6	$\frac{4}{9}$

주경로

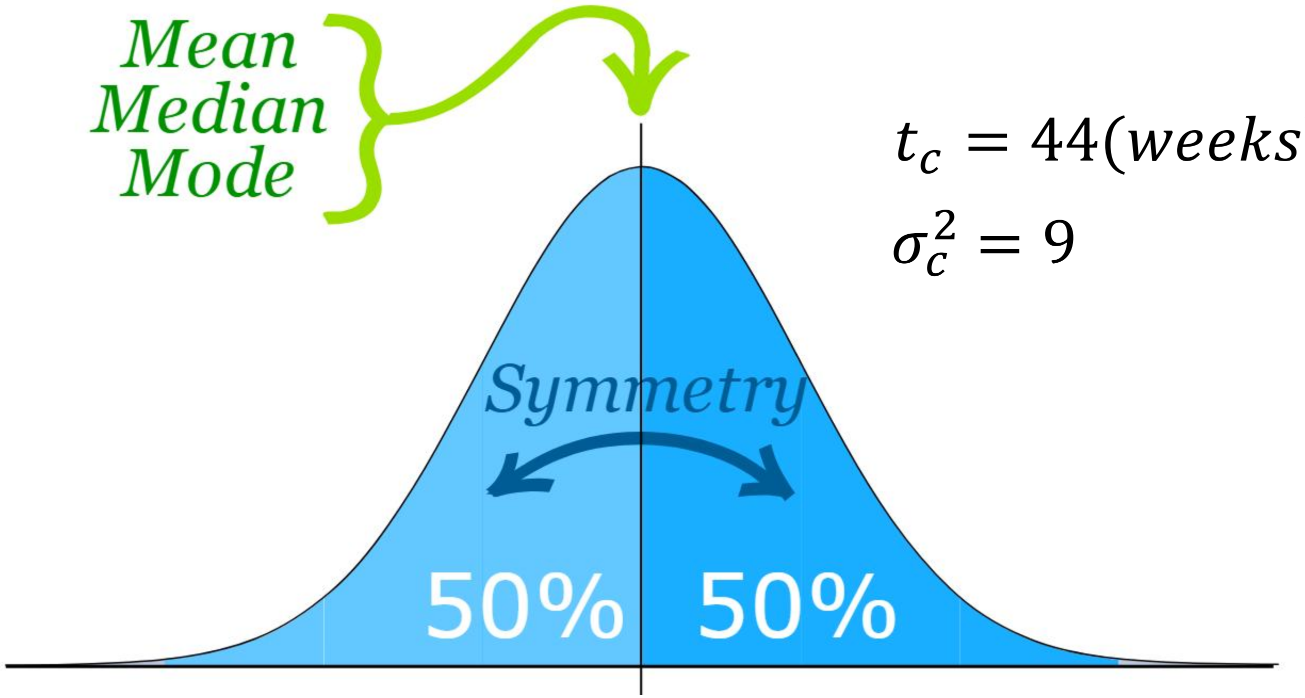
$A \to B \to C \to E \to F \to J \to L \to N$

$$t_c = 2 + 4 + 10 + 4 + 5 + 8 + 5 + 6$$
$$= 44(weeks)$$

$$\sigma_c^2 = \frac{1}{9} + 1 + 4 + \frac{4}{9} + 1 + 1 + 1 + \frac{4}{9}$$
$$= 9$$

PERT Example

- 공사를 44주 안에 완료할 확률: 50%

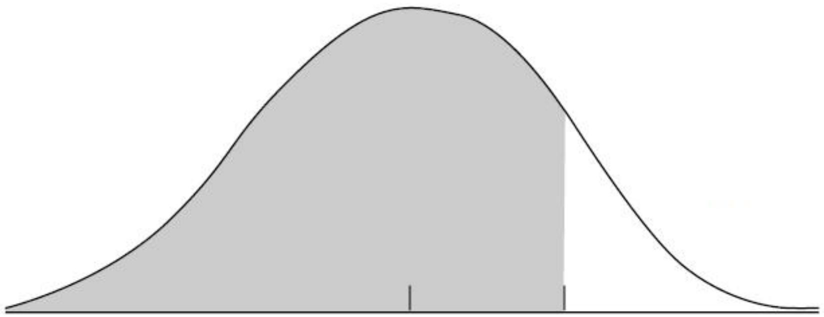


$$t_c = 44(weeks)$$
$$\sigma_c^2 = 9$$

PERT Example

- 공사를 47주 안에 완료할 확률의 계산
- X : 공사 기간 (주)

$$P(X \leq 47) = P\left(Z \leq \frac{47-44}{\sqrt{9}}\right) = P(Z \leq 1)$$
$$= 1 - 0.1587 \approx 0.84$$



$$t_c = 44(weeks)$$
$$\sigma_c^2 = 9$$

PERT Example

- 공사 기간의 95% 신뢰 구간

$$P(a \le X \le b) = 0.95$$

$$P\left(\frac{a-44}{\sqrt{9}} \le Z \le \frac{b-44}{\sqrt{9}}\right) = 0.95$$

- 정규분포표에서,

$$\frac{a-44}{\sqrt{9}} = -1.96 \qquad \frac{b-44}{\sqrt{9}} = 1.96$$

$$\therefore a = 38.12, b = 49.88\ (weeks)$$

$$t_c = 44(weeks)$$

$$\sigma_c^2 = 9$$

PERT 정리

- 불확실한 상황과 PERT

➢ 확정적인 상황을 가정한 critical path 의 적절성을 가늠함

✓ 우리가 중요하다고 생각한 프로세스가 실제로 그러하였는지 알기 어려움

✓ 자원을 낭비하고도 제대로 된 관리 효과를 얻지 못하는 경우가 많이 나타남

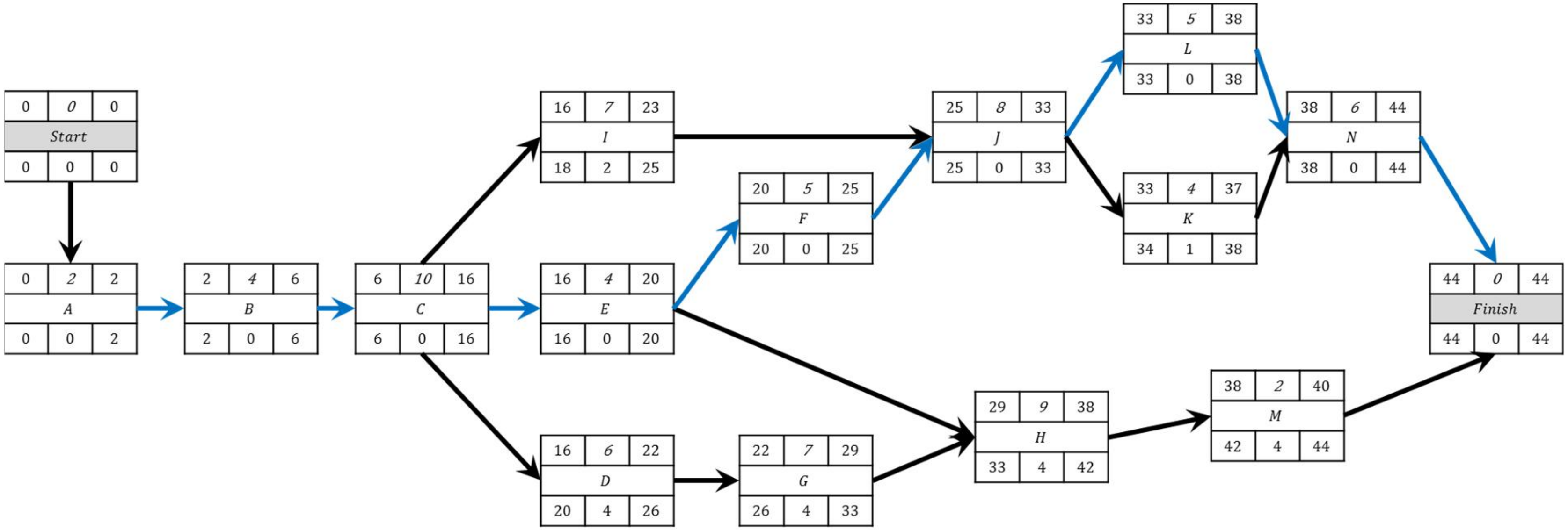
PERT의 한계 (example)

- 불확실한 상황과 PERT

Activity	Activity Description	Immediate Predecessors	Estimated Duration
A	Excavate	—	2 weeks
B	Lay the foundation	*A*	4 weeks
C	Put up the rough wall	*B*	10 weeks
D	Put up the roof	*C*	6 weeks
E	Install the exterior plumbing	*C*	4 weeks
F	Install the interior plumbing	*E*	5 weeks
G	Put up the exterior siding	*D*	7 weeks
H	Do the exterior painting	*E, G*	9 weeks
I	Do the electrical work	*C*	7 weeks
J	Put up the wallboard	*F, I*	8 weeks
K	Install the flooring	*J*	4 weeks
L	Do the interior painting	*J*	5 weeks
M	Install the exterior fixtures	*H*	2 weeks
N	Install the interior fixtures	*K, L*	6 weeks

PERT의 한계 (example)

- 불확실한 상황과 PERT



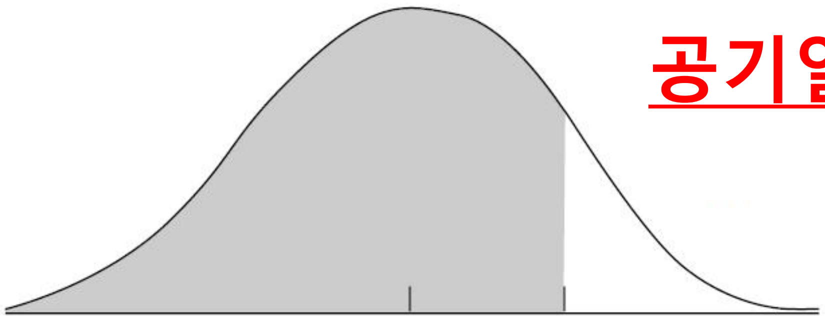
✓ $\boldsymbol{Start} \rightarrow A \rightarrow B \rightarrow C \rightarrow E \rightarrow F \rightarrow J \rightarrow L \rightarrow N \rightarrow \boldsymbol{Finish}$가 critical path라고 나타났다면, 정말 이들 프로세스만 관리하면 될까?

앞의 예로 돌아가서...

- 공사를 47주 안에 완료할 확률의 계산
- X: 공사 기간 (주)

$$P(X \le 47) = P\left(Z \le \frac{47-44}{\sqrt{9}}\right) = P(Z \le 1)$$
$$= 1 - 0.1587 \approx 0.84$$

<u>**공기일을 맞출 확률**</u>

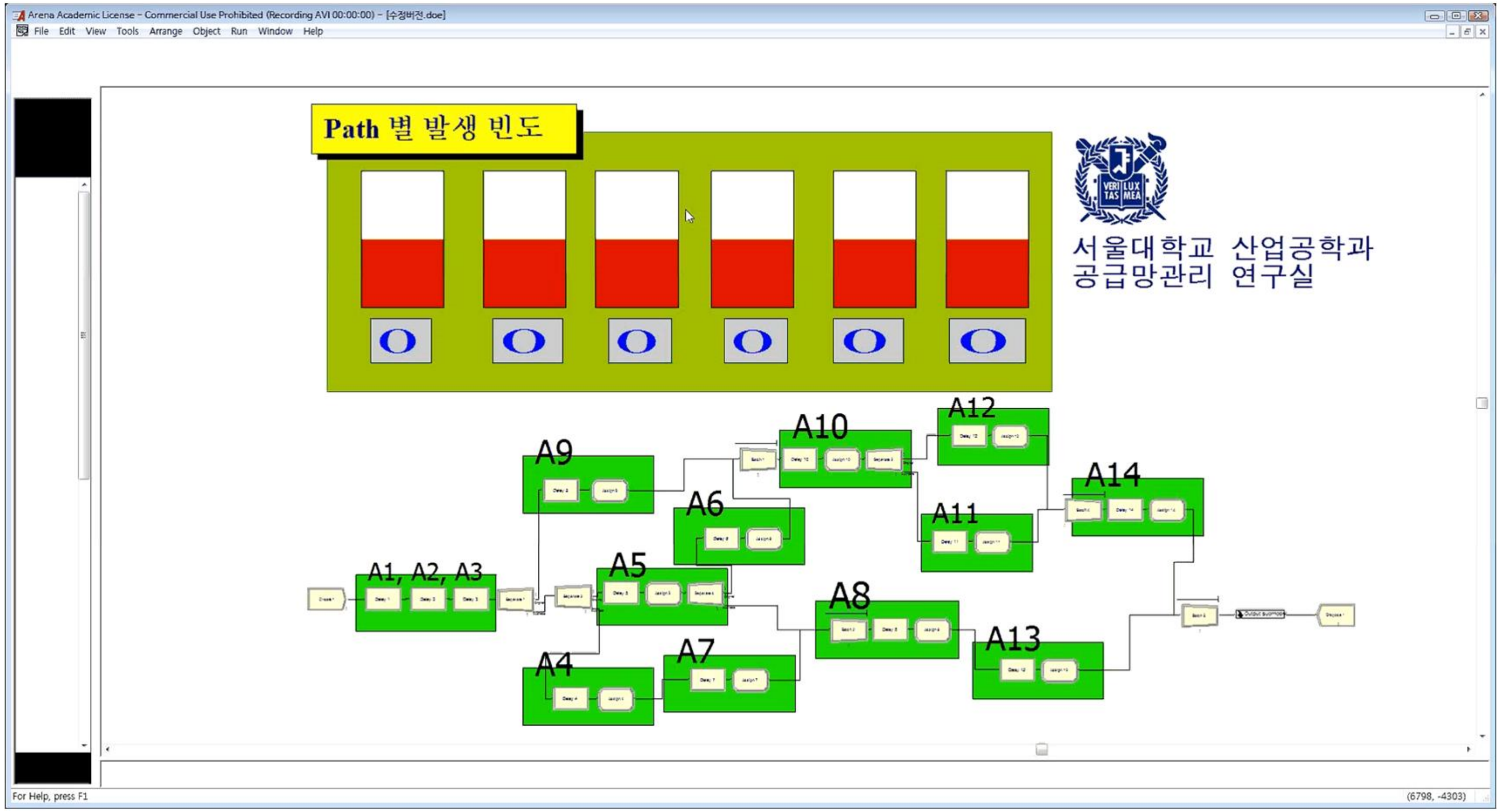


$t_c = \mathbf{44}$ **(weeks)**
$\sigma_c^{\,2} = \mathbf{9}$

시뮬레이션을 이용한 CPM

- 시뮬레이션을 이용한 CPM
  - ➢ 복잡한 시스템의 경우 시뮬레이션을 이용한 관리가 필요함
    - ✓불확실성 하에서 실제 프로세스들 사이의 시간적인 상호작용을 알 수 있음
    - ✓각 path가 critical path가 되는 확률을 알 수 있음
    - ✓어떤 프로세스를 관리해야 하는지 알 수 있음

시뮬레이션을 이용한 CPM



시뮬레이션을 이용한 CPM

- 10000번의 시뮬레이션 실험 결과

```
==========================================================================================
10000 paths in total
6 candidates found !
==========================================================================================
candidate  1 :  [1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1] , featured time:  4197
candidate  2 :  [1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1] , featured time:  577
candidate  3 :  [1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0] , featured time:  1740
candidate  4 :  [1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0] , featured time:  1150
candidate  5 :  [1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1] , featured time:  2041
candidate  6 :  [1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1] , featured time:  295

==========================================================================================
```

41.97 % 확률로 critical path

58.03 % 확률로 critical path

➢ 불확실성을 가정하지 않고 계산한 critical path:

$$\boldsymbol{Start \rightarrow A \rightarrow B \rightarrow C \rightarrow E \rightarrow F \rightarrow J \rightarrow L \rightarrow N \rightarrow Finish}$$

✓ 실제로는 58.03%의 확률로 타 path가 critical하다는 것을 관찰할 수 있음

➢ critical path 관리를 실패할 확률이 과반 이상!

시뮬레이션을 이용한 CPM

- 10000번의 시뮬레이션 실험 결과

```
====================================================================================================
10000 paths in total
6 candidates found !
====================================================================================================
candidate  1 :  [1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1] , featured time:  4197
candidate  2 :  [1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1] , featured time:  577
candidate  3 :  [1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0] , featured time:  1740
candidate  4 :  [1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0] , featured time:  1150
candidate  5 :  [1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1] , featured time:  2041
candidate  6 :  [1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1] , featured time:  295

====================================================================================================
```

41.97 % 확률로 critical path

58.03 % 확률로 critical path

➢ Critical 할 확률이 높은 path 상의 프로세스에 대한 관리가 필요함