추정 및 스케줄
Estimation & Scheduling

프로젝트 계획(Project Planning)

- Before the project can begin,

  the software team should estimate;

  - the work to be done,
  - the resources required,
  - and the time

  that will elapse from start to finish.

프로젝트 계획

The overall goal of project planning is to establish a pragmatic strategy for controlling, tracking, and monitoring a complex technical project.

Why?

So the end result gets done on time, with quality!

프로젝트 계획 Task Set (1/2)

- Establish project scope
- Determine feasibility
- Analyze risks
  - Risk analysis is considered.
- Define required resources
  - Determine require human resources
  - Define reusable software resources
  - Identify environmental resources

프로젝트 계획 Task Set (2/2)

- Estimate cost and effort
  - Decompose the problem
  - Develop two or more estimates using size, function points, process tasks or software equation, etc.
  - Reconcile the estimates
- Develop a project schedule
  - Scheduling is considered in detail.
    - Establish a meaningful task set
    - Define a task network
    - Use scheduling tools to develop a timeline chart
    - Define schedule tracking mechanisms

추정(Estimation)

- Estimation of resources, cost, and schedule for a software engineering effort requires
  - experience
  - access to good historical information (metrics)
  - the courage to commit to quantitative predictions when qualitative information is all that exists
- Estimation carries inherent risk and this risk leads to uncertainty

Scope를 이해하기 위해서는 ...

- understand the customers needs
- understand the business context
- understand the project boundaries
- understand the customer's motivation
- understand the likely paths for change
- understand that ...

Even when you understand, nothing is guaranteed!

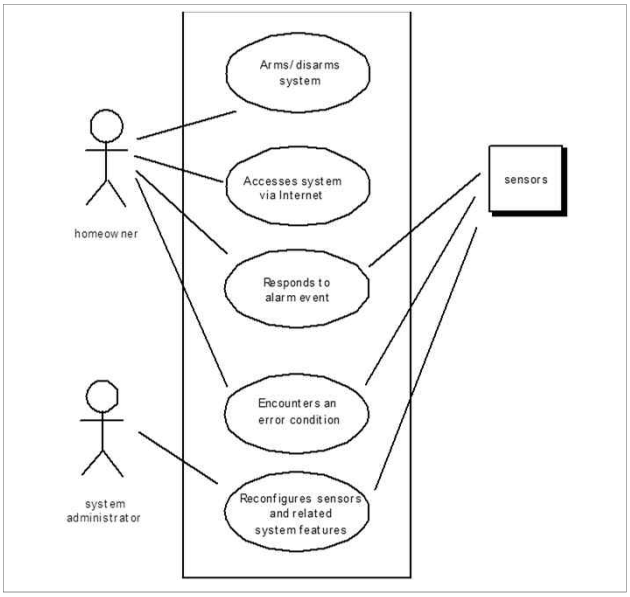
What is Scope?

- *Software scope* describes
  - the functions and features that are to be delivered to end-users
  - the data that are input and output
  - the "content" that is presented to users as a consequence of using the software
  - the performance, constraints, interfaces, and reliability that *bound* the system.
- Scope is defined using one of two techniques:
  - A narrative description of software scope is developed after communication with all stakeholders.
  - A set of use-cases is developed by end-users.

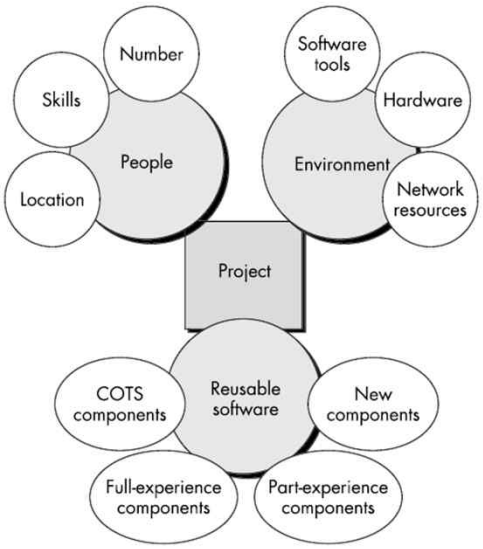
유즈케이스(Use-Case) ?

- describes system's behavior responds to user's request
- tells a stylized story about how an end-user interacts with the system

Use-Case Diagram



Resources



Project Estimation

- Project scope must be understood
- Elaboration (decomposition) is necessary
- Historical metrics are very helpful
- At least two different techniques should be used
- Uncertainty is inherent in the process

추정 요령

- Delay until late
- Past (similar) project experience
- Conventional estimation techniques
  - task breakdown and effort estimates
  - size (e.g., FP) estimates
- Empirical models
- Automated tools

Conventional Methods: LOC or FP Approach

- compute LOC or FP using estimates of information domain values
- use historical data to build estimates for the project

1. LOC Approach
(Example: CAD S/W)

Function	Estimated LOC
User interface and control facilities (UICF)	2,300
Two-dimensional geometric analysis (2DGA)	5,300
Three-dimensional geometric analysis (3DGA)	6,800
Database management (DBM)	3,350
Computer graphics display facilities (CGDF)	4,950
Peripheral control function (PCF)	2,100
Design analysis modules (DAM)	8,400
Estimated lines of code	*33,200*

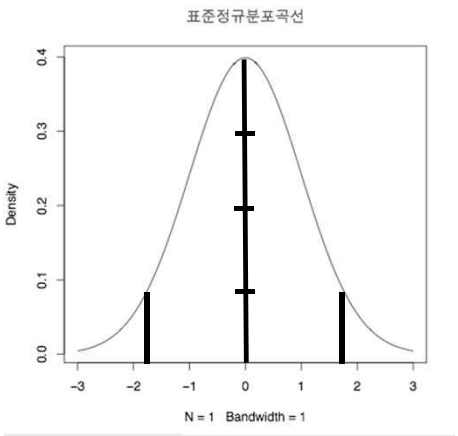
- Average productivity = 620 LOC / pm
- Average labor rate = $8,000 / pm

➔ Average cost rate = $13 / LOC

➔ estimated project cost = $13 x 33,200 = $431,600

➔ estimated effort = 33,200 / 620 = 54 PM

Example: LOC Approach



➔ 소프트웨어학과 3학년 평균 몸무게

➔ 최소값 45kg …. 중간값 55kg …. 최대값 75kg

➔ 평균 몸무게 추정값

➔ (45 + 4 x 55 + 75) / 6 = 56

$$S_{est} = (S_{opt} + 4 \times S_m + S_{pess}) / 6$$

2. FP Approach

Information domain value	Opt.	Likely	Pess.	Est. count	Weight	FP count
Number of external inputs	20	24	30	24	4	97
Number of external outputs	12	15	22	16	5	78
Number of external inquiries	16	22	28	22	5	88
Number of internal logical files	4	4	5	4	10	42
Number of external interface files	2	2	3	2	7	15
Count total						*320*

The estimated number of FP is derived:

$$FP_{estimated} = \text{count-total} * [0.65 + 0.01 * \Sigma (F_i)] = 375$$

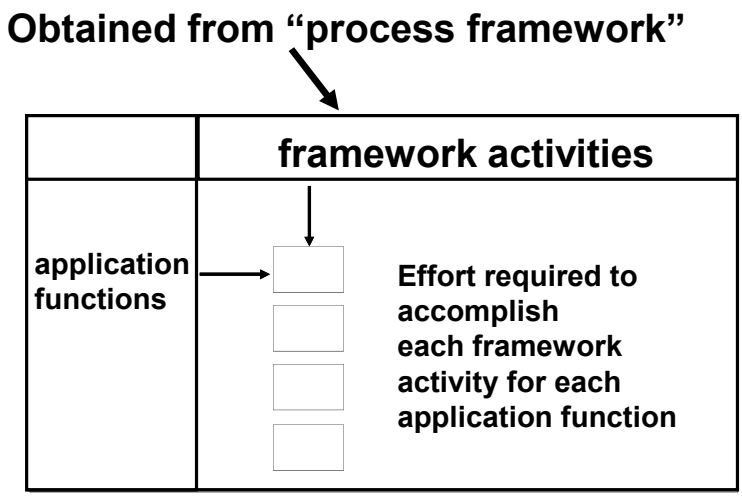
- Average productivity = 6.5 FP / pm
- Average labor rate = $8,000 / pm

➔ Average cost rate = $1,230 / FP

➔ estimated project cost = $1,230 × 375 = $461,250

➔ estimated effort = 375 / 6.5 = 58 PM

3. Process-Based Estimation



Process-Based Estimation Example

Activity →	CC	Planning	Risk analysis	Engineering		Construction release		CE	Totals
Task →				Analysis	Design	Code	Test		
Function ↓									
UICF				0.50	2.50	0.40	5.00	n/a	8.40
2DGA				0.75	4.00	0.60	2.00	n/a	7.35
3DGA				0.50	4.00	1.00	3.00	n/a	8.50
CGDF				0.50	3.00	1.00	1.50	n/a	6.00
DBM				0.50	3.00	0.75	1.50	n/a	5.75
PCF				0.25	2.00	0.50	1.50	n/a	4.25
DAM				0.50	2.00	0.50	2.00	n/a	5.00
Totals	0.25	0.25	0.25	3.50	20.50	4.50	16.50		46.00
% effort	1%	1%	1%	8%	45%	10%	36%		

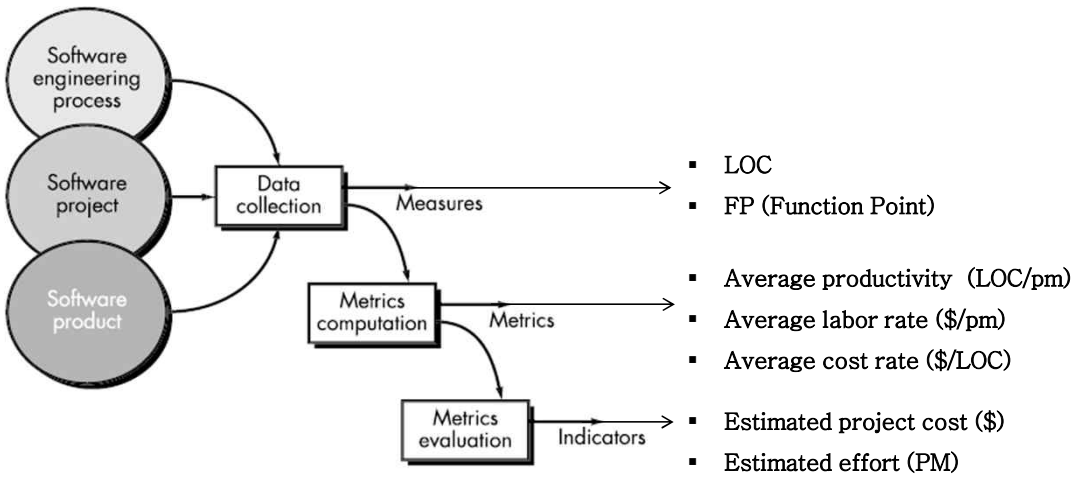
CC = customer communication CE = customer evaluation

- Average labor rate = $8,000 / pm

➔ estimated project cost = $8,000 x 46 = $368,000

➔ estimated effort = 46 PM

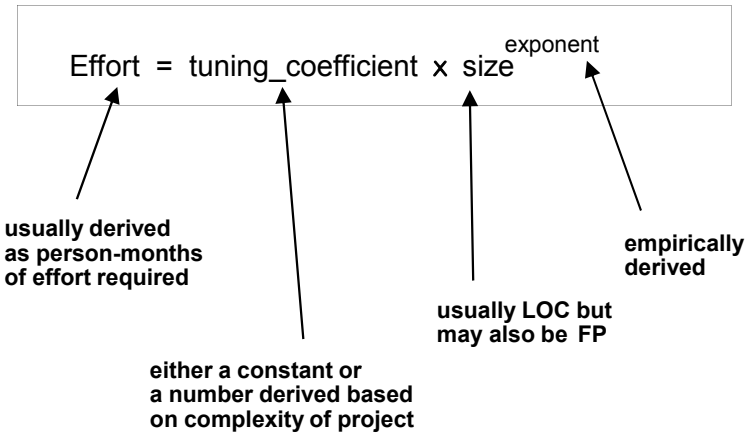
S/W Estimation 정리



- LOC
- FP (Function Point)
- Average productivity (LOC/pm)
- Average labor rate ($/pm)
- Average cost rate ($/LOC)
- Estimated project cost ($)
- Estimated effort (PM)

Empirical Estimation Models

General form:



4. Software Equation

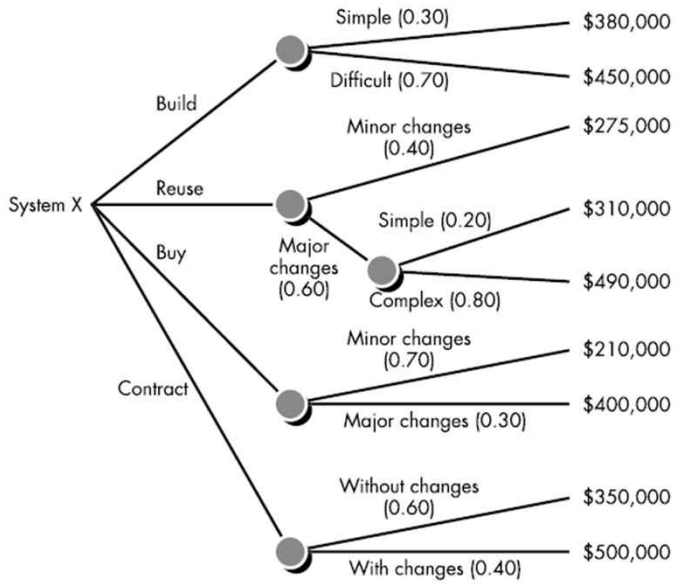
A dynamic multi-variable model

$$E = [LOC \times B^{0.333}] / P^3 \times (1 / t^4)$$

where

E = effort in person-months or person-years
t = project duration in months or years
B = special skill factor (e.g., 0.39)
P = productivity parameter
(e.g., embedded system 2,000)

The Make-Buy Decision



Computing Expected Cost

expected cost =
$\sum$(path probability)$_i$ × (estimated path cost)$_i$

For example, the expected cost to build is:

expected cost $_{build}$ = 0.30 ($380K) + 0.70 ($450K)
= $429K

expected cost $_{reuse}$ = $382K
expected cost $_{buy}$ = $267K
expected cost $_{contr}$ = $410K

Homework #1: 추정

1. Project Scope

(1) Functions

(2) Data: input & output

(3) Performance, Constraints, Interface 등등

2. Estimation

(1) LOC-based estimation

(2) FP-based estimation

(3) Process-based estimation

(4) Software equation based estimation

➔Average productivity, average labor rate, average cost rate, 기타 추정치 등은 각자 참고자료를 (또는 상상) 근거로 가정할 것!!!

➔ 과제는 항상 **2**주 뒤 강의 후 **12**시까지 전자관**411**호 과제함에 제출

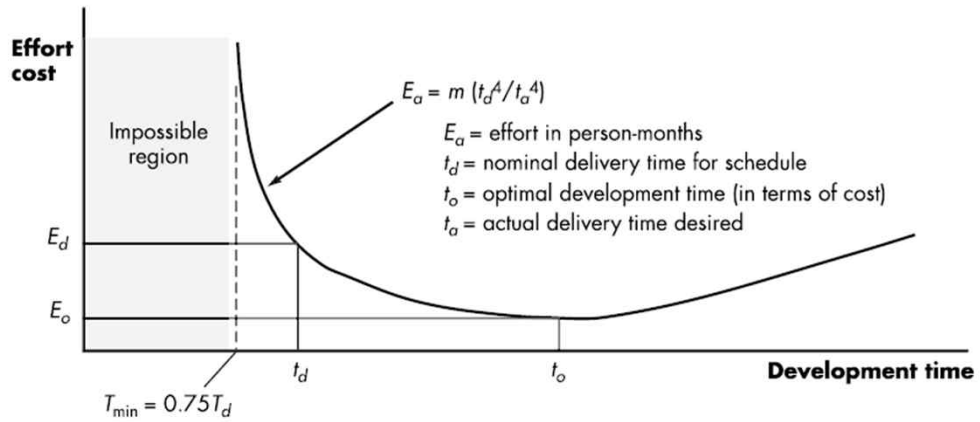
Why Are Projects Late?

- an unrealistic deadline established by someone outside the software development group
- changing customer requirements that are not reflected in schedule changes;
- an honest underestimate of the amount of effort and/or the number of resources that will be required to do the job;
- predictable and/or unpredictable risks that were not considered when the project commenced;
- technical difficulties that could not have been foreseen in advance;
- human difficulties that could not have been foreseen in advance;
- miscommunication among project staff that results in delays;
- a failure by project management to recognize that the project is falling behind schedule and a lack of action to correct the problem

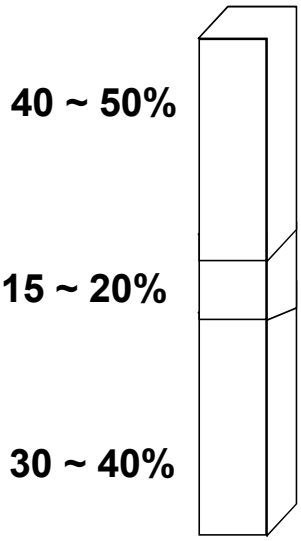
Scheduling Principles

- compartmentalization — define distinct tasks
- interdependency — indicate task interrelationship
- effort validation — be sure resources are available
- defined responsibilities — people must be assigned
- defined outcomes — each task must have an output
- defined milestones — review for quality

Effort and Delivery Time

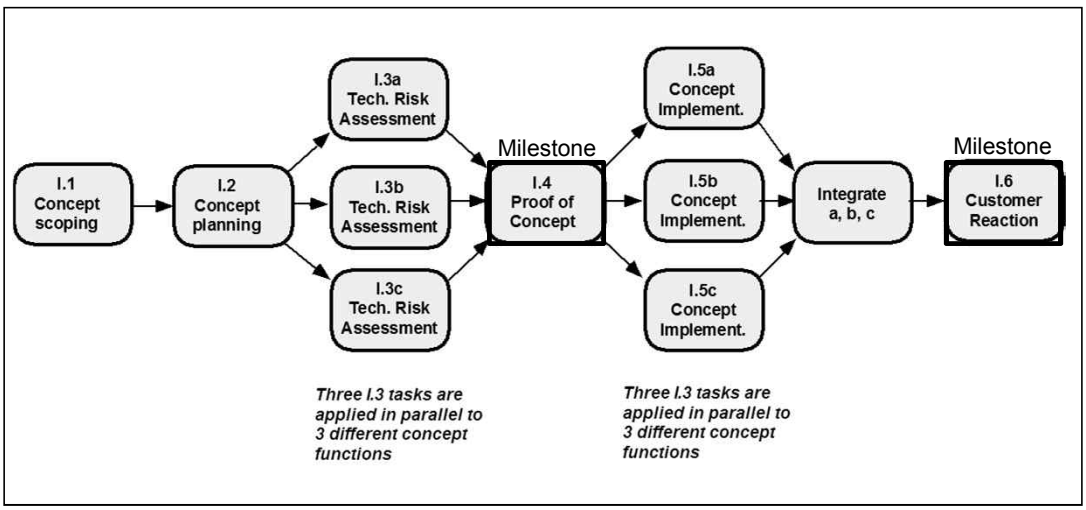


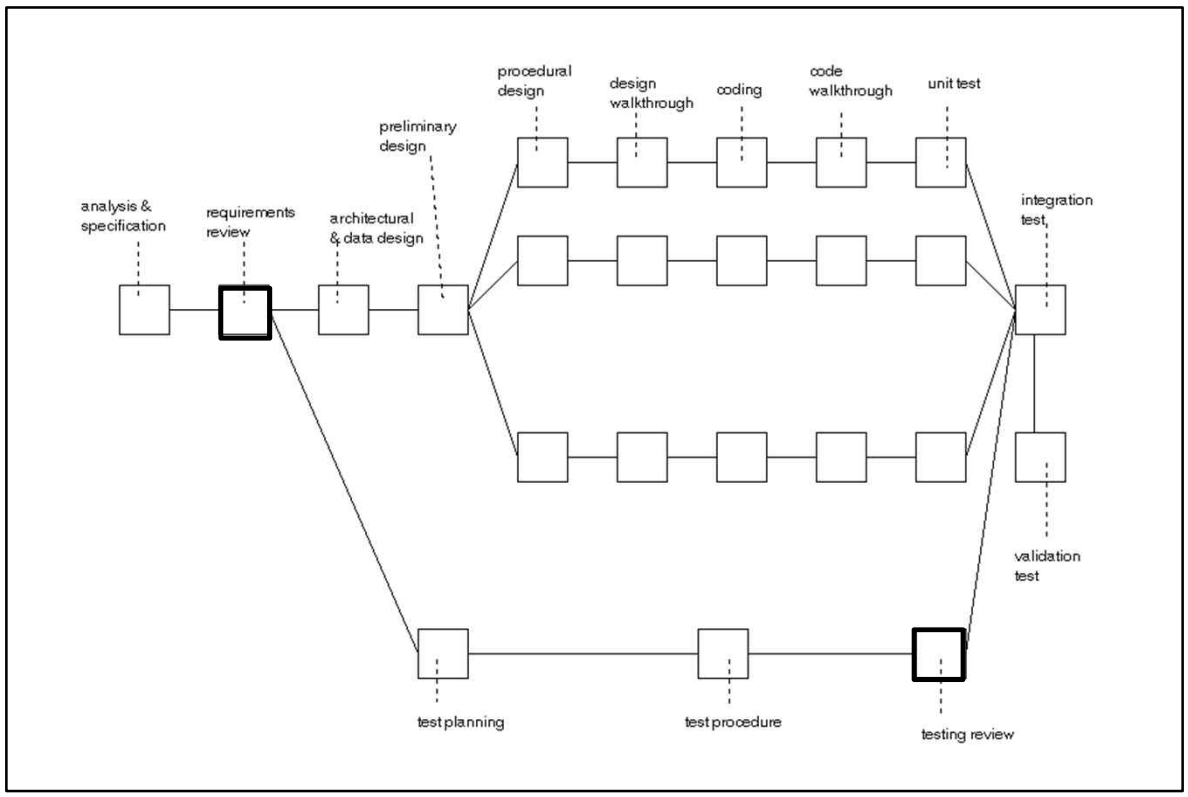
Effort Allocation (40-20-40 rule)

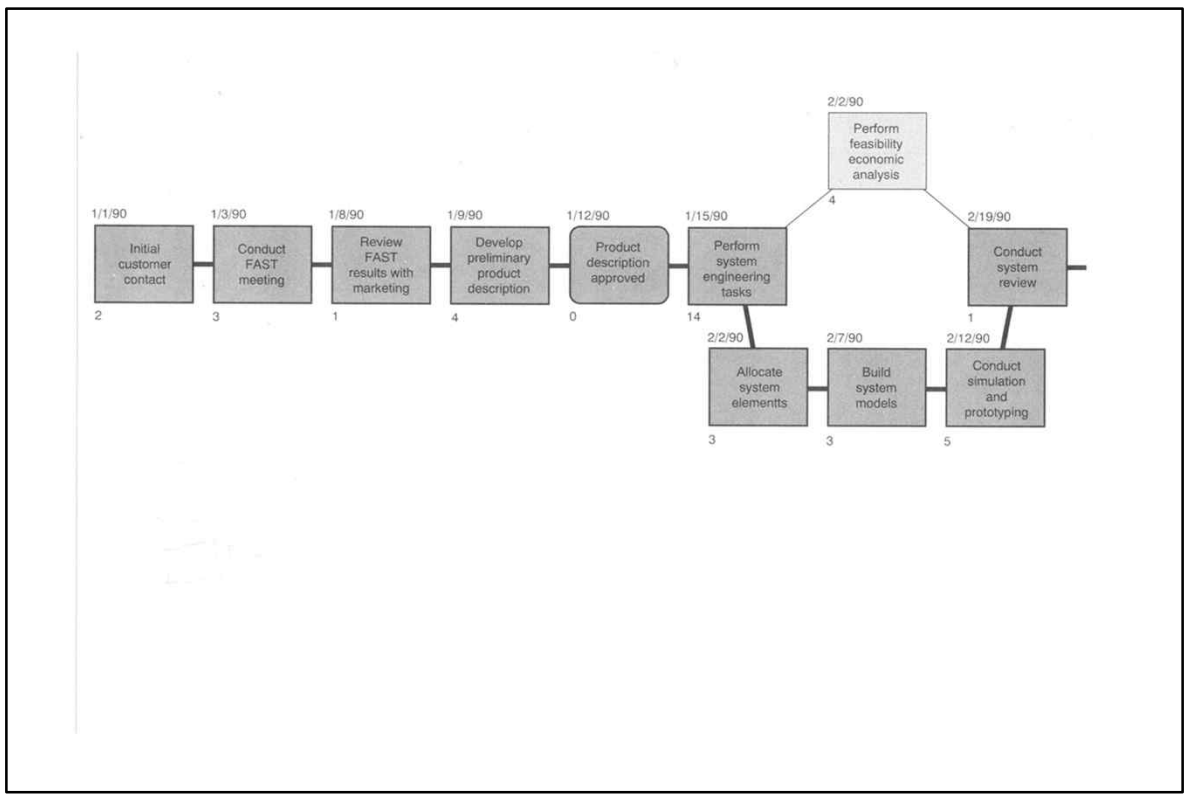


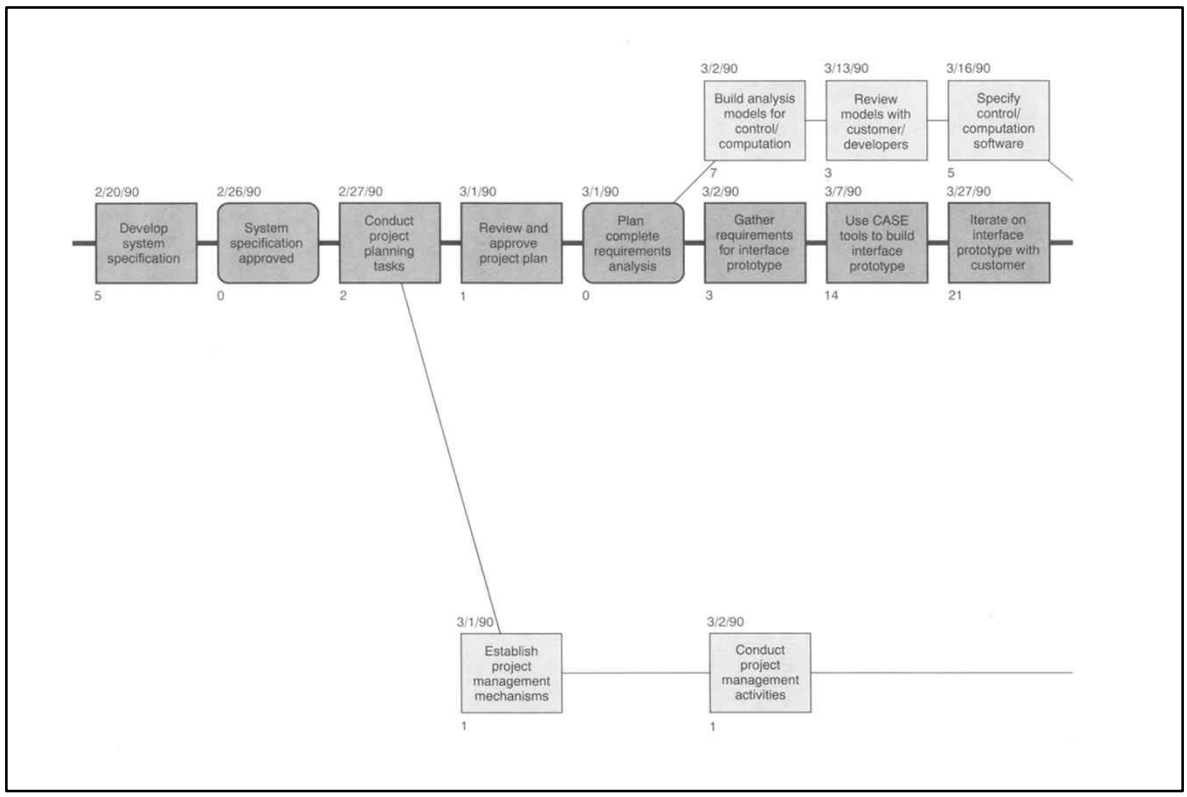
- "front end" activities
  - customer communication
  - analysis
  - design
  - review and modification
- construction activities
  - coding or code generation
- testing and installation
  - unit, integration
  - white-box, black-box
  - regression

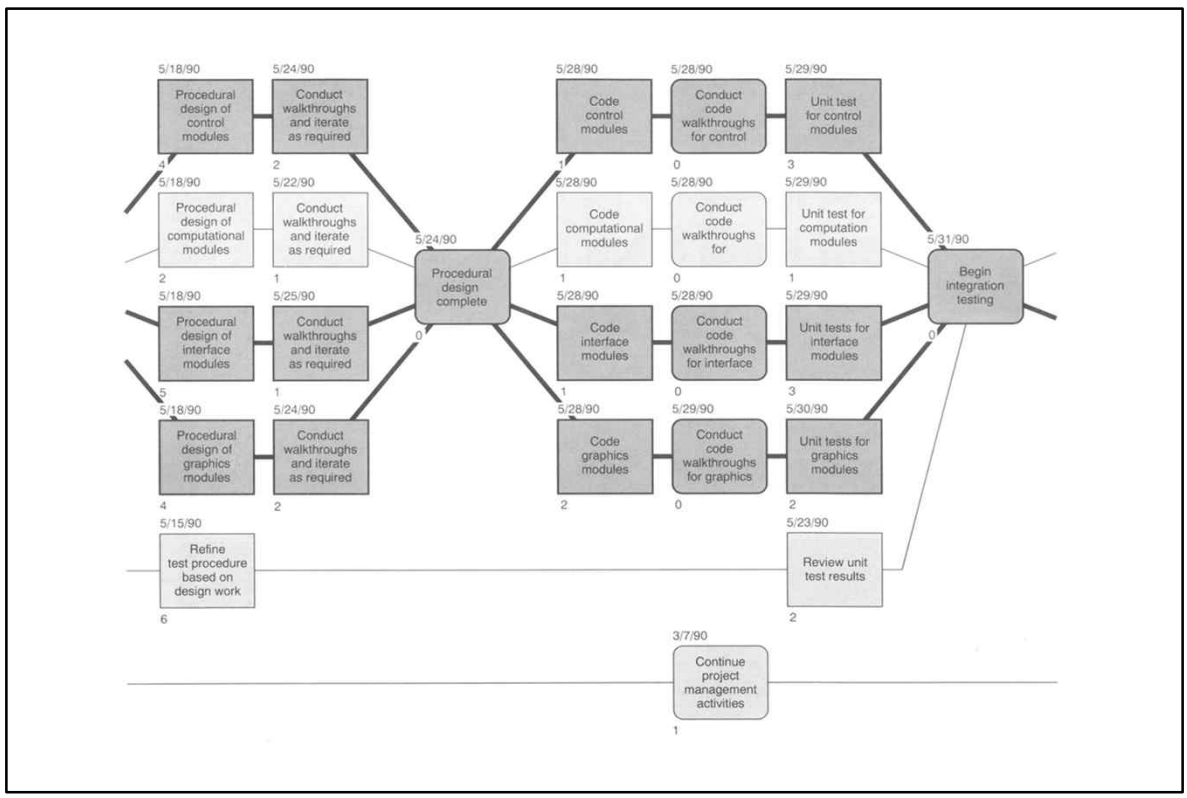
Define a Task Network

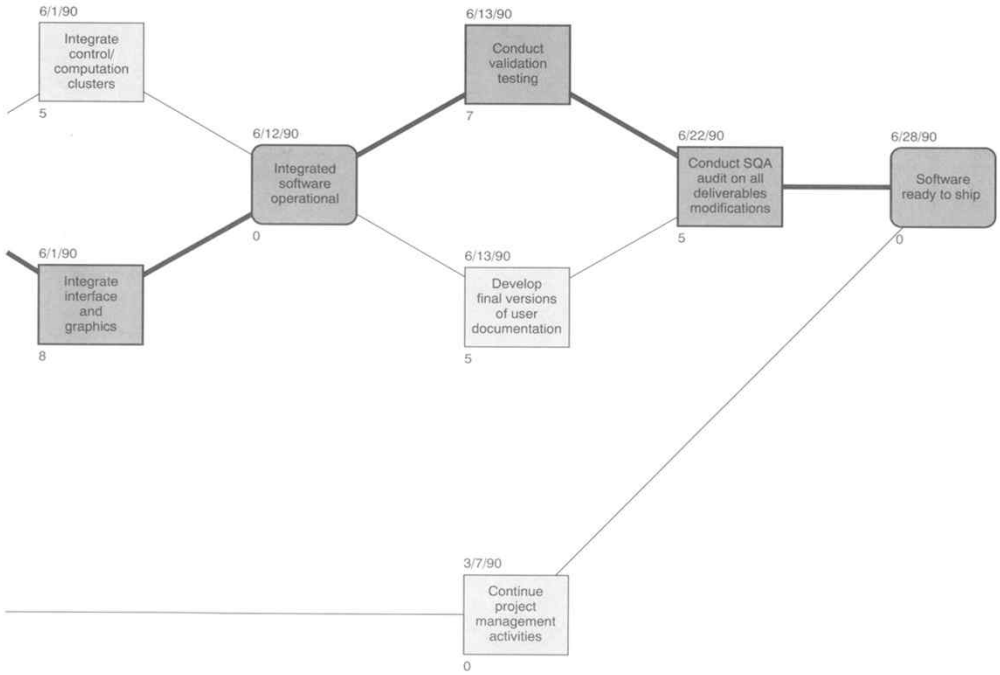



procedural design
design walkthrough
coding
code walkthrough
unit test
preliminary design
analysis & specification
requirements review
architectural & data design
integration test
validation test
test planning
test procedure
testing review


2/2/90
Perform feasibility economic analysis
4
1/1/90
Initial customer contact
2
1/3/90
Conduct FAST meeting
3
1/8/90
Review FAST results with marketing
1
1/9/90
Develop preliminary product description
4
1/12/90
Product description approved
0
1/15/90
Perform system engineering tasks
14
2/19/90
Conduct system review
1
2/2/90
Allocate system elementts
3
2/7/90
Build system models
3
2/12/90
Conduct simulation and prototyping
5

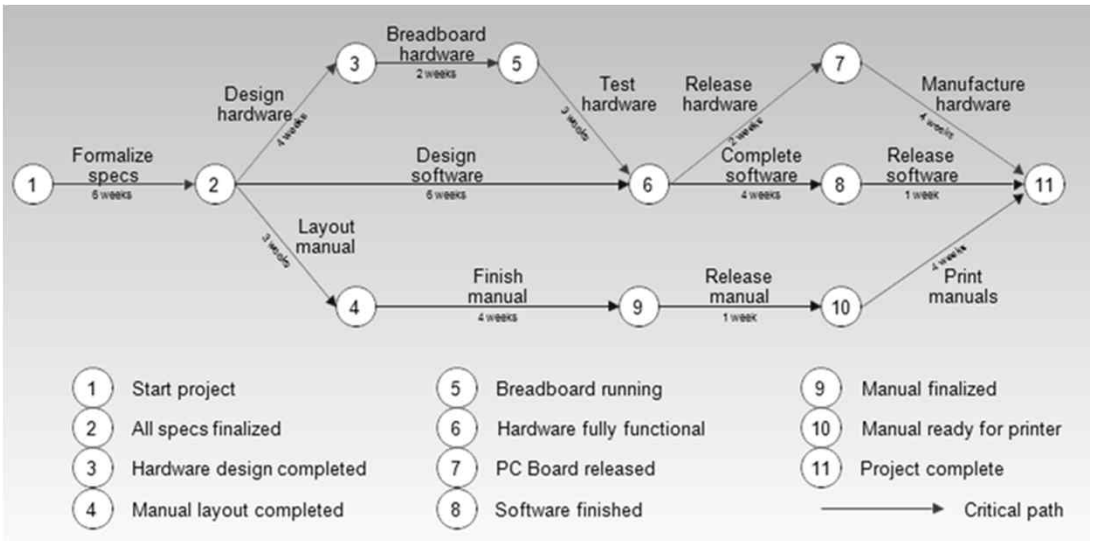

3/2/90
Build analysis models for control/ computation
7
3/13/90
Review models with customer/ developers
3
3/16/90
Specify control/ computation software
5
2/20/90
Develop system specification
5
2/26/90
System specification approved
0
2/27/90
Conduct project planning tasks
2
3/1/90
Review and approve project plan
1
3/1/90
Plan complete requirements analysis
0
3/2/90
Gather requirements for interface prototype
3
3/7/90
Use CASE tools to build interface prototype
14
3/27/90
Iterate on interface prototype with customer
21
3/1/90
Establish project management mechanisms
1
3/2/90
Conduct project management activities
1


5/18/90
Procedural design of control modules
4
5/24/90
Conduct walkthroughs and iterate as required
2
5/28/90
Code control modules
1
5/28/90
Conduct code walkthroughs for control
0
5/29/90
Unit test for control modules
3
5/18/90
Procedural design of computational modules
2
5/22/90
Conduct walkthroughs and iterate as required
1
5/28/90
Code computational modules
1
5/28/90
Conduct code walkthroughs for
0
5/29/90
Unit test for computation modules
1
5/24/90
Procedural design complete
0
5/31/90
Begin integration testing
0
5/18/90
Procedural design of interface modules
5
5/25/90
Conduct walkthroughs and iterate as required
1
5/28/90
Code interface modules
1
5/28/90
Conduct code walkthroughs for interface
0
5/29/90
Unit tests for interface modules
3
5/18/90
Procedural design of graphics modules
4
5/24/90
Conduct walkthroughs and iterate as required
2
5/28/90
Code graphics modules
2
5/29/90
Conduct code walkthroughs for graphics
0
5/30/90
Unit tests for graphics modules
2
5/15/90
Refine test procedure based on design work
6
5/23/90
Review unit test results
2
3/7/90
Continue project management activities
1

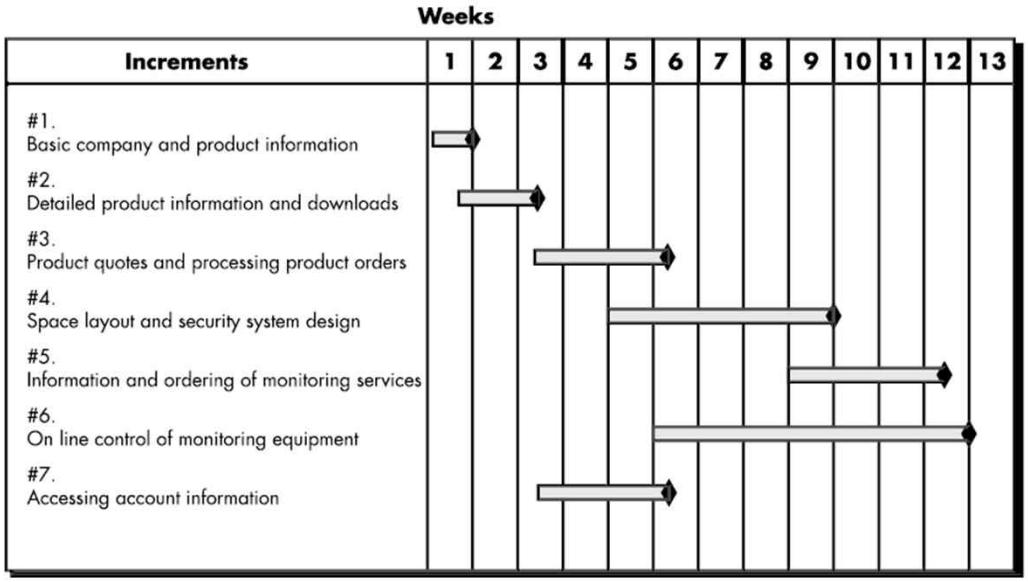


PERT / CPM

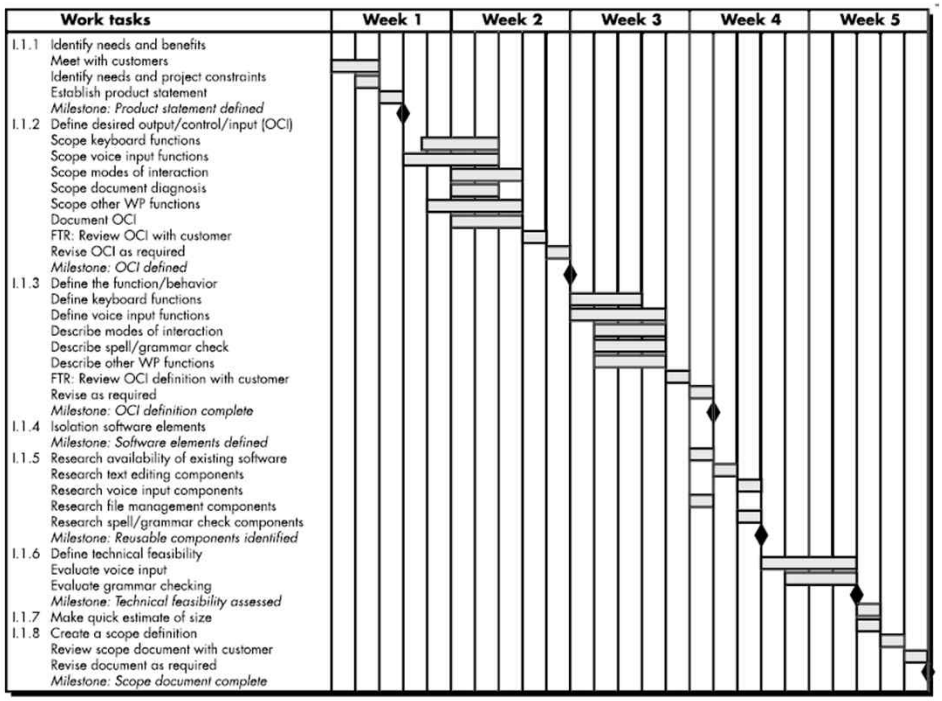
➔ Program Evaluation and Review Technique / Critical Path Method

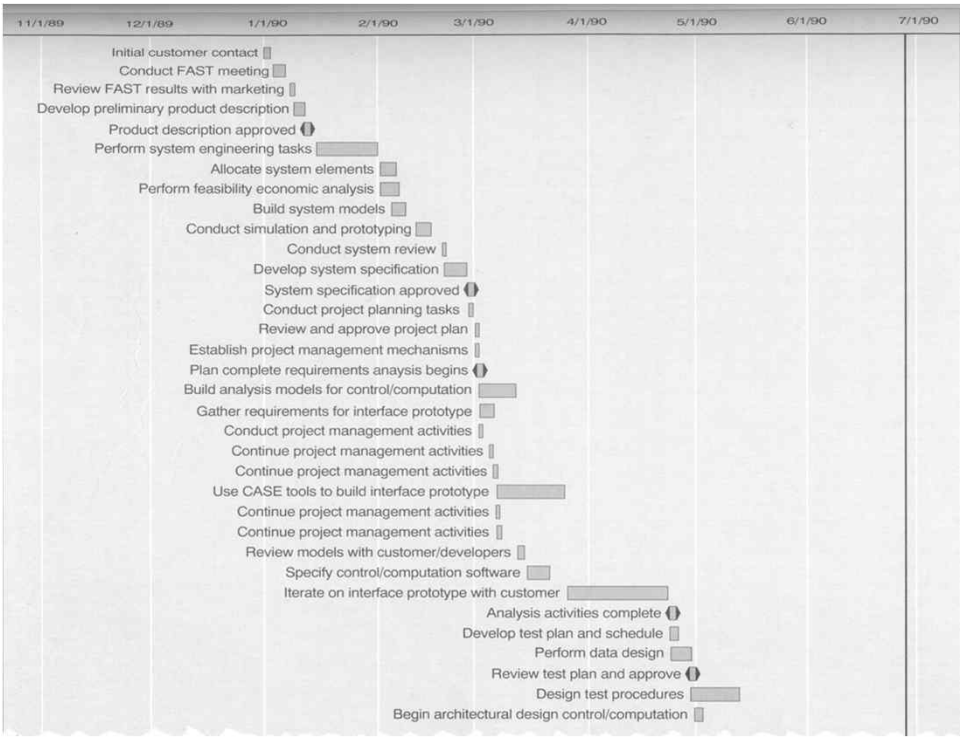


Timeline Chart



Timeline Chart





Project Table (Resource Table)

Work tasks	Planned start	Actual start	Planned complete	Actual complete	Assigned person	Effort allocated	Notes
I.1.1 Identify needs and benefits							Scoping will require more effort/time
Meet with customers	wk1, d1	wk1, d1	wk1, d2	wk1, d2	BLS	2 p-d	
Identify needs and project constraints	wk1, d2	wk1, d2	wk1, d2	wk1, d2	JPP	1 p-d	
Establish product statement	wk1, d3	wk1, d3	wk1, d3	wk1, d3	BLS/JPP	1 p-d	
Milestone: Product statement defined	wk1, d3	wk1, d3	wk1, d3	wk1, d3			
I.1.2 Define desired output/control/input (OCI)							
Scope keyboard functions	wk1, d4	wk1, d4	wk2, d2		BLS	1.5 p-d	
Scope voice input functions	wk1, d3	wk1, d3	wk2, d2		JPP	2 p-d	
Scope modes of interaction	wk2, d1		wk2, d3		MLL	1 p-d	
Scope document diagnostics	wk2, d1		wk2, d2		BLS	1.5 p-d	
Scope other WP functions	wk1, d4	wk1, d4	wk2, d3		JPP	2 p-d	
Document OCI	wk2, d1		wk2, d3		MLL	3 p-d	
FTR: Review OCI with customer	wk2, d3		wk2, d3		all	3 p-d	
Revise OCI as required	wk2, d4		wk2, d4		all	3 p-d	
Milestone: OCI defined	wk2, d5		wk2, d5				
I.1.3 Define the function/behavior							

Name	Earliest Start	Earliest Finish	Latest Start	Latest Finish	Type	Elapsed Time	Resource	Work-Days	Resource	Work-Days	Resource	Work-Days
Initial customer contact	1/1/90	1/2/90	1/1/90	1/2/90	Starting Task	2	Jennifer	1.50				
Conduct FAST meeting	1/3/90	1/5/90	1/3/90	1/5/90	Task	3	Jennifer	1	Matt	1	Mike	1
Review FAST results with marketing	1/8/90	1/8/90	1/8/90	1/8/90	Task	1	Jennifer	0.50				
Develop preliminary product description	1/9/90	1/12/90	1/9/90	1/12/90	Task	4	Jennifer	2	Matt	2		
Product description approved	1/12/90	1/12/90	1/15/90	1/15/90	Milestone	0						
Perform system engineering tasks	1/15/90	2/1/90	1/15/90	2/1/90	Task	18	Matt	6	Carolyn	3	Brian	3
Allocate system elements	2/2/90	2/6/90	2/2/90	2/6/90	Task	5	Matt	1	Carolyn	1	Jennifer	1
Perform feasibility economic analysis	2/2/90	2/7/90	2/13/90	2/16/90	Task	6	Mike	3				
Build system models	2/7/90	2/9/90	2/7/90	2/9/90	Task	3	Matt	3				
Conduct simulation and prototyping	2/12/90	2/16/90	2/12/90	2/16/90	Task	5	Matt	3	Carolyn	2		
Conduct system review	2/19/90	2/19/90	2/19/90	2/19/90	Task	1	Staff	1				
Develop system specification	2/20/90	2/26/90	2/20/90	2/26/90	Task	7	Matt	4	Jennifer	1		
System specification approved	2/26/90	2/26/90	2/27/90	2/27/90	Milestone	0						
Conduct project planning tasks	2/27/90	2/28/90	2/27/90	2/28/90	Task	2	Norm	1.50	Jennifer	1		
Review and approve project plan	3/1/90	3/1/90	3/1/90	3/1/90	Task	1	Staff	1				
Establish project management mechanisms	3/1/90	3/1/90	6/22/90	6/22/90	Task	1	Norm	1				
Plan complete requirements analysis begins	3/1/90	3/1/90	3/2/90	3/2/90	Milestone	0						
Build analysis models for control/computation	3/2/90	3/12/90	4/4/90	4/12/90	Task	11	Matt	5	Carolyn	5	Mike	5
Gather requirements for interface prototype	3/2/90	3/6/90	3/2/90	3/6/90	Task	5	Brian	3				
Conduct project management activities	3/2/90	3/2/90	6/25/90	6/25/90	Task	1	Staff	1				
Review models with customer/developers	3/13/90	3/15/90	4/13/90	4/17/90	Task	3	Staff	2				
Use CASE tools to build interface prototype	3/7/90	3/26/90	3/7/90	3/26/90	Task	20	Brian	10	Mike	2		
Specify control/computation software	3/16/90	3/22/90	4/18/90	4/24/90	Task	7	Matt	3	Carolyn	3		
Iterate on interface prototype with customer	3/27/90	4/24/90	3/27/90	4/24/90	Task	29	Brian	10				
Analysis activities complete	4/24/90	4/24/90	4/25/90	4/25/90	Milestone	0						
Develop test plan and schedule	4/25/90	4/27/90	5/4/90	5/8/90	Task	3	Andy	2				
Continue project management activities	3/5/90	3/5/90	6/26/90	6/26/90	Task	1	Staff	1				
Perform data design	4/25/90	5/1/90	4/25/90	5/1/90	Task	7	Matt	4	Carolyn	4		
Review test plan and approve	4/27/90	4/27/90	5/9/90	5/9/90	Milestone	0	Andy	0.50	Matt	0.50	Brian	0.50
Begin architectural design control/comp.	5/2/90	5/4/90	5/3/90	5/7/90	Task	3	Matt	1	Carolyn	1	Jennifer	1
Conduct architectural design interface	5/2/90	5/7/90	5/2/90	5/7/90	Task	6	Brian	4	Carolyn	2		
Design test procedures	4/30/90	5/11/90	5/9/90	5/22/90	Task	12	Andy	7	Matt	1	Jennifer	1
Integrate program architecture	5/8/90	5/9/90	5/8/90	5/9/90	Task	2	Matt	1	Brian	1		
Review test procedure and approve	5/11/90	5/11/90	5/23/90	5/23/90	Milestone	0						
Review and approve program architecture	5/10/90	5/14/90	5/10/90	5/14/90	Task	5	Staff	2				
Perform modifications based on review	5/15/90	5/17/90	5/15/90	5/17/90	Task	3	Matt	1	Brian	1		
Data and architectural design complete	5/17/90	5/17/90	5/18/90	5/18/90	Milestone	0						
Continue project management activities	3/6/90	3/6/90	6/27/90	6/27/90	Task	1	Staff	1				
Procedural design of control modules	5/18/90	5/23/90	5/18/90	5/23/90	Task	6	Matt	4				
Procedural design of computational modules	5/18/90	5/21/90	5/23/90	5/24/90	Task	4	Carolyn	2				
Procedural design of interface modules	5/18/90	5/24/90	5/18/90	5/24/90	Task	7	Brian	4				
Procedural design of graphics modules	5/18/90	5/23/90	5/18/90	5/23/90	Task	6	Jennifer	3				
Refine test procedure based on design work	5/14/90	5/18/90	5/23/90	5/29/90	Task	5	Andy	3				
Conduct walkthroughs and iterate as required	5/24/90	5/25/90	5/24/90	5/25/90	Task	2	Staff	1				
Conduct walkthroughs and iterate as required	5/22/90	5/22/90	5/25/90	5/25/90	Task	1	Staff	1				
Conduct walkthroughs and iterate as required	5/25/90	5/25/90	5/25/90	5/25/90	Task	1	Staff	1				

Schedule Tracking

- conduct periodic project status meetings in which each team member reports progress and problems.
- evaluate the results of all reviews conducted throughout the software engineering process.
- determine whether formal project milestones have been accomplished by the scheduled date.
- compare actual start-date to planned start-date for each project task listed in the resource table.
- meet informally with practitioners to obtain their subjective assessment of progress to date and problems on the horizon.
- use earned value analysis (EVA) to assess progress quantitatively.

Earned Value Analysis (EVA)
획득 가치 분석

- Earned value
  - is a measure of progress
  - enables us to assess the
    "percent of completeness" of a project
    using quantitative analysis
    rather than rely on a gut feeling

Computing Earned Value-I

- The *budgeted cost of work scheduled* (BCWS) is determined for each work task represented in the schedule.
  - $BCWS_i$ is the effort planned for work task *i.*
  - To determine progress at a given point along the project schedule, the value of BCWS is the sum of the $BCWS_i$ values for all work tasks that should have been completed by that point in time on the project schedule.
- The BCWS values for all work tasks are summed to derive the *budget at completion,* BAC. Hence,
- $BAC = \sum (BCWS_k)$ for all tasks *k*

Computing Earned Value-II

- Next, the value for *budgeted cost of work performed* (BCWP) is computed.
  - The value for BCWP is the sum of the BCWS values for all work tasks that have actually been completed by a point in time on the project schedule.
- "the distinction between the BCWS and the BCWP is that the former represents the budget of the activities that were planned to be completed and the latter represents the budget of the activities that actually were completed." [WIL99]
- Given values for BCWS, BAC, and BCWP, important progress indicators can be computed:
  - Schedule performance index, SPI = BCWP/BCWS
  - Schedule variance, SV = BCWP – BCWS
  - SPI is an indication of the efficiency with which the project is utilizing scheduled resources.

Computing Earned Value-III

- Percent scheduled for completion = BCWS/BAC
  - provides an indication of the percentage of work that should have been completed by time *t.*
- Percent complete = BCWP/BAC
  - provides a quantitative indication of the percent of completeness of the project at a given point in time, *t.*
- *Actual cost of work performed,* ACWP, is the sum of the effort actually expended on work tasks that have been completed by a point in time on the project schedule. It is then possible to compute
  - Cost performance index, CPI = BCWP/ACWP
  - Cost variance, CV = BCWP – ACWP

주요 용어

- **BCWS - Budgeted Cost of Work Scheduled**
  (계획된 비용)
- **ACWP - Actual Cost of Work Performed**
  (실제 사용된 비용)
- **BCWP - Budgeted Cost of Work Performed**
  (실제 진행된 일량에 해당하는 비용)

Derived Metrics

- SPI: Schedule Performance Index

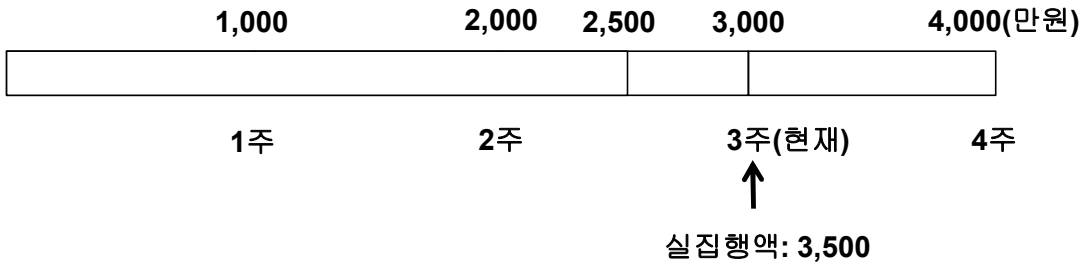
  SPI = BCWP/BCWS
  SPI < 1 means project is behind schedule

- CPI: Cost Performance Index

  CPI = BCWP/ACWP
  CPI < 1 means project is over budget

- CSI: Cost Schedule Index (CSI = CPI x SPI)
  The further CSI is from 1.0, the less likely project recovery becomes.

Example1

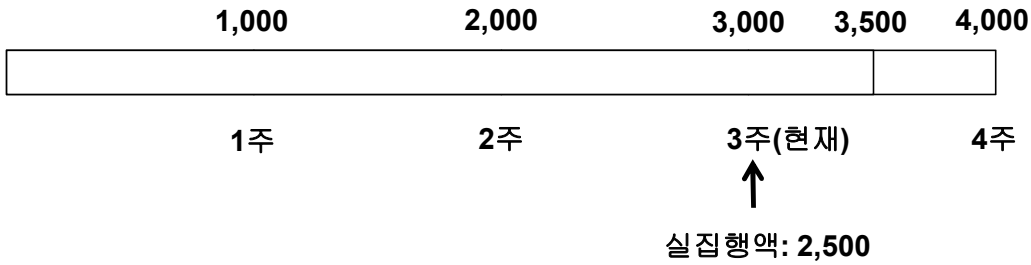


- BCWS = 3,000
- BCWP = 2,500 ➔ SPI = BCWP/BCWS = 2,500/3,000 = 0.83
- ACWP = 3,500 ➔ CPI = BCWP/ACWP = 2,500/3,500 = 0.71

➔ Percent scheduled completion = BCWS/BAC = 3,000/4,000 = 0.75

➔ Percent completion = BCWP/BAC = 2,500/4,000 = 0.625

Example2

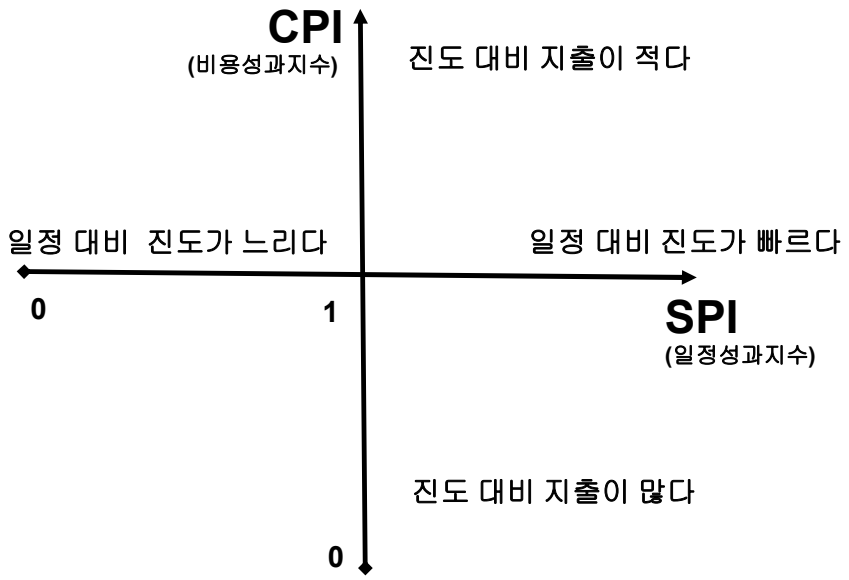


- BCWS = 3,000
- BCWP = 3,500 ➔ SPI = BCWP/BCWS = 3,500/3,000 = 1.17
- ACWP = 2,500 ➔ CPI = BCWP/ACWP = 3,500/2,500 = 1.4

➔ Percent scheduled completion = BCWS/BAC = 3,000/4,000 = 0.75

➔ Percent completion = BCWP/BAC = 3,500/4,000 = 0.875

SPI vs. CPI

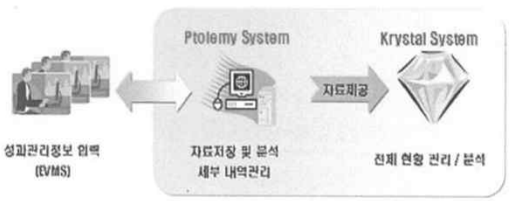


천리안 위성 개발 사례 (항공우주연구원)

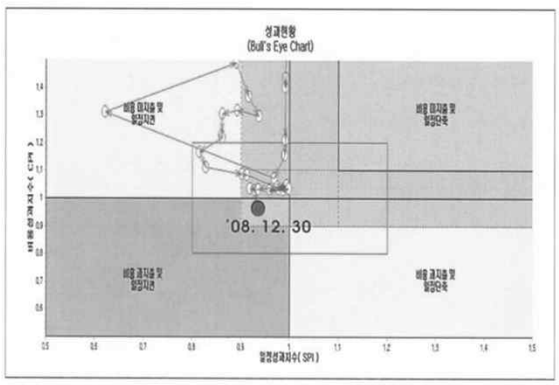
4.2.2 EVMS(Earned Value Management System) 를 활용한 통합성과 관리기술

➢ 수행성과
- EVMS 관리 S/W활용을 통한 효율적 비용/일정관리 수행
- 매월 비용 및 일정에 대한 성과지수 분석
- 비용/일정 성과지수 0.9 이상 목표 달성

➢ 달성내용
- 국내 국방사업 최초 Earned Value 개념을 도입하여 비용/일정 성과관리 수행
- 비용초과 / 일정지연 10% 수준 이내 통제관리



〈 EVMS 운용 체계 〉



< '08. 12월 현황 >

Homework #2: 일정관리

1. Scheduling

(1) PERT (Task Network)

(2) Timeline Chart

(3) Project Table