Mythical Man-Month

Ch 2. The Mythical Man-Month

> *챕터를 읽고..*
> csed434 project 는 이제껏 해봤던 소프트웨어 프로젝트 중 가장 plan 과 milestone 에 직접 신경을 쓰며 진행했던 project 이다. Scheduling 에 가장 많은 시간을 투자했고, test 는 생각보다는 적게 걸렸지만 unit test가 잘 된 덕이었던 것 같다. Manpower 가 해결하지 못하는 것이 있다는 부분은 해당 챕터를 접하면서 처음 생각해보게 되었는데, 실제로 진행되는 소프트웨어 프로젝트도 책에서 소개된 대로 각 task 사이에서 유기적인 communication 이 필요하고, 또 어느 부분은 sequential 하게 이루어질 테니 개발기한의 하한은 있을지라도 manpower로 개발 기간의 상한을 줄인다는 생각은 위험하다는 것을 여실히 알게 되었다.

- 많은 소프트웨어 프로젝트가 개발 기한을 맞추지 못하는 이유
  1. Optimism. 개발에 걸리는 시간을 estimate 하는 technique의 부족으로, '*all will go well*' 하는 생각.
  2. Effort(가해진 노력)과 progress(진행도)의 혼동.
  3. '*The courteous stubbornness of Antoine's chef*' 의 부족
  4. Schedule progress 가 제대로 고려되지 않음.
  5. Schedule 을 맞추지 못하는 상황에서 manpower 로 해결을 하려 함.
- Optimism
  : 개발자들은 대체로 낙관하고, 모든 것이 계획대로 되리라 생각한다. 하지만 실제에서 delay가 없는 경우는 극히 드물다.
- The Man-Month
  - Men과 Months, 즉 인력과 시간은 어떤 프로젝트가 인력간의 소통이 전혀 없이도 할 수 있는 작업들로 분배될 수 있는 경우에만 interchangable 하다. 모두 sequential 한

작업들로 분배될 경우, 인력에 관계 없이 작업 시간은 일정하다. 우리가 주목하는 software construction은 팀 내에서 complex interrelationships 가 요구되고, 아래와 같은 관계를 따른다.

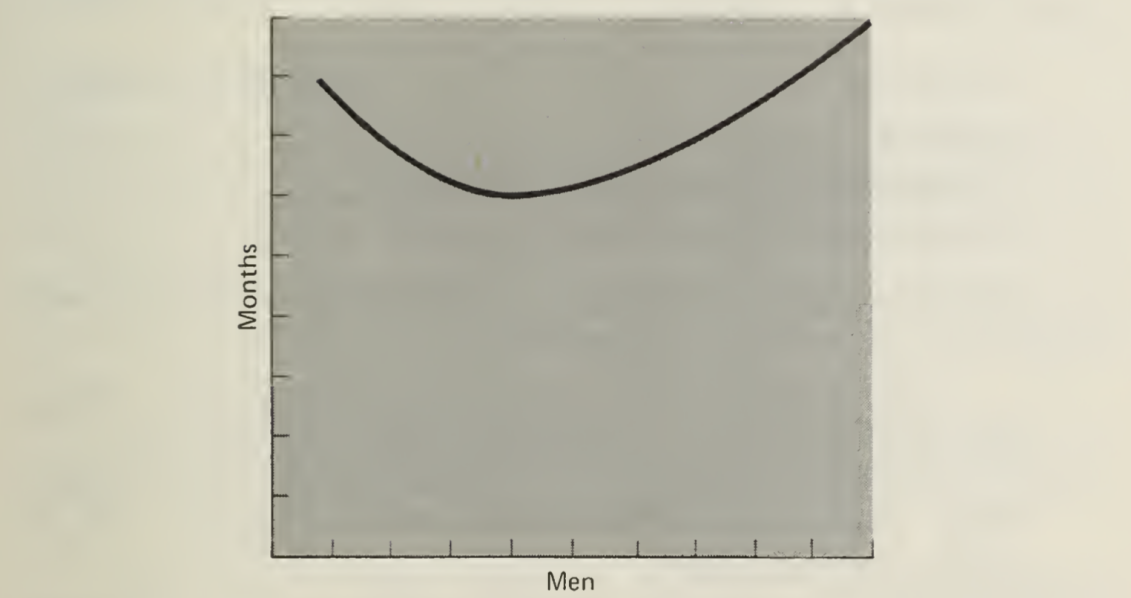

Fig. 2.4 Time versus number of workers—task with complex interrelationships

- System Tests (recommended - by author's experience)
  : 보통보다 planning 기간을 크게 잡았지만 부족할 수 있고, 가장 estimate 하기 쉬운 coding 이 가장 적은 시간을 차지하며, debugging 이 절반을 차지하지만 부족할 수 있다고 하고. test 에서 delay 가 가장 많이 발생할 수 있다고 설명한다.
  - $1/3$ planning
  - $1/6$ coding
  - $1/4$ component test and early system test
  - $1/4$ system test, all components in hand
- Gutless Estimating
  : 고객의 의견에 따라 schedule이 조절가능한 것으로 보이지만, 실제 완성을 따졌을 때는 그렇지 않다. 터무니없이 적은 시간만에 오믈렛을 만들라는 주문을 한 고객은 더 기다리거나, 덜 익을 오믈렛을 먹을 수밖에 없다. 요리사가 시간에 맞추기 위해 화력을 올린다면 오믈렛의 일부분은 제대로 구워질 지 몰라도, 어떤 부분은 타게 될 수밖에 없다..

: 요리처럼, 소프트웨어 프로젝트는 제작 기간을 산정할 눈에 보이는 지표를 발견하기 힘들어서 그런지, 위 요리의 예시와 같이 다른 분야에 비해 잘못된 일정을 잡는 경우가 많다. 그렇기에 더더욱, productivity figures, bug-incidence figures, estimating rules 등 최대한 정량적으로 파악할 수 있는 지표를 개발하고 알려 기간의 estimate 를 잘 할 수 있도록 해야 하고, project manager 는 그나마 정확한 estimate가 되지 않을 때는 optimism에 기반하지 않고 기간을 추정할 수 있도록 주의해야 한다.