**The Mythical Man-Month**

20220019 안재영

**1장: "The Tar Pit"**

* 대규모 소프트웨어 개발은 마치 "타르 구덩이" 같다. 타르 속에 갇힌 동물들은 어느 쪽발이라도 움직이는 데 별 지장이 없지만, 동시적인 요인들에 의해 움직임은 점차 둔해진다. 대규모 프로그래밍도 마찬가지로, 문제의 까다로움에 모두가 놀라면서도 그 본질의 분별은 어려워진다.
* 프로그래밍의 즐거움 : 무언가를 만드는 데서 오는 순전한 기쁨, 다른 이들에게 쓸모 있는 사물을 만드는 데서 오는 기쁨, 서로 맞물리는 복잡한 법칙들의 조화를 바라보는 기쁨, 지속적인 배움에서 오는 기쁨, 유연하고 다루기 쉬운 표현 수단을 사용한 작업이 주는 기쁨은 프로그래밍의 즐거움을 만드는 요소들이다.
* 프로그래밍의 고달픔 : 반면, 완벽함에 대한 요구, 업무에 대한 자주적 결정력의 부족, 불가피한 타인의 프로그램에의 의존, 사소한 버그 수정이라는 “일”, 오히려 개발 막바지에 더 발견되는 버그들, 완성품이 한물간 것으로 치부되게 만드는 빠른 발전 속도 등은 프로그래밍을 힘들게 만들기도 한다.
* 많은 이들을 허우적거리게 만드는 타르 구덩이임과 동시에 나름의 즐거움과 고달픔을 담고 있는 창조적 활동, 이것이 바로 프로그래밍이다.

**2장: "The Mythical Man-Month" (사람과 시간이 비례하지 않음)**

* 부족한 시간은 소프트웨어 프로젝트를 망가뜨리는 최다 요인이다.
  + 우리의 추정 능력은 형편없다. 모든 일이 잘 될거라는 가정을 하고는 한다.
  + 추정에 있어 투입 공수와 작업 진척도를 혼동해 인원, 기간이 상호 교환 가능하다고 여기고는 한다.
  + 자기가 내린 추정이 스스로도 불확실하기에, 정중한 고집이 부족하다.
  + 일정 진척도가 제대로 모니터링되지 않는다.
  + 일정이 어긋날 경우의 일반적인 대응이 인력 증원으로, 이는 사실 사태를 악화시킬 뿐이다.
* 낙관적인 프로그래머
  + 낙관론자들은 “이번에는 돌아갈 겁니다.”, “방금 마지막 버그를 수정했습니다.”라 하고는 한다.
  + 가장 잘못된 가정은 “모든 작업이 예정 내로 완료될 것”이라는 것이다.
  + 프로그래밍의 재료인 생각은 다른 창작의 목재, 물감처럼 물리적 한계를 주지 않고, 이에 프로그래머들 사이에 낙관주의가 만연하게 된다.
* 맨먼스
  + 프로그래머들은 추정 및 일정 관리에서 투입된 노력을 맨먼스라는 단위로 계산하는 실수를 저지른다.
  + 작업 진척도는 투입된 사람의 수와 달수의 곱에 비례하지 않는다.
  + 작업에 순서가 있어 나누기 힘들다면 쏟아붓는 노력은 도움이 될 수 없다.
  + 분할은 가능하나 하위 작업의 처리에 소통이 필요하다면, 소통을 위한 노력은 사람의 수가 증가할수록 n(n-1)/2배가 되고는 한다.
  + 사람의 투입은 때로는 일정을 단축시키기는커녕 늘어지게 만들 수 있다.
* 시스템 테스트에는 일정에 다음과 같은 일정 분배는 꽤 효율적이다.
  + 계획 수립 1/3 코딩 1/6 구성 요소 및 초기 시스템 1/4 준비 후 1/4
  + 계획 수립에는 많은 시간을 배정하며, 추정하기 쉬운 코딩의 부분에는 비교적 적은 일정을 배정한다.
  + 시스템 테스트는 처음부터 오랜 시간이 계획되지 않지만, 불가피하게 정작 테스트가 시작되었을 때 더 많은 시간을 잡아먹으며 모두를 불안에 빠트리고는 한다. 따라서 초기 일정 수립 시에 테스트에 충분한 시간을 할애하는 것은 중요하다.
* 소심한 추정
  + 일을 맡긴 고객이 결과를 재촉할 때, 일이 예정보다 앞당겨 끝날 순 있지만 항상 올바른 결과물이 더 빠른 시간에 완성되는 것은 아니다.
  + 고객이 바라는 날짜에 맞추려고 잘못된 일정을 세우는 일은 생각보다 프로그래머들이 자주 하는 실수 중 하나이다. 고객이 바라는 날짜가 정량적이지 못한 그들의 예감에 의한 희망 사항이라면, 프로그래머들은 생산성 수치, 버그 발생률 등의 근거를 바탕으로 당당하게 스스로의 추정치를 방어해야 한다.
* 일정 붕괴의 악순환
  + “늦어진 소프트웨어 프로젝트에 인력을 추가로 투입하면 더 늦어지게 된다”는 브룩스의 법칙으로 맨먼스 미신을 걷어낼 수 있다.
  + 프로젝트의 소요 기간은 순서가 있는 내부 요소에 좌우되며, 필요한 인원수는 독립된 하위 작업의 수에 좌우된다.
  + 위의 두 정보를 파악한 관리자는 일정을 짤 때 더 적은 수의 사람과 더 긴 기간은 고려할 수 있으나, 더 많은 사람과 더 짧은 기간에 대해서는 실행 가능한 일정을 만들 수 없다.
  + 부족한 시간 탓에 망가진 프로젝트는 잘못된 프로젝트의 원인 중 최다를 차지함을 잊지 말아야 한다.