

Textbook Summary

The Mythical Man-Month by Frederick P. Brooks Chapter 2

20180916 임경빈 컴퓨터공학과, kbini@postech.ac.kr

Chapter 2 : The Mythical Man-Month

시간 부족으로 인해 소프트웨어 프로젝트가 실패하는 이유는 여러가지가 있다. 첫째는 낙관주의에 기초한 잘못된 추정, 둘째는 인원과 기간이 상호 교환 가능하다는 잘못된 가정, 셋째는 확신이 들지 않는 잘못된 추정, 넷째는 일정 진척도의 모니터링 부재, 다섯째는 신중하지 않은 인력 추가 투입으로 인해서이다.

대체로 소프트웨어 엔지니어들은 낙관적인 경향을 보인다. 그 이유는 컴퓨터라는 분야가 생겨난 지 오래 지나지 않아서 일수도 있고, 최종 목표에만 집중하는 사람들만 남았기 때문일 수도 있다. 도로시 세이어즈에 따르면 인간은 창조 활동을 할 때 마음속에 이미 만들어진 완성물을 현실의 수단을 통해 구현한다. 창작 활동에서 대부분 이런 구현은 재료의 물리적 특성으로 인해 다루기 어렵지만 컴퓨터 프로그래밍은 그 특성상 구현하는 행위 자체가 매우 쉬우므로 낙관주의가 만연할 수 있는 것이다.

일반적으로 노동력이 투입되는 프로젝트에서는 '맨먼스 Man-month'라는 단위를 통해 투입된 자원을 계산한다. 이는 사람과 일정이 서로 교환 가능하다는 전제가 깔려있는데, 이를 컴퓨터 프로그래밍에 적용하는 것은 적절하지 않다. 소프트웨어 개발 작업은 분할하기 어려운 경우가 많으며 서로 의사소통 하는데 필요한 시간과 노력 또한 고려되어야 하기 때문이다.

대부분 소프트웨어 프로젝트를 진행하면 시스템 테스트에 많은 시간을 할애하지 않지만 예기치 못한 일들로 인해 실제로 나중에는 거의 절반에 가까운 시간을 테스트에 쓰게 된다. 만약 테스트에 충분한 시간을 배정하지 않는다면 프로젝트 말미에 일정이 늦어지기 때문에 프로젝트가 실패하게 될 가능성이 커진다.

만약 중요한 소프트웨어 프로젝트가 일정에 뒤처지고 있을 경우 인력을 추가로 더 투입하여 남은 기간 동안 생산량을 늘리려고 시도할 수 있다. 하지만 이는 일을 더 그르칠 수 있다. 만약 프로젝트에 참여하지 않았던 인력을 추가할 경우 그 인력을 교육시키기 위해, 또 남은 일정을 다시 재분배하기 위해 추가적인 시간이 소모된다. 브룩스의 법칙은

“늦어진 소프트웨어 프로젝트에 인력을 추가로 투입하면 더 늦어지게 된다”는 것이다. 만약 이런 상황에 처하게 된다면 맨먼스의 미신을 바로 걷어내고 더 긴 시간을 잡는 것이 최선일 것이다.