

Composable IT

“netj와 함께 고민해보는 우리 IT의 미래”

신재호 <netj@sparcs.org>

언제: 2007-03-23 (금)

어디서: SPARCS, KAIST

여러분의 요즘 관심사?

- Vista? Leopard? Beryl?
- Office 2007? ODF? Ribbon?
- Linux? Fedora Core 6? Ubuntu 7.04? Debian etch? openSUSE?
- OpenSolaris? ZFS? dtrace?
- Virtualization? Grid?
- IPv6? P2P? RFID?
- D? Ruby? Perl6? Java 5? .NET?
- XML 1.1? XSLT 2.0? XQuery 1.0?
- IE7? HTML5? AJAX? JSON?
- F/OSS? GPLv3? Software Patents?
- XP? TDD?
- RoR? Django? REST? SOA? SaaS?
- UCC? Springnote? me2day? OpenID? Swivel?
- 시즌2? 싸이2?
- LKIN? ARA? ska?
- Anti-Spam? OpenWeb? DRM?
- clix? iPhone? MacBook? VAIO G?
- NDSL? Wii? PS3?

netj?

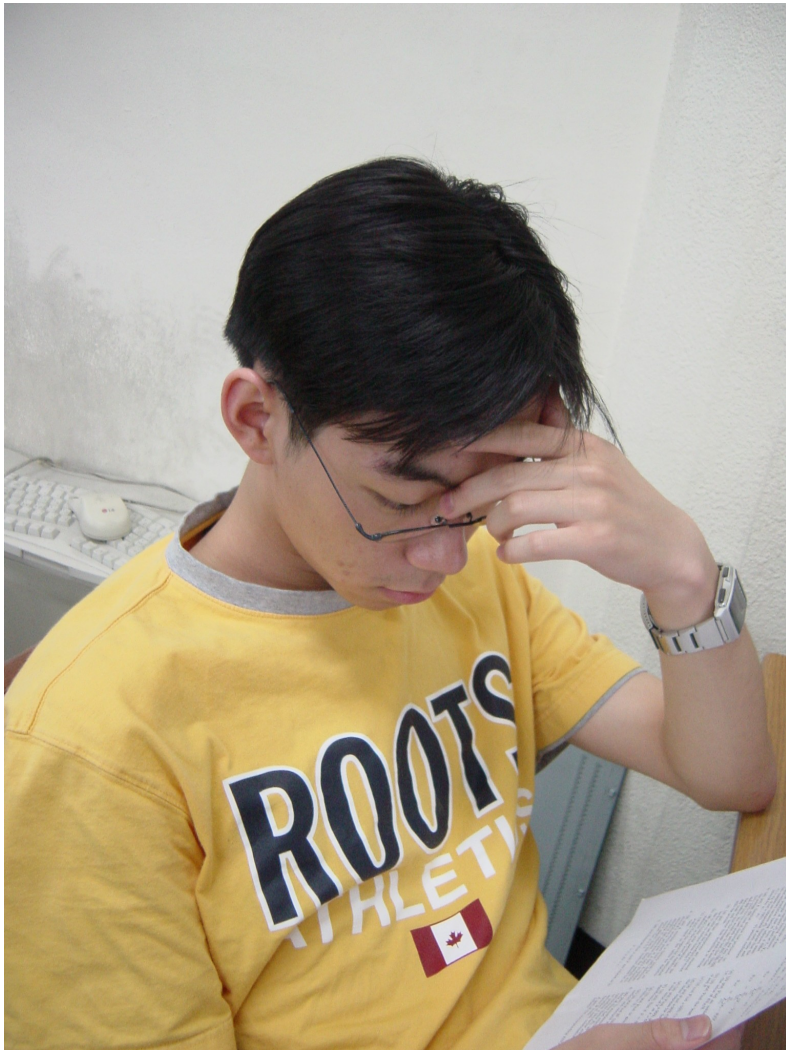


KAIST

한국과학기술원
Korea Advanced Institute of Science and Technology

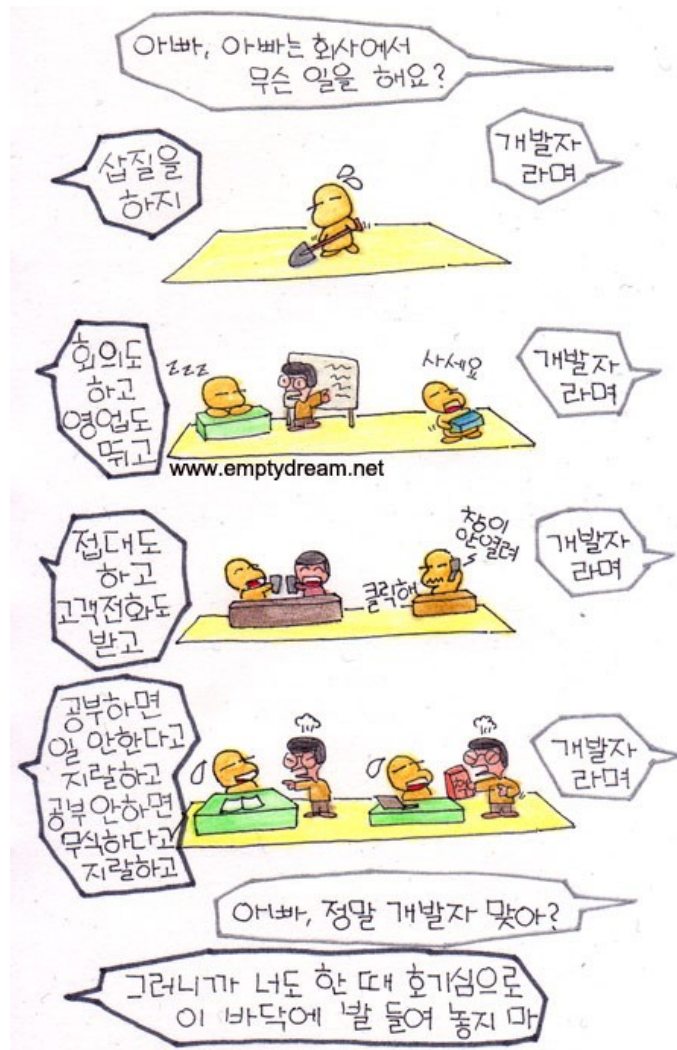


netj의 요즘 관심사



- IT의 방향
 - 우리 IT가 왜
요 모양 요 꼴이 됐나
하는 걱정

개나 소나 개발



- 낮은 진입장벽?
 - 그만큼 개판
- 전문가로 성장?
 - no career path

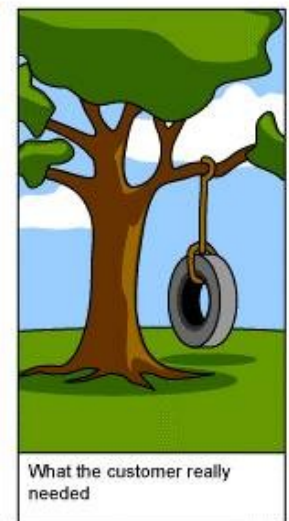
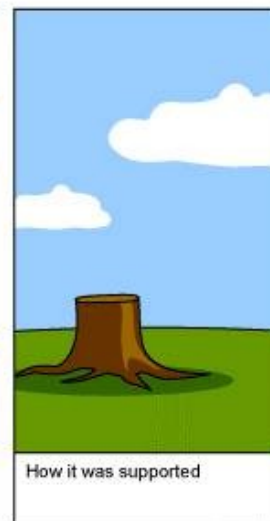
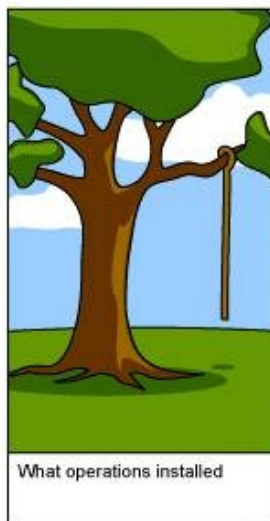
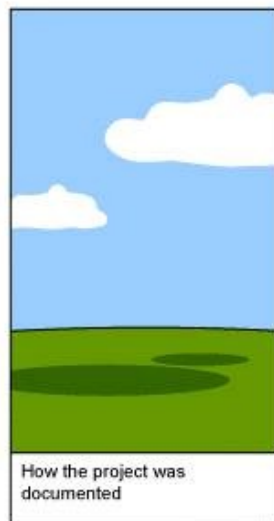
코딩 기피



- 너도나도 관리자
- 코딩은 누가 하나

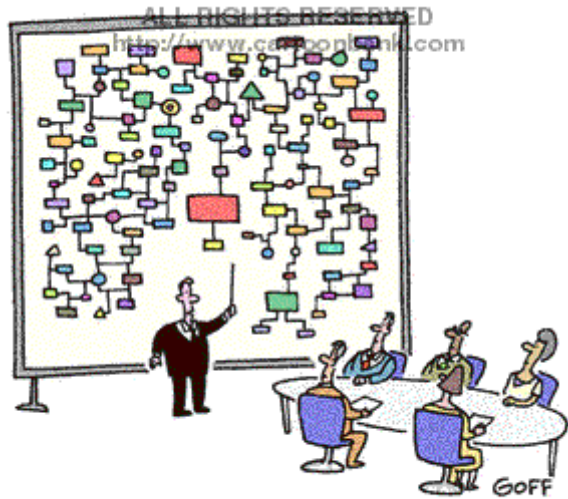
현실과 바람

- 크나큰 괴리



노동의 불균형

- 개발자 vs. 사용자
 - 사용자는 클릭만 해야 한다!
 - 개발자가 모든 일을 대신 해라



"And that's why we need a computer."



"On the Internet, nobody knows you're a dog."

막다른 길

- 반성이 필요해!!!
 - 정말로 이 길 밖에는 없었을까
 - 우리가 뭘 잘못된 걸까



도구에 대한 오해

- 소프트웨어란 무엇인가
- 정보기술이란 무엇인가
- 인터넷이란 무엇인가
- 컴퓨터란 무엇인가



도구는 ~~문제를 해결해주는~~ 답이 아니라
문제를 해결해가는 **과정**이다

두 갈래

- Vim
- Unix
- Web
- set of tools
- do-one-thing-best
- open
- extensible piece
- collaboration
- ubiquitous
- Emacs
- Windows
- Desktop
- huge IDE
- just-do-everything
- closed
- 1-click solution
- collection
- centralized

IT가 가야 할 길

- CLI, drag&drop
- learning curve
- 매일 쓰는 컴퓨터
 - 익히면 익힐수록 더 잘 쓸 수 있어야 한다
 - Long Term Productivity!
- 전통적 1-click GUI
- learning gap
- 공항의 정보 단말기
 - 배울 필요가 없어야 한다
 - 쓰기 쉬우면 그만이다

Composable IT

- 성장 가능한 IT
 - 만든 것 끼리 어울릴 수 있게
 - 조합하고 차곡차곡 쌓을 수 있게
- 지속 가능한 IT
 - 필요한 것을 쉽게 만들어 쓸 수 있게
 - 언제나 더 어려운 문제를 풀어나갈 수 있게

IT의 비밀

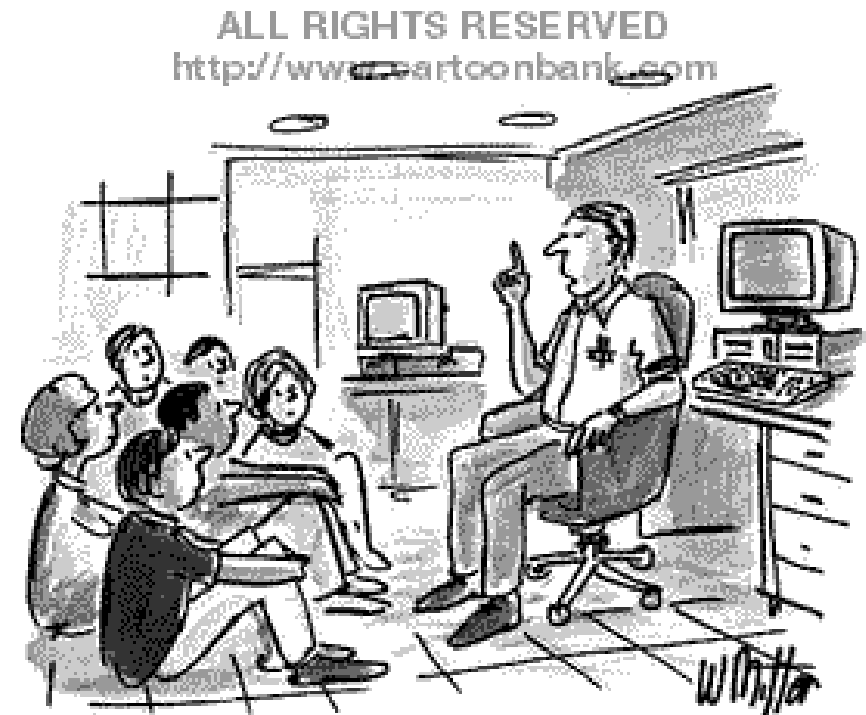
- 모든 개인 능력 차이는 **editor**에서 비롯
 - 자료 가공/처리 능력
 - 프로그래밍 능력
 - 설계 능력
 - 기획 능력
 - ...
- 좀 더 정확히는, **기계 다루는 능력**

너희가 컴퓨터를 쓸 줄 아느냐

1. Vim, Microsoft Excel – 자료 잘 다루기
2. bash, Windows 탐색기 – 파일 잘 다루기
3. Mozilla, IE, Google – 정보 잘 다루기
4. mutt – 메일 잘 다루기
5. POSIX tools (coreutils, ...), ssh, rsync, ...

학교에서 잘 안 가르쳐주는 IT

- bash, perl, python
- gcc, g++, make
- cvs, svn, darcs
- TCP/IP, DNS, HTTP, SMTP, IMAP, POP, FTP, IRC, NNTP
- HTML, CSS, Javascript, CGI
- XML, XSLT, XQuery
- PKI, PGP
- RFC, IEEE, ISO, W3C



*"Those who do not learn from the future
are destined to make mistakes in it."*

읽어보기

- What's GNU

<http://www.linuxjournal.com/article/2762>

- Seven habits of effective text editing

<http://www.moolenaar.net/habits.html>

정리하며

오늘은 더 나은 내일을 위한 날!

질문+답은 뒤풀이 가서... :^)