Open Source 프로젝트 추진방안

이 민석

국민대학교 소프트웨어학부





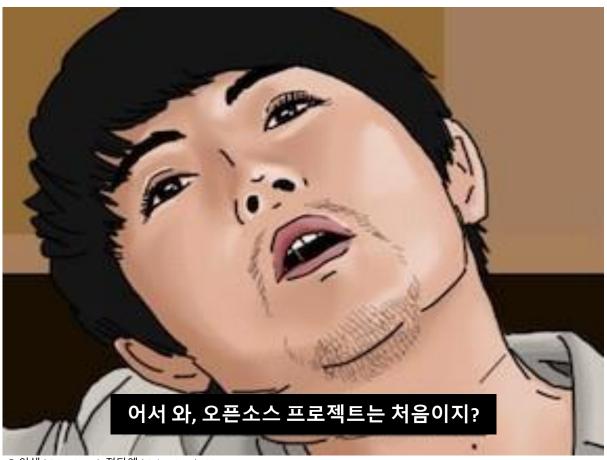
오픈소스 소프트웨어 20년, DMZ 선언문

(오픈소스와 평화는 공짜가 아니다)

소프트웨어에 오픈소스라는 접두사가 붙은지 20년이 되었다. 이제 오픈소스는 너무도 자연스러운 문화 현상이자 방법론이 되었다. 모든 기술적 혁신은 오픈소스에서 시작되며, 혁신의 지속가능성도 오픈소스에 의해서 담보된다. 한반도에서도 오픈소스 소프트웨어라는 거대한 흐름을 막을 수는 없었다. 수 많은 개발자들과 커뮤니티들이 즐겁게 그리고 자랑스럽게 오픈소스에 기여하고 있다. 아직도 이 시대적 사명을 거부하는 무리들이 있으나 점차 세력을 잃어가고 있다. 머지않아, 이 땅에서도 오픈소스라는 기차에 모든 소프트웨어가 올라타게 될 것이다.

우리는 아직도 아물지 않은 한반도의 비극적 역사가 어설프게 봉합된 이곳 DMZ에서 오픈소스 소프트웨어 20주년을 축하한다. 우리는 이제 남한과 북한의 소프트웨어 개발자들이 오픈소스라는 공간에서 만나 교류하게 될 것을 확신한다. 우리는 새로운 남북협력시대를 맞아 세계인이 같이 공감하며 참여하는 거대한 오픈소스의 물줄기가 이곳에서 다시 시작되기를 기원한다.





© 이샘 (<u>@0713sam</u>), 정다예 (<u>@dayejung</u>). CC BY.

저는 인공지능이 싫어요

이세돌 vs. AlphaGo @2016





그 전, 2015년 한 해 동안

- 구글은 '텐서플로'
- 마이크로소프트도 'DMLT'
- IBM은 '시스템ML'
- 삼성전자는 '벨레스(Veles)'
- 페이스북은 '토치(Torch)'

지금은….

- Apache Spark, MLlib
- **Apache Singa**
- Caffe
- Microsoft Azure ML Studio
- AML(Amazon Machine Learning)
- Microsoft CNT
- **Brainstorm**
- mlpack 2
- Marvin
- Neon

가 오픈소스로 공개되어 있음

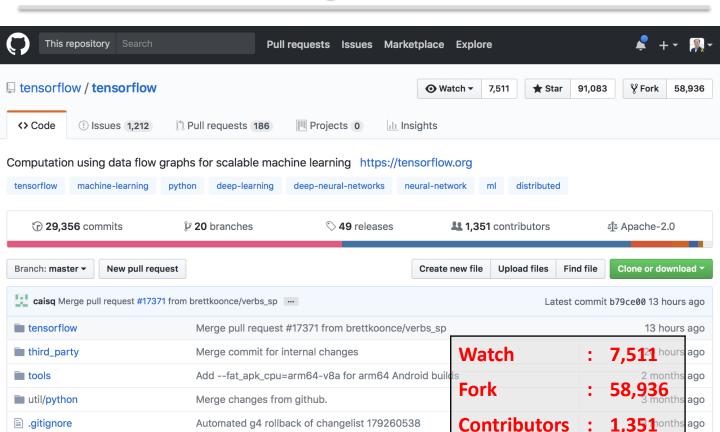


구글 tensorflow @ github.com

.gitignore

E ADODTEDO ----I

ACKNOWLEDGMENTS



1,351 nonths ago

2 years ago

Automated g4 rollback of changelist 179260538

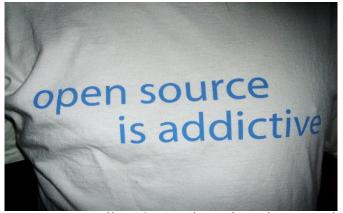
TensorFlow: Improve performance of Alexnet

to Annual Charles and a second



Open Source:

사용자의 요구를 가장 효과적으로 수용하고, 품질 높은 개발을 할 수 있는 잘 증명된 개발 패러다임



https://www.flickr.com/photos/mager/2299296256/

Open Source:

모여서 재미있고 참여와 공유, 기여로 발전하는 커뮤니티 문화 현상







1985.4

FREE SOFTWARE
FOUNDATION





Linux





Open Source Projects







http://en.wikipedia.org/wiki/List_of_free_and_o pen source software packages































elastic















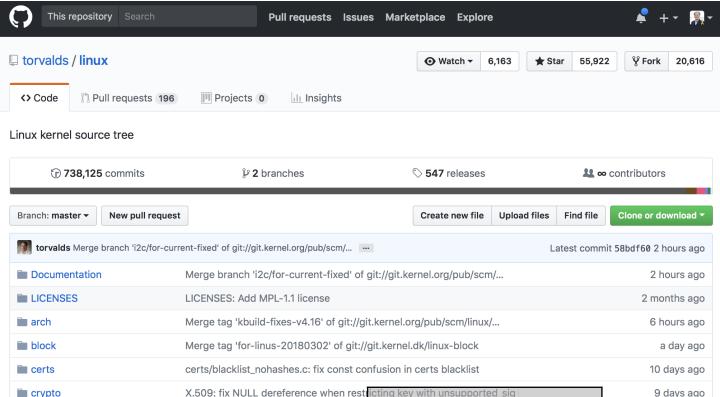












Merge branch 'i2c/for-current-fixed' of git://git.kernel.org/pub/scm/...

Merge tag 'ceph-for-4.16-rc4' of git://gthub.com/ceph/ceph-client

Merge branch 'libnvdimm-fixes' of git://git.kernel.org/pub/scm/linux/.

kbuild: remove all dummy assignments to 2011

6,163

20.616

 ∞

2 hours ago

a day ago

2 hours ago

4 months ago

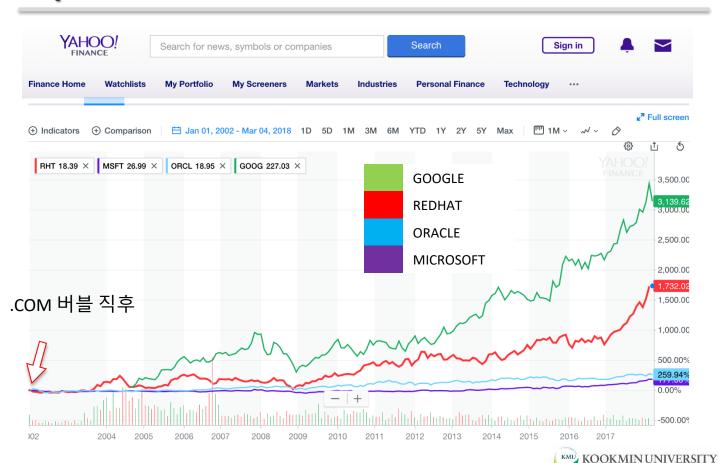


drivers

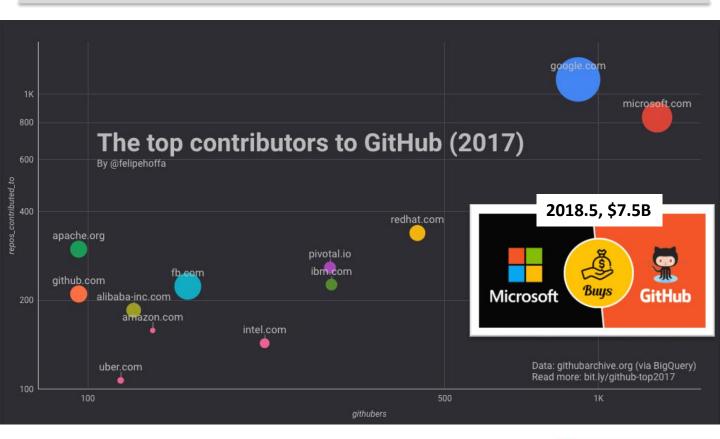
in firmware

fs fs

Open Source BIZ의 성장



Top contributors to GitHub





Open Source Software, now

- 78% of companies are run based on Open Source Software (2015)
 - Less than 3% of companies do not use Open Source
- When companies need Software,
 66% of companies look for Open Source first (2015)
- More than 65% of companies contributing to Open Source (2016)
- 90% of respondents say Open Source improves efficiency, interoperability, innovation (2016)
- 55% say Open Source is securer than non-Open Source (2015)

(ref, 2015, 2016 Research by North Bridge, Black Duck)

http://www.slideshare.net/blackducksoftware/2015-future-of-open-source-survey-results

https://www.slideshare.net/blackducksoftware/2016-future-of-open-source-survey-results



Software = Open Source Software

Developer = Open Source Developer



Open Source의 정의

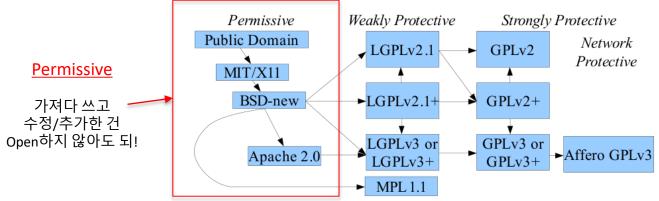
- 정의1: 자유롭게 열람, 사용, 수정하고,
 다른 사람에게 배포할 수 있도록 만든 모든 것.
 (소프트웨어, 하드웨어, 설계, 저작물, 아이디어, ...)
 - ✓ 강한 규칙: 라이선스 (저작권)
 - http://opensource.org/ GNU, BSD, Apache, MIT 등 70+ 종
- 정의2: Community에서 Open Project로 **집단 협업**으로 만든 모든 것
 - ✓ 강한 절차(process)
 - 동료 검토 (Peer Review)
 - 효율적 의사결정



Open Source License

약은 약사에게 법은 변호사에게

- OSS 라이선스는 기본적으로
 - ✓ 소프트웨어의 사용을 장려하기 위한 문서
 - ✓ 참여자들의 (경제적 또는 비 경제적) 이익이 최대가 되는 합의
 - 코드 자산, 특허, 상표권
 - ✓ 공유와 헌신에 대한 예의
 - Reputation, 원본 소스의 존중
 - ✓ 저작자의 오픈 소스에 관한 철학의 존중
 - 파생물의 공개 조건



Open Source 생태계, 핵심 질문...

- 왜 Open 해야 하는가?
 - 소스를 Open하면 안되는 이유가 있는가?
 - Open 하고, 이제 그 프로젝트는 포기하는 것인가?
- 무엇을/어디에/언제/어떻게 Open할 것인가?
 - 어디까지 (핵심 기술을) Open할 수 있는가?
 - 어느 시점에 Open할 것인가?
 - 어떤 라이선스를 선택할 것인가?
 - (커뮤니티에 의한) 지속가능성을 어떻게 담보할 것인가?
- Open한 뒤에는 뭘 먹고 살 것인가?
 - 우리의 Business Model 은?
 - 남들은 내 소스를 가져다가 어떻게 할까?
 - 프로젝트(소스)에 내재된 기술은 계속 발전을 할까?





Open이 꺼려지는 이유들 : Myth and Fact

- 시장에서의 경쟁 구도가 무서워서
 - 소스가 공개될 뿐, 기술 주도권 및 ownership은 유지
 - 기술의 Originality를 더 명확하게 유지
 - 커뮤니티에 들어온, 경쟁 기관들의 역량으로 우리 프로젝트의 품질/수월성 제고
- 현재의 코드가 Mature하지 않아서
 - Open을 위해서라도 개발 프로세스 및 코드 리뷰 체계를 정비할 기회
 - 커뮤니티와의 협업이 코드의 품질을 개선
- 지켜져야 할 영업 비밀이 있어서
 - 정말로 중요하다면 일부 코드를 비공개로 유지할 수 있음
 - 그리고 License Compliance를 위한 Governance 유지 통해서
- 사악한 Hacker가 우리 Source를 보니까
 - 소스를 이용하는 착한 Hacker가 먼저 보안 문제를 발견하고 빠르게 수정



왜 Open Source로 해야하는가?

- OSS 자체적인 Business 모델로서의 가치
 - 이미 성공적인 큰 OSS 프로젝트, 커뮤니티에만 해당
- <-----

"No Vendor Lock-in"

<<u>----</u>

- 핵심 역량을 발휘할 수 있는 부분에 집중
 - 세상 모든 OSS는 내 친구, 내 재산
 - 다른 참여자/커뮤니티와 함께 역할 분담, 동반 성장
 - 기술 부채를 떨어내고, 기술 자산을 쌓기에 집중



- 조직 스스로 또는 외부에 대한 긍정적인 메시지
 - 좋은 사람들과의 협력 가능성
 - 빠른 다양성의 확보
 - 우리 조직의 역량과 착한 회사 이미지를 세계 만방에 알리는 방법
- 인력 양성/ 리크루팅의 기회
 - 선수들이 인정한 선수들이 커뮤니티에 있음
 - 취업 전에 역량을 인정받을 거의 유일한 방법





이런 회사가 Open Source Software 회사

- 1. 뭔가가 땡길 때, Open Source를 먼저 생각합니다.
- 2. 개발자들이 커뮤니티에 나가 우리의 기술과 로드맵을 설명합니다.
- 3. 우리 회사의 개발자들을 기다리는 커뮤니티가 있습니다.
- 4. 초보 커뮤니티 개발자들의 참여를 위해 노력합니다.
- 5. 커뮤니티 개발자들의 기여를 언제나 존중합니다.
- 6. Open Source 소프트웨어 저작권을 존중합니다.
- 7. 진짜 오픈을 합니다.





https://www.flickr.com/photos/29998767@N07/

언제/무엇을/어디에/어떻게 Open할 것인가?

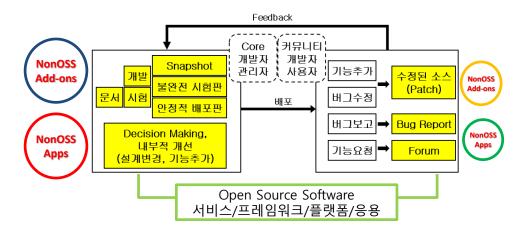
- 무엇을 만들까 고민하는 시점이 가장 정답
 - ✓ 성공한 Open Source의 기본은 Requirement를 수집할 때
 - ✓ 커뮤니티가 원하는 feature들을 Agile 하게 Release 가능
- 우리 프로젝트의 일부 모듈이 유용해진 시점
 - ✓ 다른 프로젝트와 엮어 데모가 가능해질 때
 - ✓ 우리 프로젝트의 가능성을 확인, 초기 사용자, 개발자 층을 확보
- Use Case를 만들어서 Test가 가능한 시점
 - ✓ 결과물 전체가 온전히 Build 가능해지고
 - ✓ 개발자들에게 흥미 있는 (and/or 사용자에게 유용한) 예제를 같이 제시





언제/무엇을/어디에/어떻게 Open할 것인가?

- 핵심 기술 영역을 반드시 포함해서 오픈
 - 누군가에게 의미 있으려면, 내게도 의미가 있어야..
 - Roadmap, 커뮤니티 Governance를 같이 공개
 - 특정 고객을 위한 부가가치 영역(동그라미부분)은 아직도 closed
 - 진정 보호하고픈 IP 부분은 우회적인 (덜 최적화된) 코드로 Open 가능
- 기술/서비스 Homepage를 유지하고,
 - 코드는 남들처럼 GitHub에...





언제/ 무엇을/어디에/어떻게 Open할 것인가?

- 기여자에 대한 배려 (community governance)
 - 관리자, 코어 개발자 (committer), 기여자들 목록의 공개
 - 기여 방법 / 절차를 소개 (Pull Request / Mailing List)
 - 커뮤니티의 의사결정 방법
 - CLA (Contributor License Agreement)
- 특히, 신규 (초보) 개발자를 위한 배려
 - 확실히 돌아가는 예제와 Build 문서, Tutorial 동영상
 - Play Ground (Sand Box) 제공
 - Virtual Machine (Container) Image와 실행하는 법 + Link
 - ✓ 모든 개발, 실행 프로그램을 담고,
 - ✓ 자신의 시스템에 영향없이 시험, Build 해볼 수 있는 환경



- 동기를 불러일으키기 위한 동영상,사진, 자료, 기사들
- Blog / Paper / Technical Report : Architecture, 기술적 기초
- API, Build / Test 방법, Coding Style
- FAQ





Open 한 뒤에, 해야할 일들

- 외부 기여자에 대한 진정성 있는 Care (라고 쓰고 홍보라고 읽기)
 - Mail 응대, Pull Request에 대한 반응
 - Review is Everything
 - 블로깅, Meet-Up
- 그리고 마케팅
 - 개발자 컨퍼런스 참가
 - OSCON과 같은 다양한 주제의 컨퍼런스에서 자랑
 - 더 크고 유명한 파트너사의 컨퍼런스에 묻어가기
 - 개발자 컨퍼런스 개최
 - 우리 기술과 Roadmap을 공유하여 참여 동기 유발
 - Talks : 우리 프로젝트, 관련있는 다른 프로젝트, 개발자 문화, ...
 - Showcase / Workshop / Hackathon
 - 학교와 협력
 - In-class Project, Capstone, R&D, ...





Head of Open Source Software Job Description

- Key Functions
 - Carry out detailed analysis of technical and product marketing information to create and guide OSS project developments
 - Engage directly with internal customers, 3rd party suppliers and key Open Source Software projects to guide the technology to be used
 - Represent at external events and Open Source forums, giving presentations and participating in discussions
 - Promoting Open source best practices across engineering groups
 - Team building, recruitment, project planning and budgeting for OSS projects

+ 실행력 있는 Evangelists, Governance 조직



Open Source □ Business Model

- Open 한 뒤에도, 아직 우리는
 - 해당 핵심 기술 영역에서 우위를 유지
 - 핵심 기술의 Ownership을 유지
 - 저작권자/상표권자/특허권자 지위를 유지
 - Open 하지 않는 소스를 유지



- Typical Open Source Business Model
 - 1. Open Source로 Project를 진행하는 것 자체가 이득 !!
 - 2. Subscription (aka, Technical Support, Education...)
 - 3. Dual License (GPL + Commercial License)
 - 4. Dual Version (Community version, Commercial Version)
 - 5. Certification + Brand Package

* 대개는 여러 BM을 같이 운영



요약

- Open Source Software 20년
 - Why 에서 What으로, What 에서 How로
 - 공유에서 참여로, 참여에서 멘토링으로
- Way to Open Source

- Step 2 : 진정성 그리고 기다림

- Step 0 : 통제를 포기하는 신념

- Step 1 : Open하고 있어야 Open



http://polinlove.tistory.com/2126



Minsuk Lee

국민대학교 소프트웨어학부

http://hl1itj.tistory.com



이 저작물은 크리에이티브 커먼즈 [저작자표시-비영리-동일조건변경허락 2.0 대한민국 라이선스]에 따라 이용할 수 있습니다.