

OSS와 RMS 개발·보급 전략

| 일 시 | 2021. 4. 27. (화) 14:00 ~ 16:00

| 장 소 | 정부대전청사(2동 403-1) 온·오프라인 병행

| 주 최 | 행정안전부 국가기록원



행정안전부
국가기록원

정책포럼 진행 일정

시 간	진행순서	내 용	비 고	
14:00~14:10	10'	개회	포럼 주제 및 참석자 소개	사 회 자
14:10~14:30	session 1	<div style="text-align: center;">주 제 1</div> OSS의 의미와 라이선스 준법성 및 공개SW 활용전략	발 표	
14:30~15:00			토 론	
15:00~15:10	10'	질의·응답	[주제1]에 대한 질의·응답	참 석 자
15:10~15:30	session 2	<div style="text-align: center;">주 제 2</div> 기록관리시스템 개발 확산을 위한 소스 오픈 정책 도입전략	발 표	
15:30~15:45			토 론	
15:45~15:55	10'	질의·응답	[주제2]에 대한 질의·응답	참 석 자
15:55~16:00	5'	폐회	마무리 및 폐회	사 회 자

Contents ||

|| 주 제 1

- ▶ OSS의 의미와 라이선스 준법성 및 공개SW 활용전략 **1**
이민석(국민대학교)
- ▶ (토론) 오픈소스를 지탱하는 몇가지 원동력 **14**
김두현(건국대학교)
- ▶ (토론) '오픈소스 소프트웨어(OSS)의 의미와 라이선스 준법성
및 공공소프트웨어 활용전략'에 대한 토론문 **23**
한복미(정보통신산업진흥원)

|| 주 제 2

- ▶ 기록관리시스템 개발 확산을 위한 소스 오픈 정책 도입전략 **27**
한능우(국가기록원)
- ▶ (토론) RMS 개발확산을 위한 소스오픈 정책 도입전략 **38**
안대진(아카이브랩)

주 제 1

OSS의 의미와 라이선스 준법성 및 공개SW 활용전략

이 민 석(국민대학교)

오픈소스 소프트웨어의 의미와 라이선스 준법성, 공공 소프트웨어 활용전략

이 민석

이노베이션 아카데미 학장
(국민대학교, 소프트웨어학부 교수)



Innovation Academy

Microsoft Encarta

- Microsoft의 백과사전
 - Bill Gates 80년대 중반부터 '힘주어' 시작
 - 핵심은 CD-ROM 기반 Multi-media 백과사전
 - 당시 경쟁사 : Britannica (1768년 ~)
 - Britannica의 1990년 매출 : US\$650M



- 1993년 Encarta 첫번째 버전 출시
- 2009년 62,000 articles, and ...

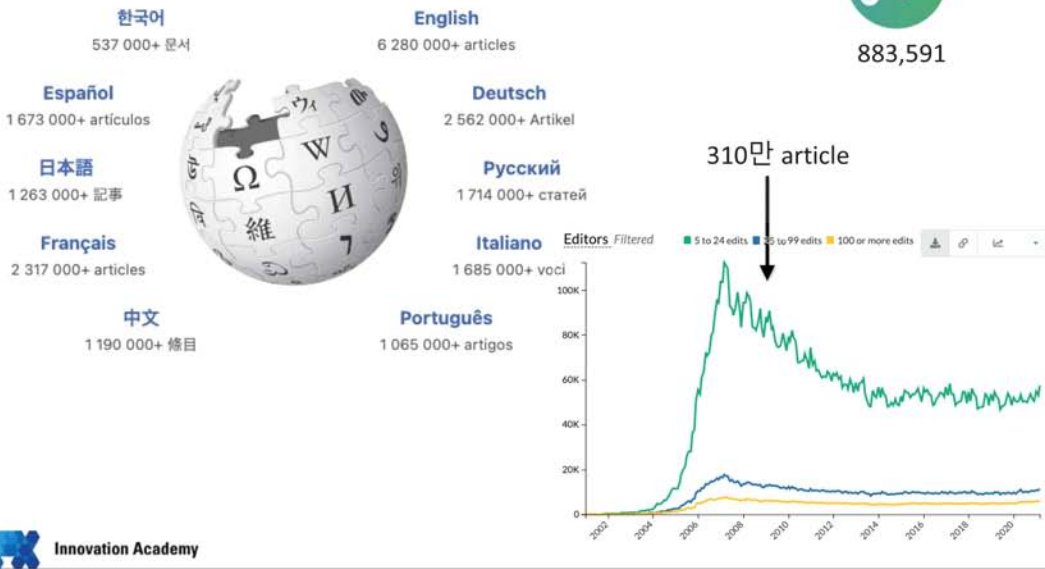
- Britannica, on 2012, ...



62,000 Articles

Wikipedia

- 2001년에 시작



Lee, Sedol vs. AlphaGo, 2016

저는 AI가 싫어요



- AI Issues
 - ✓ The Ethics of AI
 - ✓ To order or To be ordered
 - ✓ Basic Income, Robot Tax

On the year of 2015,

- 'TensorFlow' by Google
- 'DMLT' by Microsoft
- 'SystemML' by IBM
- 'Ves' by Samsung
- 'Torch' by Facebook

Now,

- Apache Spark, MLlib, Apache Singa, Caffe,
- MS Azure ML Studio, AML, MS CNTK, Brainstorm, mpack 2, Marvin, Neon, ..
- 그리고, <https://arxiv.org/>

as Open Source



Why Open Source Software Contribution?

- **It's fun**
- It can lead to increased happiness
- Learn to code
- Improve coding skills
- Gain early experience
- Improve production level software experiences
- Train you in ways that other people aren't
- Provide a platform to do whatever you want
- Attract quality fellow developers
- Experiencing critical mass
- Stand on the shoulders of giants
- Involve in every design decision
- Build features you want
- Community and peer recognition
- Greater job prospects
- There are good eyes on who is contributing what
- It shows you're innovative
- Positions of public trust
- Direct democracy
- Trust enables cooperation
- Self-organization with shared goals
- Learn to communicate positively
- Learn respect other people's ideas
- It can inspire passion
- Teaching the next generation
- It's idealistic
- **It's the way of the future**



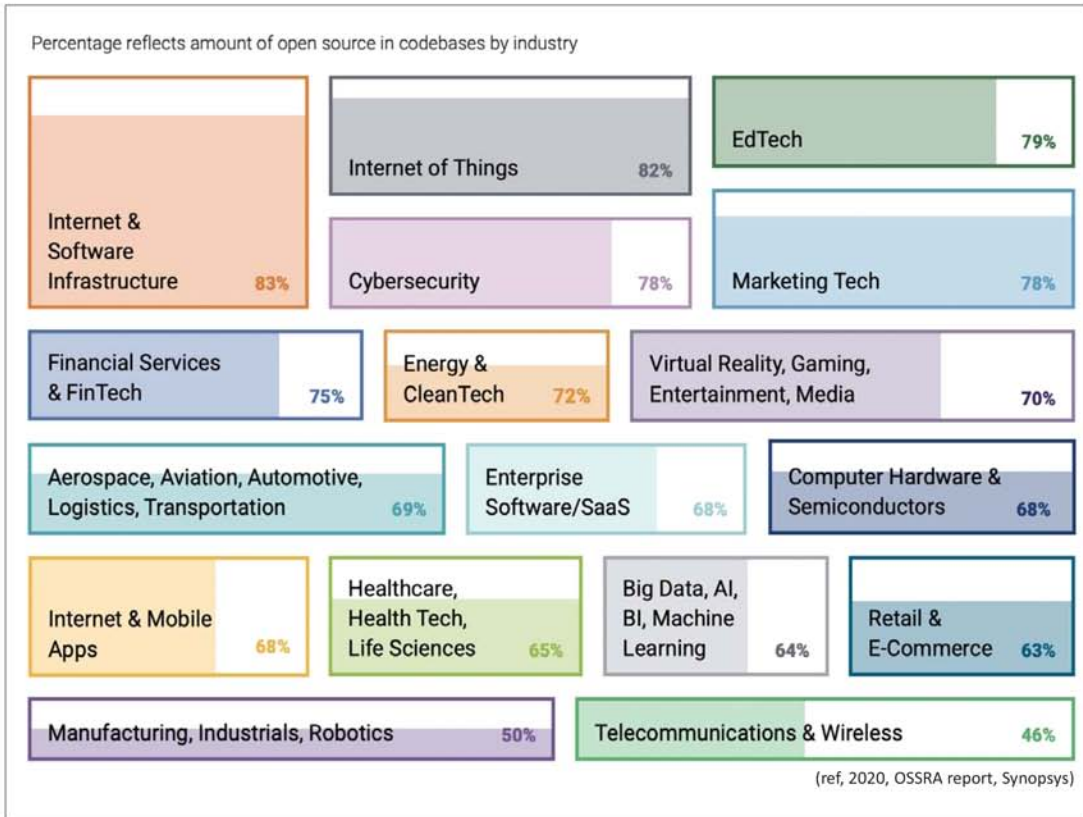
Why Open Source Software ?

- Multiplying the company's investment
- Benefitting from the most recent advance
- Spreading knowledge of the software
- Increasing the developer base
- Upgrading internal developer skills
- Building reputation
- Recruiting and retaining developers
- Faster startup of new companies and projects



Ref. Open Source in the Enterprise, Andy Oram & Zaheda Bhora
(<http://opensource.amazon.com>)





오픈소스 비즈니스



오픈소스, 현재 상황

- 커뮤니티 주도의 오픈소스 프로젝트
- 개발자들의 재단 (사상적 연합체)



- 오픈소스 프로젝트의 재단
- 개발자들의 재단 (기술적 커뮤니티 연합체)
- 회사 주도의 오픈소스 프로젝트
- 오픈소스의 적극적 사업화 (클라우드)
- 회사들의 재단 (사업적 보호막)
- 회사들의 개발 연합체 (거대한 Lock-in을 추구하는 플랫폼)



Open Source의 정의

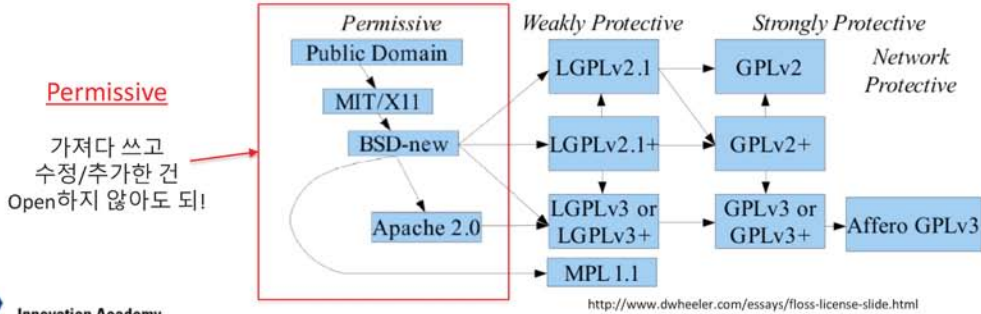
- 정의 1 : 자유롭게 열람, 사용, 수정하고,
다른 사람에게 배포할 수 있도록 만든 모든 것.
(소프트웨어, 하드웨어, 설계, 저작물, 아이디어, ...)
- ✓ 강한 규칙: 라이선스 (저작권)
 - <http://opensource.org/> - GNU, BSD, Apache, MIT 등 70+ 종
- 정의 2 : Community에서 Open Project로 집단 협업으로 만든 모든 것
- ✓ 강한 절차(process)
 - 동료 검토 (Peer Review)
 - 효율적 의사결정



Open Source License

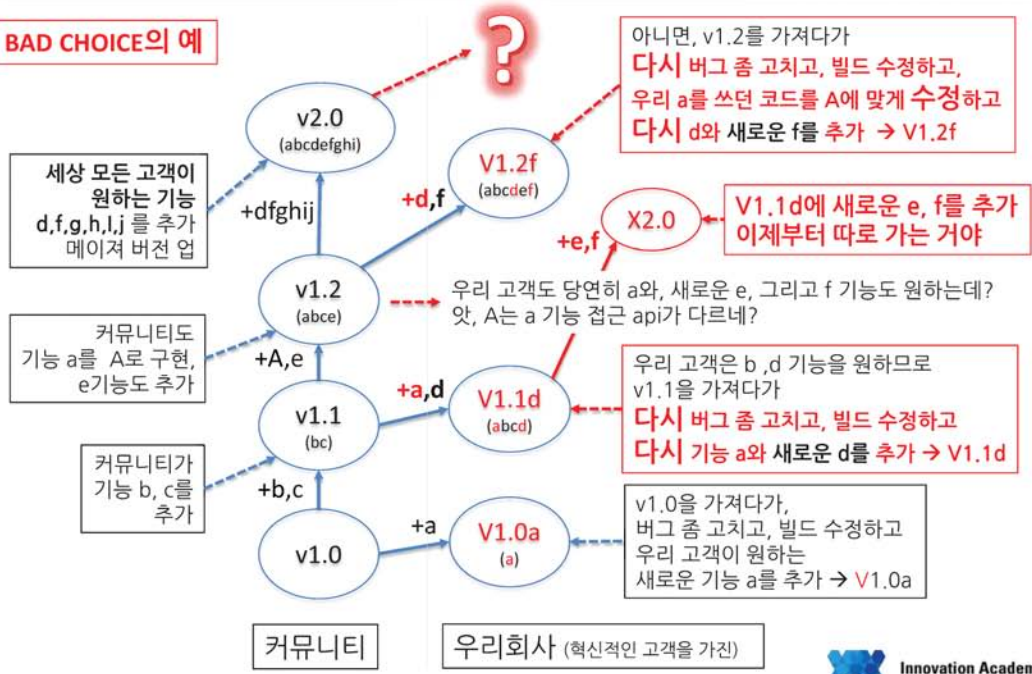
약은 약사에게
법은 변호사에게

- 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



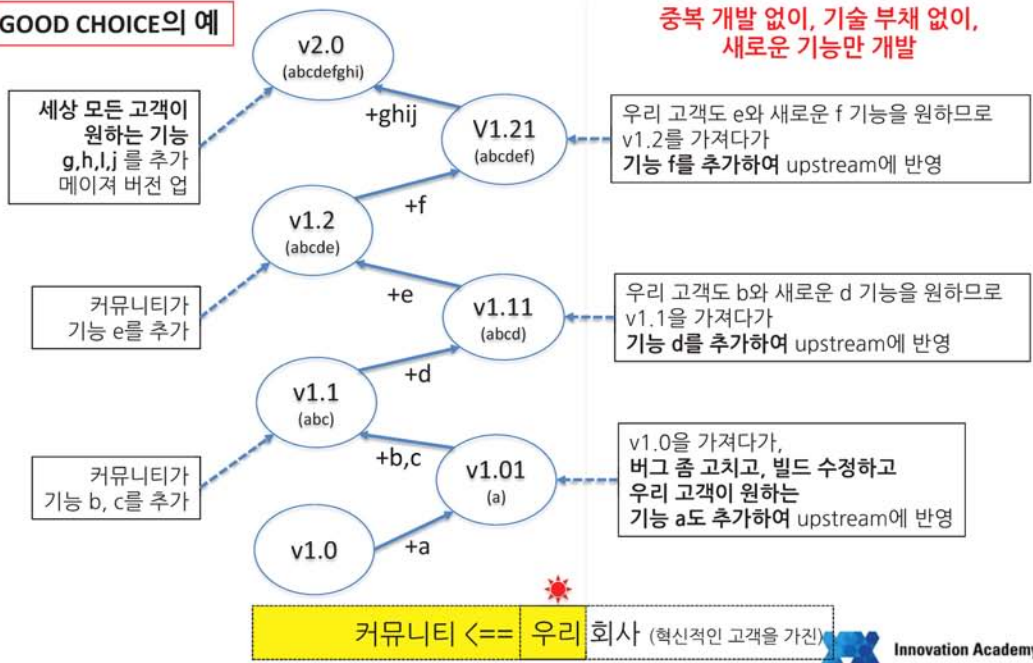
Permissive License는 동작할까?

BAD CHOICE의 예



Permissive License는 동작할까?

GOOD CHOICE의 예



오픈소스 소프트웨어 라이선스의 특별한 문제

- 기술적, 법적 문제
 - 본질적으로 완벽하기 어려운 소프트웨어의 문제
 - 품질 보증 vs 공유와 기여
- 계약적, 법적 문제
 - 계약서의 완결성 부족
 - 분쟁 해결, 언어, 관할 법원, 준거법
- ❖ 문제의 이유
 - Copyleft Movement로 시작된 선언적 강령으로서의 라이선스
 - 공동 목표를 위한 행동 지침으로 법적 조치를 전제로 하지 않음
 - 법률 전문가가 아닌 개발자들이 작성
 - 특정 계약 상대방이 아닌 "공중(public)"을 위한 간결한 공지 목적
 - 미국에서 시작되어 국제적 시각과 고려가 부족

라이선스 준법성 준수를 위한 OSS 거버넌스

- OSS 거버넌스 필요성
 - 오픈소스 소프트웨어를 활용한 개발의 증가
 - 외부의 기술적 자산을 활용하여 비용 절감, 내부 역량은 핵심 기술 개발에 집중
 - 오픈소스 라이선스는 복잡하고 다양함
 - OSI 인증 라이선스를 포함하여 2,000여 개의 저작권
 - 복합적 IP 자산 보호 및 관리 필요
 - 소프트웨어, 특허, 상표권
 - 외주 개발에서의 공급망 관리 필요
 - 외주 개발 소프트웨어의 잠재적 위험성 증가
- OSS 거버넌스의 목적
 - 오픈소스 소프트웨어 사용에 따른 관리 체계 확보
 - 의무 사항의 준수로 장기적 법률적, 기술적 문제 사전 방지
 - 오픈소스 라이선스 생태계 파악
 - 라이선스의 출처 관리, 준법성 확인
 - 궁극적으로 조직 내부의 지적 자산의 관리
 - OSS의 전주기적 관리 및 통제 프로세스 확보



라이선스 준법성 검토 사항

- 사용권 고지의 의무
 - 누가 해당 소프트웨어를 개발하였는지 공지함
 - 고객에게 어떤 오픈소스를 사용하였는지 알림
- 저작권 고지의 의무
 - 소스 코드 상에 이미 표시되어 있는 저작권 관련 문구는 절대로 삭제하여서는 안 됨
- 소스 코드 공개의 의무
 - 일부 오픈소스 라이선스는 오픈소스로 개발한 결과물의 소스코드 공개를 의무화 하고 있음
- 특허 포기의 의무
 - 사용자가 특허 포기의무가 있는 라이선스로 배포된 오픈소스로 만든 2차 저작물에 대하여 특허권리를 주장할 수 없음

주로 가져다 쓰는 쪽의 고민
=
잘 가져다 쓰게 하고 싶은 쪽의 고민



공공 프로젝트를 ...

Open Source로 하기로 했다면
공공, 민간, R&D의 구분이 애매해짐



공공 프로젝트를 오픈소스로 해야하는 이유

- 세금으로 만든 것 아닙니까?
- “No Vendor Lock-in”으로 서비스 지속 가능성 향상
- 국가적 이득과 일자리 창출, 개발 생태계 활성화
 - 대개 외국으로 나가는 기술 라이선스료를 없애고
 - 국내 인력으로 추가 기능 개발, 유지보수 : 일자리 창출
- 핵심 역량을 발휘할 수 있는 부분에 집중
 - **기술 부채에 대한 고민없이, 기술 자산 쌓기에 집중**
 - 지속적인 유지보수의 일부를 개발 커뮤니티에서 !!
 - 높은 개발 역량을 가진 그룹과의 협력 가능성 → 혁신의 가속화
- 더 많은 사업화 기회, 연구/개발 인력의 커리어 개발
 - 더 진보된 연구, 사업화 기회를 민간에 제공
 - 개발 업체에게는 매우 좋은 reference를 제공



공개되지 않던 이전 공공 프로젝트와 다른 점

“개발자”라는 새로운 고객
“오픈소스 산업”이라는 어려운 고객



“커뮤니티”라는 새로운 환경

- 개발자 (재미를 먹고 사는)
- 소스를 활용하는 소프트웨어 기업 (혁신을 만드는)
- 소스가 포함된 소프트웨어 사용 기업 (편리를 추구하는)



오픈소스 공공 프로젝트

- 새로운 과제 관리 거버넌스가 필요
 - 갑(공공) - 을 - 병 - 정 → 갑(공공) - (을, 을, 을, 을, 을, ...)
 - 통제가 안되는 **강력한** 을들이 존재
 - 'Open Source' 또는 'Community' 개발 Practice 경험이 부족한 을
 - (약간의) 라이선스 이슈
 - '약간' 인 이유는 우리도 공개할 것이기 때문에
 - 하지만 legacy 코드는 문제
 - (약간의) 보안 이슈
 - 보안 스캔과 빠른 패치로 해결
 - 계약 관행 (RFP, 계약, 검수, 유지보수, ...)
 - 기능 + 결과물 중심에서
 - 개발 과정에 관한 요구 + 커뮤니티 관리를 추가하고
 - 검수 후 유지보수 체계 (지속적 라이선스 및 보안 패치)
 - 이후, 라이선스의 변경, 외부 개발자의 저작권 권리 관계 등



몇 가지 국내 사례

- 기술은 개발했으나, 커뮤니티 오픈소스화는 잘 안되고 있음
 - 하모니카
 - 부여 Linux, 구름
 - 전자정부 표준 FrameWork
 - PaaS-TA
 - NIPA의 기업 지원 사업들
 - 국책 연구 기관들의 프로젝트
 - 여러 국가 R&D
- 커뮤니티를 고려한
오픈소스화 전략 부족
(커뮤니티 Building / 마케팅 예산 배정 필요)



요약

- Open Source Software 20년
 - Why 에서 What으로, What 에서 How로
 - 공유에서 참여로, 참여에서 적극적 생산으로
- Way to Open Source
 - 전략이 필요
 - 거버넌스가 필요



Minsuk Lee

이노베이션 아카데미

<http://hl1itj.tistory.com>



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

오픈소스를 지탱하는 몇가지 원동력

김 두 현(건국대학교)



오픈소스를 지탱하는 몇가지 원동력

2021. 4. 27

오픈소스 소프트웨어 재단
김 두 현

Contents

- I 오픈소스 최근사례
- II 오픈소스 다양성
- III 오픈소스 개발자 커뮤니티
- IV 오픈소스 라이선스
- V 오픈소스 재단
- VI 오픈소스 사례 창출
- VII 토론 주제

I 오픈소스 최근 사례

01 Ingenuity

▶ 2021년 4월 19일

(출처) [Flying on Mars fueled with open-source software | ZDNet](#)

▶ Flying on Mars fueled with open-source software

- "... Its mission is to show that it's possible to fly on Mars using commercial off-the-shelf (COTS) hardware and **open-source software**."
- GitHub CEO Nat Friedman and his team and the JPL Ingenuity crew took a long hard look into the helicopter's code and found that "**nearly 12,000 developers on GitHub contributed to Ingenuity's software via open source**."
- The developer list was created by JPL providing GitHub with a comprehensive list of every version of every open source project used by Ingenuity. GitHub could then identify all the contributors who made these projects and their **dependencies**. Some of those honored, such as Linux's creator Linus Torvalds, are famous developers.



(이미지출처) <https://www.nasa.gov/image-feature/jpl/ingenuity-s-first-black-and-white-image-from-the-air>

I 오픈소스 최근 사례

02 오픈소스가 선사하는 가치

주요 가치

1. 개발 효율성 제고
2. 자원 재활용
3. 생태계 구축
4. SW 품질향상
5. 고급 인재양성

공개SW 활용과 커뮤니티 개발자를 통한 시간, 인력 등 SW 개발비용 절감

하나의 오픈소스가 다양한 연구개발 및 사업화 등으로 재활용 가능한 공공재로서 역할

커뮤니티가 새로운 아이디어의 창구 역할을 하며, 새로운 응용분야로 퍼져 나가는 구심점

기획, 설계, 구현 및 시험 등 개발과정이 공개 진행되어 보다 좋은 품질의 기술로 발전

커뮤니티 방식 개발로 다양한 계층의 개발자 간 교류로 글로벌 SW 개발자로 성장 가능



▶ 오픈소스 가치를 실현하고 지탱하는 주요 원동력에 대하여 살펴보고자 함

II 오픈소스 다양성

시스템 소프트웨어, DBMS, Web, etc...

▶ 운영체제	▶ 클라우드	▶ 웹(WEB)	▶ WAS (Web Application Server)	▶ 데이터관리
Asianux CentOS Debian Fedora FreeBSD HamoniKR ORACLE LINUX RED HAT ENTERPRISE LINUX Scientific Linux SUSE ubuntu	Apache CloudStack CLGUDya 지연클라우드 CLOUD FOUNDRY OPENSHIFT openstack RED HAT OPENSTACK PLATFORM RED HAT CLOUDFORMS OPENSHIFT ENTERPRISE Selim Cloud Integrator	APACHE RED HAT JBOSS WEB SERVER NGINX resin	RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM NAWAS RED HAT JBOSS FUSE uWSGI WildFly	CUBRID MariaDB mongoDB MySQL PERCONA PostgreSQL EDB eXperDB ankus hazelcast HORTONWORKS ARCUS HamoniKR Resall continuent

※ 출처 : 국방분야 공개SW 현황 조사, SPRi 소프트웨어정책연구소, 2017.12.

III 오픈소스 개발자 커뮤니티

01 개발자 현황

글로벌 오픈소스 프로젝트 현황

- 전 세계 대표 공개SW 소스코드 저장소인 깃허브(Github)에 약 38만건의 공개SW 프로젝트 진행
- 약 2,000만 명의 사용자. 개발자가 참여

국내 커미터 현황

- 국내 SW개발자 중 커미터는 516명으로 총 1,398개의 글로벌 공개SW 프로젝트에 참여
- [참고] 커미터란?
 - 공개 소스 프로젝트에서 가장 핵심이 되는 개발자로 코드 수정 권한을 갖고 해당 프로젝트에 직접 관여하는 사람.
 - 공개SW 프로젝트에서 소스코드 개발에 참여할 뿐만 아니라 컨트리뷰터가 제출한 소스코드를 리뷰하여 프로젝트에 반영하는 최고급 SW 기술자

III 오픈소스 개발자 커뮤니티

01 개발자 현황(공헌자 비율)

▶ 국외의 영향력있는 오픈소스의 경우

글로벌 대기업이 주도하지만 직원보다 일반개발자의 참여비율이 높은 구조

프로젝트 창시기업	오픈소스 프로젝트	전체개발자(명)	프로젝트 이해당사자(명)	이해당사자 공헌비율	일반개발자(명)	일반개발자 공헌비율
구글	GO	910	118	13%	792	87%
	TensorFlow	791	119	15%	672	85%
마이크로소프트	VisualStudio	307	38	12.4%	269	87.6%
	CNTK	165	75	45.5%	90	54.5%

인포월드(IDG) 2018 최고의 오픈소스 소프트웨어



III 오픈소스 개발자 커뮤니티

02 개발자 커뮤니티 (국내 사례)

커뮤니티명	내용	사이트
Textcube	<ul style="list-style-type: none"> 구 테라플즈 개인이 블로그를 운영하기 위해, 설치하는 웹어플리케이션 팀블로그를 운영할 수 있으며, OpenID 기반의 댓글을 남길 수 있음 다국어 지원, 플러그인 구조를 이용한 확장성 제공, 스킨의 자유로운 설치 제공, 플러그인과 스킨을 공유할 수 있는 게시판 제공 	http://www.textcube.org
eXria	<ul style="list-style-type: none"> 토마토시스템 RIA, eXtensible RIA Architecture 	http://www.exria.org
KLDP (Korean Linux Documentation Project)	<ul style="list-style-type: none"> 한국 대표 리눅스 커뮤니티 대표 국내 OSS 사용자 및 개발자 프로젝트 사이트 784개의 개발자 커뮤니티 프로젝트 	http://kldp.net
네이버 개발자 커뮤니티	<ul style="list-style-type: none"> 지식의 공유와 상생 Jindo/ NTAF(NHN Test Automation Framework) 개발자용 나눔고급 코딩공감/ CUBRID/ nFORGE Xpress Engine/Dist/ Neptune 등의 프로젝트 	http://dev.naver.com
다음 개발자 커뮤니티	<ul style="list-style-type: none"> 창의적인 개발자를 위해 열린 다음의 공유서비스 GLAMJI(GNU/Linux+Apache+MySQL+Java+IDE), MODXSS 	http://dna.daum.net
anyframe	<ul style="list-style-type: none"> 오픈소스 기반의 어플리케이션 프레임워크와 이를 쉽게 사용할 수 있도록 체계적인 가이드 제공 Anyframe/ Gen/ IAM/ Monitoring/ Oden/ Query Manager 	http://www.anyframejava.org
uEngine BPM	<ul style="list-style-type: none"> Business Process Management를 오픈소스화 국내 OSS 커뮤니티 uEngine BPMS/ uMonitor RFID / USN BPMS/ uBrain Rule Engine/ Kalm 등 	http://www.uengine.org
한국 공용 버그질라	<ul style="list-style-type: none"> 글로벌 오픈소스 커뮤니티에서 활동하는 한국 개발자들이 소스 개발에 관심있는 국내 개발자들을 위해 소스코드 버그들을 미리 판별해 주고 이를 해외 프로젝트에서 버그로 처리하는 방식 및 퍼치 제작 등을 멘토링해 주기 위해 만들어진 사이트 	http://bugzilla.kr
기타 국내 오픈소스 사용자 커뮤니티	<ul style="list-style-type: none"> 한국리눅스유저그룹 SUSE Linux 사용자 모임 GNOME 한국 사용자 모임 KELP 데이터베이스 사랑넷 (DSN) Korea Apache Group PHPSCHOOL 한국 자바 개발자 협의회 (JCO) 	<ul style="list-style-type: none"> 데브피아 한국 소프트웨어 커뮤니티 연합 (SCA) 한국 스프링 사용자 모임 오픈 오피스 한국어 커뮤니티 김프 코리아 한국 무들 사용자 모임 한국 모질라 커뮤니티

※ 출처 : 공개SW 군 확산을 위한 정책수립 연구, 국안성립그룹, 2018.12.



IV 오픈소스 라이선스

주요 공개 소프트웨어 라이선스의 특징 비교

라이선스의 특징 및 의무사항	BSD	Apache 2.0	GPL 2.0	GPL 3.0	LGPL 2.1	MPL	CDDL	CPL/EPL
복제-배포-수정의 권한 부여	○	○	○	○	○	○	○	○
배포시 라이선스 사본 첨부		○	○	○	○	○	○	○
저작권고지사항 또는 Attribution 고지사항 유지	○	○	○	○	○	○	○	○
배포시 소스코드 제공 의무(Reciprocity)와 범위			derivative work	work based on the program	derivative work	file	file	module
조합저작물(Larger Work)작성 및 타 라이선스 배포 허용	○	○			○	○	○	○
수정시 수정내용 고지		○	○	○	○	○	○	○
명시적 특허라이선스의 부여		○		○		○	○	○
라이선스가 특허소송 제기시 라이선스 종료		○		○		○	○	○
이름, 상표, 상호에 대한 사용제한	○	○				○	○	
보증의 부인	○	○	○	○	○	○	○	○
책임의 제한	○	○	○	○	○	○	○	○

※ 출처 : 국방분야 공개SW 현황 조사, SPRi 소프트웨어정책연구소, 2017.12.



V 오픈소스 재단

01 오픈소스 재단의 필요성

▶ 법적 지원 측면

- 과실로 인한 소송, 계약 상의 책무 등으로부터 참여자를 보호할 수 있는 방어벽 역할
- 라이선싱, 저작권, 특허, 기타 지적 재산관리를 위한 **법적 틀 지원**
- 오픈소스 소프트웨어 **라이선스 개발 및 적용 가이드**

▶ 개발자 지원 측면

- 소프트웨어 저장소와 Certificate 등 **기술 지원 서비스**
- 프로젝트 **운영지원** - 회원관리, 개발자료 관리, 회계관리, 보도 및 홍보

▶ 기업 지원 측면

- 기업들 간에 오픈소스 프로젝트를 **공동으로 추진**하기 위한 신뢰성 있는 비영리 독립단체 필요
- 경쟁 기업들의 경우 기업 입장에서 **중립적**이면서도 경쟁적이지 않은 매개체 필요



V 오픈소스 재단

02 주요 오픈소스 SW 재단

※ 출처 : 글로벌 오픈소스 프로젝트 현황, 한국정보화진흥원(NIA), 2018.11.

오픈소스 재단명	재단 개요
아파치 소프트웨어 재단 (Apache Software Foundation : ASF)	<ul style="list-style-type: none"> 아파치 소프트웨어 프로젝트를 지원하는 비영리 재단 다양한 오픈소스 프로젝트를 지원·관리함 1999년, 아파치 그룹과 델라웨어사의 합병으로 설립
리눅스 재단 (Linux foundation)	<ul style="list-style-type: none"> 리눅스 발전, 협업 개발을 위한 비영리 연합체 2007년 OSD(오픈 소스 개발 랩)과 FSG(자유 표준 그룹)의 합병으로 설립
클라우드 파운드리 재단 (Cloud Foundry Foundation)	<ul style="list-style-type: none"> 2015년 클라우드 파운드리 재단이 비영리 독립 리눅스 재단 협업 프로젝트의 하나로 설립 Cloud Foundry를 전세계 클라우드 컴퓨팅을 위한 선도적인 애플리케이션 플랫폼으로 만드는데 목적
소프트웨어 프리덤 컨저번시 (Software Freedom Conservancy)	<ul style="list-style-type: none"> Open Source Software (FOSS) 프로젝트의 홍보, 개선, 개발 및 법률 서비스, 인프라 지원을 하는 비영리 자선단체 Software Freedom Law Center (SFLC) 의 도움으로 2006 년에 설립 40개 이상의 회원 프로젝트를 운영
오픈스택 재단 (OpenStack Foundation)	<ul style="list-style-type: none"> 2012년 창설된 비영리 단체인 OpenStack Foundation은 개방형 인프라의 글로벌 개발, 배포 및 채택을 지원 OpenStack 클라우드 운영시스템 개발, 배포, 도입 프로젝트 지원
프리 소프트웨어 재단 (Free Software Foundation)	<ul style="list-style-type: none"> 1985년 자유 소프트웨어의 생산과 보급을 장려하기 위해 리처드 스톨만이 세운 자유 소프트웨어 관련 재단 GNU 프로젝트의 자유 소프트웨어 작성 및 자유 소프트웨어 운동과 자유 소프트웨어 커뮤니티를 위한 법적, 구조적 문제 지원
이클립스 재단 (Eclipse Foundation)	<ul style="list-style-type: none"> 2004년 설립된 이클립스재단은 비영리단체이며 오픈소스 기반의 통합개발환경인 이클립스를 개발하는 커뮤니티를 지원하는 조직 IT 기반환경을 제공, 지적 재산권(EPL) 관리, 개발 커뮤니티를 지원, 개발 생태계 발전에 목표
오픈소스이니셔티브 (Open Source Initiative : OSI)	<ul style="list-style-type: none"> 1998년 오픈 소스 소프트웨어 사용을 장려하기 위하여 만들어진 단체 오픈소스의 장점을 홍보 및 주장하는 활동 및 OSI 라이선스 검토와 승인 등 OSD(Open Source Definition)를 관리



▶ 국내에는 공개SW 관련협회 및 개발자센터만 있고 관련 재단은 거의 없었으나 재단설립준비위원회가 2016년에 순수민간단체 '오픈소스SW재단'을 출범(http://ossf.org/)

V 오픈소스 재단

03 기업 참여

Rank	Company	Employees Contributing
1	Microsoft	4,550
2	Google	2,267
3	Red Hat	2,027
4	IBM	1,813
5	Intel	1,314
6	Amazon.com	881
7	SAP	747
8	Thoughtworks	739
9	Alibaba	694
10	GitHub	676

No	프로젝트	주관	Github fork	Github favorite	License
1	텐서플로우(TensorFlow)	Google	69,496	114,288	Apache2.0
2	케라스(Keras)	Google	13,445	35,309	MIT
3	파이토치(PyTorch)	Facebook	5,020	21,095	BSD
4	Fast.ai	Fast.ai	3,431	9,631	Apache2.0
5	체인너(Chainer)	Preferred Networks	1,137	4,277	MIT

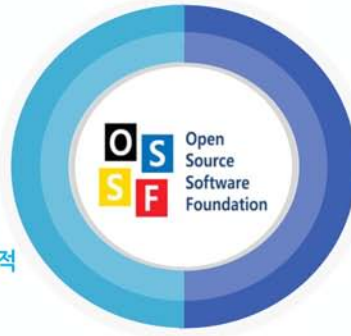


인포월드(IDG) 2018 최고의 오픈소스 소프트웨어

V 오픈소스 재단

04 OSSF

OSSF는 비영리
공익재단으로서,
(1) 오픈 소스 소프트웨어
발전에 기여하며
(2) OSS 관련 국내 산업
활성화를 위한 종합적이고
실질적인 활동을 전개하고
(3) 일관성 있고 체계적인 법적
위험관리 방안을 제공한다.



OSS 관련 국내외 조직들과
협력을 통하여,
(1) 재단 관련 기관과 기업
그리고 커뮤니티를 대변하며
(2) 우수한 국내 SW 인력의
양성과 세계 활동을 지원하고
(3) Asia의 OSS 중심축(Hub)이
되어 OSS의 아시아 브랜드를
만들어 나간다.



VI 오픈소스 사례 창출

정부 주도 사례

▶ **구름플랫폼**

- 국가-공공기관 업무환경 보안주권 부재, 클라우드 업무환경 패러다임 등장, 新업무환경에서의 단말 보안기술-체계 필요에 따라 정부 R&D 지원으로 개발
- 2019년 소스코드 오픈

▶ **PaaS-TA (<https://www.paas-ta.kr>)**

- 오픈소스 기반의 PaaS(Platform as a Service)
 - ✓ 개방형 클라우드 기술 개발, 표준화, 공공 부문 선도 적용 · 확산, 국내 기업 기술 경쟁력 강화 지원 목적
 - ✓ 과학기술정보통신부와 한국지능정보사회진흥원(NIA)이 연구개발(R&D) 지원
- 2016년 4월 Paas-TA 1.0 (공식 버전) 공개, 2021년 2월 Paas-TA 5.5 - Semini 출시
- 차세대 전자정부 서비스의 공통기반인 전자정부 클라우드 플랫폼에 적용 추진 중

▶ 그 외 하모니커 등 몇가지 사례가 있으며, 생태계 구축 및 공공서비스에 도입 추진 중



VII 토론 주제

토론 주제

- ▶ 국가 기록물 관리라는 특이한 미션과 오픈소스의 개방성 간의 부합성에 대한 면밀한 검토가 필요
- ▶ 즉, 정부주도의 솔루션 관리 및 보급으로부터 코드 공개를 통한 생태계 주도형 서비스 보급으로 전환을 의미! 총체적 대응이 가능한가?
 - 공공 서비스로서의 **균일성**을 보장할 수 있을까? 관련 기관의 산발적 도입에 따른 도입 가이드라인 제정 및 관리 체계를 어떻게 갖출 것인가?
 - 디펜던시를 갖는 다양한 오픈소스 결합 시 라이선스(저작권) 관리와 분쟁 예방 방안은?
 - 기업체의 수익성을 전제로 하는 개발자, 기업, 정부(정책)가 참여하는 생태계가 구축, **활성화**될 수 있는가?
 - 생태계의 **지속성** 지탱을 위한 수단은 무엇인가? 궁극적으로 민간재단으로의 이양이 가능한가? (PaaS-TA 사례: NIA 주도의 기술지원센터 운영)



주 제 1 토론

‘오픈소스 소프트웨어(OSS)의 의미와 라이선스 준법성 및 공공소프트웨어 활용전략’에 대한 토론문

한복미(정보통신산업진흥원)

‘오픈소스 소프트웨어(OSS)의 의미와 라이선스 준법성 및 공공소프트웨어 활용전략’에 대한 토론문

한 복 미(정보통신산업진흥원)

먼저, 오픈소스 소프트웨어의 의미와 라이선스 준법성 및 공공소프트웨어 활용전략에 대해 발표해주신 이민석 학장님에 대해 감사드립니다. 특히, 공공 프로젝트를 왜 오픈소스로 해야 하는지에 대한 설명과 이로 인해서 공공(정부)는 무엇을 더 준비해야 하는지 잘 설명해주신 거 같습니다.

아시다시피, 이미 국내 및 국외의 많은 소프트웨어는 오픈소스를 활용하여 만들어져 있습니다. 이제 오픈소스 활용 없이 만들었다라고 하면 아무도 믿지 않을뿐더러 오히려, 좋은 (이미 만들어진) 기술을 안쓰고 왜 스스로 처음부터 만들었는지 의문을 갖게 됩니다. 즉, 오픈소스를 활용하면 신기술 적용, 개발효율성, 비용절감 등 다양한 이익이 있기 때문입니다.

하지만, 오픈소스를 잘 활용하는 것과 지금 논의가 되고 있는 오픈소스로 만드는 것(공개하는 것)은 다르게 여겨지기도 합니다. 국내의 많은 소프트웨어 기업들은 오픈소스를 많이 활용하고 있지만, 추가한 기능들을 기존의 공개된 소스코드에 반영(업로드)하거나, 자사가 개발한 소스코드를 공개하는 것에 대해서는 부정적인 입장이 있는 것도 사실입니다. 글로벌 시총상위 10개 기업중 7개 기업이 IT기업이며, 각각의 기업들이 오픈소스를 기업의 혁신전략으로 가져가고 있다라는 사실도 있고, 미국 실리콘벨리의 많은 스타트업 기업들은 오픈소스를 활용하고 소스코드 공개를 전략으로 취하고 있으나 아직 국내 소프트웨어 산업에서는 오픈소스를 전략적으로 접근하지 못하는 것이 현실이기도 합니다. 이제는 소프트웨어 기업의 핵심기술이 소스코드에서 데이터로 변화하고 있다는 패러다임도 중요한데 말입니다.

기업이 왜 오픈소스 전략을 가지고 오픈소스를 활용하거나 공개해야 하는지는 많은 이유가 있습니다. 먼저, 기술적인 측면에서, 새로운 글로벌 신기술은 대부분 오픈소스로 개발되어지고 있습니다. 국내 기업이 이렇게 만들어진 기술에 대해 가져다 쓰기만 한다면, 국내 기업은

글로벌 기술의 주도권을 얻기는 힘들 것입니다. 또한 공개된 메인기술이 업데이트될 때마다, 기업은 스스로 커스터마이징 한 부분에 대해서 매번 재개발을 해야하므로, 반복적인 작업이 이뤄지고 이로 인해 개발 효율성이 매우 낮아집니다. 이런 상황이 아니라면, 기업은 신기능이 적용된 오픈소스로 버전업을 하지 않으므로 기업의 입장에서는 자사 소프트웨어의 경쟁력이 떨어질 것입니다. 다음으로는 개발 인력적인 측면이 있습니다. 기업에서는 요즘처럼 좋은 개발자를 구하기가 어려운 적도 없다고 합니다. 그러나, 사실은 국내 소프트웨어 기업은 항상 개발자의 부족을 어려움으로 호소하고 있었습니다. 특히 사회초년생 개발자가 아니라 조금 할 줄 아는 개발자를 끊임없이 찾고 있었습니다. 글로벌 기업인 Google에서 소스코드를 공개하고 오픈소스를 많이 활용하는 이유에 대해서 여러 가지가 있겠지만, 대표적으로 좋은 인력을 구하기 위해서라고 합니다. Google에서는 자사의 솔루션/서비스에 대해서 어떤 오픈소스를 활용하는지를 알리고, 개발된 소스코드를 공개하면서 양질의 소프트웨어 개발자가 Google을 선택할 수 있는 기회를 많이 제공하고 있습니다. 즉, 능력 있는 개발자는 본인이 관심 있는 오픈소스를 사용하는 기업을 선택하게 됩니다. 글로벌 대표 기업인 Google조차도 좋은 인력을 구하기 위해 이러한 노력을 하고 있는 상황입니다.

또한, 기업의 자사 소프트웨어에 대한 저변확대 부분으로, 소프트웨어 마케팅 방법이 이제 바뀌고 있습니다. 기업이 자사 소프트웨어로 시장 점유율을 높이기 위해서 예전에는 영업직원이 발로 열심히 뛰었고, 그러다 보니 글로벌 진출에는 한계가 많았습니다. 물론 소프트웨어 품질도 중요합니다. 이제는 고객이 공개된 소스코드를 직접 사용해 보고 고객이 먼저 기업에게 연락하고, 많은 개발자들이 사용자(테스터)가 되어 테스트를 해주는 상황이 되었습니다. 이로 인해 소프트웨어의 품질이 올라가고, 소프트웨어의 저변확대가 되면서 글로벌 진출도 가능하게 됩니다. 이러한 이유들을 가지고, 글로벌 기업은 오픈소스 전략을 취하면서 기업혁신을 도모하고 있습니다. 그러면 소프트웨어 기업이 아닌, 공공 프로젝트는 왜 오픈소스 전략이 필요할까요?

원론적으로 오픈소스는 대표적인 공공재에 해당한다고 생각합니다. 국가 예산으로 만들어진 소프트웨어의 소스코드를 공개함으로써 많은 사람들이 같이 활용하고 그로 인해 다수의 이익이 생기고, 국가의 소프트웨어 경쟁력이 올라간다면 가장 이상적일 것입니다. 또한 기록물관리 소프트웨어처럼 다수의 기관에서 사용하면서 비슷한 요구사항에 대해 개별적으로 개발된다면 이 역시 예산낭비이며 비효율적입니다. 그렇다고 중앙의 기관에서 일괄개발해서 배포하는 것 또한 각각의 개별기관의 요구사항이 반영되기 어려운 구조로 불편함이 생길 수 있습니다. 지금처럼 개발자의 이직이 빈번한 개발 현장에서 소스코드의 공개로서 많은 개발자들의 참여하게 되고, 특정 유지관리 기업으로부터 종속을 극복하고 다양한 기업이

유지관리를 할 수 있다라는 것도 공공 프로젝트의 오픈소스 전략 필요성 중에 하나입니다.

또한 오픈소스 라이선스 이슈도 있습니다. 오픈소스를 활용하게 되면 오픈소스 라이선스에 대한 준법 의무가 생깁니다. 오픈소스 라이선스는 소프트웨어 저작권자가 소스코드를 공개하면서 누군가가 활용할 경우, 해야 할 의무사항을 명시하고 있습니다. 소소한 고지 의무부터 소스코드 공개까지 라이선스에 따라 다양합니다. 적어도 국가 예산으로 만들어지는 공공 프로젝트는 라이선스 문제가 발생되면 안된다고 생각합니다. 국가의 품격, 도덕성과 관련있습니다. 개발자 또는 기업에서 오픈소스 라이선스 준수가 많이 어려운 일은 아닙니다만 꼭 체크해야 할 일입니다. 따라서 이러한 오픈소스 관리영역도 매우 중요합니다. 공공 프로젝트에서 필수적으로 활용되는 오픈소스에 대한 관리도 꼭 챙겨야 할 부분입니다.

마지막으로, 오픈소스라는 것이 소스코드 공개만 하면 되는 것이 아닙니다. 물론 소스코드가 공개된 것이 오픈소스가 맞으나, 소스코드만 공개된다고 모든 문제가 해결되지는 않습니다. 저절로 소스코드의 품질이 올라가는 것도 아니고, 알아서 개발자들이 개발에 참여하지도 않으며, 사용하고 피드백을 주지도 않습니다. 즉, 소스코드 공개와 함께 개발자들이 쉽게 참여할 수 있도록 친절한 안내와 다양한 설명, 홍보, 밋업, 세미나 등이 필수적이며, 공개하는 기관/기업에서는 개발자들이 관심을 갖을 수 있도록 노력을 아끼지 말아야 합니다. 즉 오픈소스는 이러한 커뮤니티를 통해 유지되고 발전·개선됩니다. 공공 프로젝트 역시, 정부예산으로 커뮤니티를 유지하는 데는 한계가 있습니다. 예산지원이 종료되더라도 프로젝트가 유지되려면 소스코드 공개 시점부터 건강한 커뮤니티가 생성되고 활성화되어 자생력을 갖는것이 매우 중요합니다.

저희 정보통신산업진흥원(NIPA)에서는 과기정통부와 함께 국내 소프트웨어 산업 경쟁력 제고를 위해 기업과 기관의 오픈소스 활용·기여·개발 자양성·커뮤니티 지원 등 다양한 업무를 수행하고 있습니다. 특히 오픈소스 소프트웨어 통합지원센터(OpenUP)을 개소하여 기업과 기관의 오픈소스 이슈, 라이선스 검증·컨설팅을 지원하고 있는데요. 작년 말에는 국가기록원의 기록관리시스템(RMS)에 대한 라이선스 분석을 진행하였고 컨설팅을 수행한 바 있습니다.

이미 오래전에 개발되어, 많은 이해관계 있고, 라이선스 문제로 인해 소스코드 재개발 필요 등 다양한 이슈가 있을 RMS에 대해 오픈소스화(化)하는 것이 현실적으로 당장은 어려울 수도 있습니다. 그러나 국가기록원의 RMS 오픈소스 전략수립 및 소스코드 공개 등에 대해 논의가 시작되었다라는 것이 매우 중요하고, 이를 위해 단계적으로 준비하는 과정이 검토되어야 할 것으로 생각합니다.

기록관리시스템 개발 확산을 위한 소스 오픈 정책 도입전략

한능우(국가기록원)



2021년 제1차 기록관리 정책포럼

주제발표 2

RMS 개발·확산을 위한 소스오픈 정책 도입전략

일 시 : 2021.4.27.(화) 14:00 ~ 16:00

발 표 자 : 한 능 우(국가기록원 디지털기록혁신과장)



RMS 개발·확산을 위한 소스오픈 정책 도입전략

Contents

- I 개요
- II 각 기관·산업계 인식 및 소스 현황 검토
- III 고려 사항 및 대안 분석
- IV 맺음말

I 개요

정책적 논의 배경

기록관리 자율성 강화의 대내외 분위기 확산
RMS 보급 정책의 전환기로 향후 정책방향 설계

소스 오픈을 위한 현황조사

기록관리시스템 사용기관 및 산업계 대상 인식조사
RMS 2.0 소스분석 및 공개SW 라이선스 컨설팅

- ⇒ 소스오픈 전략의 현실화를 위해 기술·제도적 여건과 고려 요인 검토 필요
- ⇒ 세부 대안의 타당성과 실현가능성 검토를 통해 최적의 소스오픈 정책 모색

3 / 19

II

각 기관·산업계 인식 및 소스현황 검토

1. RMS 보급체제 개선 및 소스오픈에 대한 인식
 - 1) 사용기관 인식
 - 2) 산업계 인식
2. RMS 2.0 소스 분석 및 기술성 검토



4 / 19

II-1 RMS 보급체제 개선 및 소스오픈에 대한 인식
사용기관 인식

1) 사용기관 인식

표준 RMS 보급정책 만족도 및 자율개발 의사	
대상	표준 RMS 도입 및 운영 기관 (787개)
기간	2021.1.25. ~ 2021.2.9.
방식	공문 설문조사
응답률	33.3% (공문 수신처 392개 중 130개 응답)

조사항목

- ◆ **표준 RMS 일괄 보급정책 만족도**
 - 현 체제의 기록관리업무 도움정도
 - 향후 보급체제 개선방향
- ◆ **시스템 자율개발 의사 등**
 - 기관별 자율개발 의지 및 예산, 인력 상황
 - 자율개발 기대효과 및 중점 추진정책
- ◆ **기타 서술형 응답**
 - 제도개선 사항 및 기타 개선안 의견

※ 소속기관, 군 기관 및 하급교육행정기관은 각 부처, 국방부 및 교육청에서 수신·전달

II-1 RMS 보급체제 개선 및 소스오픈에 대한 인식
사용기관 인식

1) 사용기관 인식

RMS 개발·보급 체제 개선방향

현 체제 유지
75%

자율개발 전환
2%

현 체제 + 자율개발
22%

기타
1%

〔 현 체제 유지 선택 이유 〕

	이유
1	표준 보급방식인 온나라시스템과의 연계·통일성 고려
2	새로운 시스템 개발이나 유지보수가 어려움
3	기관의 예산 및 인력 부족
4	기관 업무부담이 적어서
5	현 시스템 및 운영환경이 익숙하여 개선 의지가 없음

6 / 19

II-1 RMS 보급체제 개선 및 소스오픈에 대한 인식 사용기관 인식

1) 사용기관 인식

< 기관별 자율개발 및 커스터마이징 의사 >

기관 수	중앙 행정 (군 기관 포함)	지방자치단체		교육청	국공 립대	공공 기관	기타 기관	사립대
		광역	기초					
54	3	6	31	10	2	2	-	-

시스템 자율개발 희망 수준

일부 기능 커스터마이징 희망(53개)
완전 자율개발·운영 의사(1개)

커스터마이징을 원하는 기능

특정 기능(검색 등) 수정 / 기록관리
프로세스 개선 / 기관 고유 기록물 등록

예산 및 인력 상황

충분하지 않음(15개)이 가장 많으며
일부 기관(4개)은 추후 확보 가능에 응답

7 / 19

II-1 RMS 보급체제 개선 및 소스오픈에 대한 인식 사용기관 인식

1) 사용기관 인식

Q. 기록관리시스템 자율개발·운영과 관련하여 국가기록원이 가장 중점을 두어 추진해야 하는 정책은?

순위	정책
1	기관 요청 시 RMS 소스의 적극 제공 (28%)
2	RMS 개발에 대한 교육 및 컨설팅 (23%)
3	관련 규격 등 기준 관리와 준수 여부 점검 (22%)
4	개발된 시스템 인증 (15%)
5	정보화 사업 추진에 대한 교육 및 컨설팅 (11%)

응답 결과

- **현 보급 체제에 대한 만족도 자체는 높음**
 현 체제 유지 및 요구사항 신속 대응(75%)
 현 체제 + 자율개발 허용(22%)
- **기관 자율개발 및 커스터마이징 수요 존재**
 일부 RMS 1.0, 2.0 기관(41%, 54개)
 완전 자율개발(1개) < 커스터마이징(53개)
- **RMS 소스에 대한 적극적 제공 요청 ↑**
 소스 관리와 개방에 대한 기관의 수요가 상당

8 / 19

II-1

RMS 보급체제 개선 및 소스오픈에 대한 인식

사용기관 인식

2)

산업계 인식

산업계 간담회

RMS 개발·유지보수 관련 업체, 전문가 참석(17곳)
시스템 자율개발 관련 산업계 수요,
정책전환을 위한 고려사항 및 지원책 의견수렴

▼

시장 생태계 침체, 다양한 기록관리 수요 반영 곤란으로
RMS 보급체제의 전면적 개선 필요 주장

의견수렴 결과

- **RMS 보급의 자율성 확대에 긍정적으로 인식**
전면 자율개발보다는 현 체제를 유지하면서 자율개발을 허용하자는 현실적 주장 우세
- **RMS 기능개발 사업에 참여사 표시**
일부 수요기관 및 일부 커스터마이징 등
- **소스에 대한 적극적 개방 요청**
중장기적으로 기록시스템 전반에 대한 오픈 소스 정책 수용 검토

9 / 19

II-2

RMS 2.0 소스 분석 및 기술성 검토

RMS 2.0 소스 현황

RMS 2.0 소스 검증 개요 (오픈소스sw 종합지원센터)

- 검증 대상 파일(4871개) 중 공개SW 사용 파일 수(342/7.02%)
- 충돌 라이선스 파일수(111개, 2.28%)

라이선스 컨설팅 결과

- RMS 2.0에 전체 소스코드 공개 등 강한 의무사항을 가지는 오픈소스 사용
- 사용된 오픈소스의 라이선스들이 서로 충돌하는 상황

⇒ 의무사항 위반, 충돌 라이선스에 대한 소스의 삭제·대체 등의 후속 조치 필요

⇒ 기록관리시스템 s/w의 저작권과 라이선스에 대한 명확한 명시 필요

10 / 19

II-2 RMS 2.0 소스 분석 및 기술성 검토

< 참고 > RMS 2.0에서 검출된 주요 오픈소스 라이선스 및 의무사항

의무사항 라이선스	추가조건 부여금지	소스코드 공개의무	특허 라이선스 부여	특허 라이선스 종료	리버스 엔지니어링	DRM 금지	원저작자에 수정물 제공
Apache-2.0			○	○			
CDDL		△ (라이브러리)	○	○			
CPL 1.0		△ (라이브러리)	○	○			
EPL 1.0		△ (라이브러리)	○	○			
GPL v2.0 or later	○	○	○				
GPL v3.0 or later	○	○	○	○	○	○	
LGPL v2.1 or later	○	△ (라이브러리)	○				

11 / 19

III

고려 사항 및 대안 분석

1. 고려 사항

- 1) 기술적 구조 및 저작권 문제
- 2) 소스오픈 정책 수립 범위·방법
- 3) 유관시스템 연계 및 협의 사항

2. 대안 분석

- 1) 표준 RMS 소스의 버전 통합 추진
- 2) 생산시스템과의 연계에 따른 구분
- 3) 오픈소스화 및 소스오픈 정책 수립



12 / 19

Ⅲ-1 고려 사항

1 기술적 구조 및 저작권 문제

기술적 구조

모듈화 설계(구조)

※ 오픈소스에 적합한 S/W의 구조 개발방식

- AMS: 적용 O
- RMS: 적용 X

↓

소스 오픈 O
오픈소스 정책 Δ

RMS의 경우 모듈화 방식으로 설계되지 않아
오픈소스의 확장적 적용이 어려울 수 있음

※ 소스 오픈에 대한 준비 필요(구조)

저작권 문제

기존: 매년 다른 사업자가 들어와서 개발

2016년	c-RMS	C + 기록원
2013년	RMS2.0	B + 기록원
2006년	RMS1.0	A + 기록원

신규 프로젝트: 오픈조건 계약에 명시 X

↓

별도로 법적고려 필요

13 / 19

Ⅲ-1 고려 사항

2 소스오픈 정책 수립 범위방법

소스오픈 정책

기록관리시스템

- 기존: RMS 1.0, 2.0, c-RMS
- 신규: 차세대 c-RMS

↓

상이한 버전의 기록관리시스템 체제를 고려하여
소스오픈을 어떻게 구현하고 확장시킬 것인가에 대한
맥락적인 문제

커뮤니티 유지·활성화 측면

- 중앙기록물관리기관의 역할과 개입 수준
기록관리시스템 개발 환경의 특수성
(적은 참여업체, 작은 사업규모 고려)
- 커뮤니티의 핵심 참여자별 가용자원 상이
기관 및 업체별로 예산, 인력, 인프라 등이
다른 현실 고려

14 / 19

Ⅲ-1 고려 사항

3 유관시스템 연계 및 협의사항

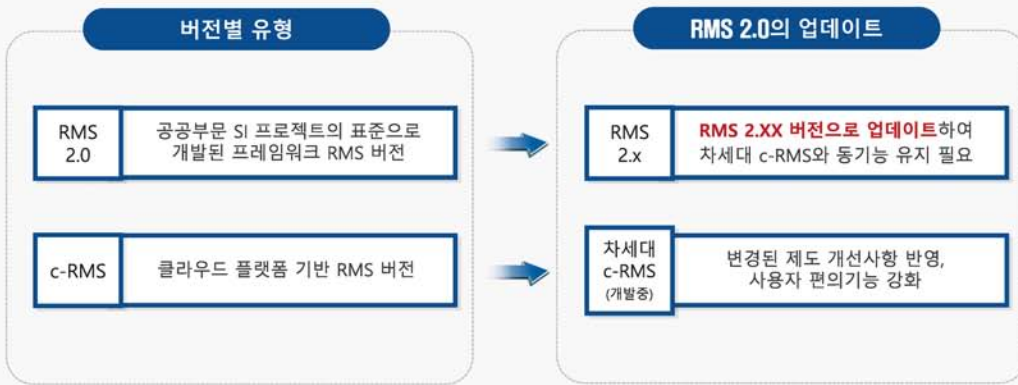


※ 지자체를 포함하는 온나라시스템의 확산정책이 RMS의 보급 정책에 큰 영향을 미치는 구조

15 / 19

Ⅲ-2 대안 분석

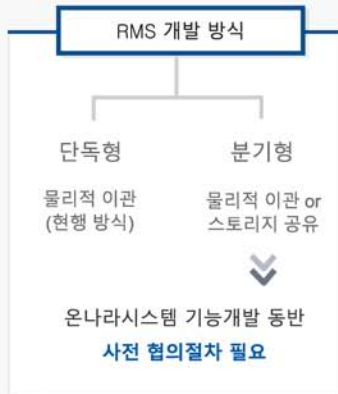
1 표준 RMS 소스의 버전 통합 추진



16 / 19

Ⅲ-2 대안 분석

2 생산시스템과의 연계에 따른 구분



지자체의 개발 환경(클라우드, 독립)에 따라 시스템 개발 방식도 상이

단독형: 온나라기록물을 물리적으로 이관받는 방식의 개발 모델(현행 RMS)

분기형: 온나라기록물의 물리적 이관 or 스토리지 공유 중 선택가능한 개발 모델(온나라)

RMS 소스 오픈 시

각 개발 환경에 대응하여 RMS의 효율적 개발이 가능하도록

온나라시스템의 API 개방, DB 접근 등 정책적 공조, 협의 절차가 요구됨

17 / 19

Ⅲ-2 대안 분석

3 오픈소스화 및 소스오픈 정책 수립

오픈 소스화 및 소스오픈 정책 수립

- 단계적·부분적 소스 오픈 추진
현실적 여건을 고려하여 소스에 대한 일회적 개방을 시작으로 장기적인 자율화 지향 정책 수립
- 저작권, 오픈소스 라이선스 명시
S/W 사업 발주 시 저작권 및 라이선스에 대한 명확한 기재
- 충분한 환경 분석 및 수요 조사
적극적 지원정책이 요구될 만큼의 개발환경이 구축될 것인가에 대한 충분한 사전 수요조사 필요

18 / 19

IV 맺음말

소스 공개의 의미 재고

- 현재 정책상에서도 개별기관의 커스터마이징 목적의 경우 → RMS 소스 제공이 가능 ※ 보안서약 작성 등 전제
- RMS 관련 사업에 기참여한 업체는 이미 RMS 소스를 가지고 있을 가능성
- 소스오픈과는 별개로 일부 산업계에서 자체 RMS 솔루션 개발, 사업화 진행 중

오픈 소스 체제로의 이행

- 선언적 의미의 소스 오픈 → 자율화 지향의 정책적 선언을 기점으로 점차 자율화의 범위와 대상을 확대
- 장기적으로 개방, 자율, 협력의 가치가 실질적으로 실현되는 단계로 나아가는 것이 바람직

RMS 개발확산을 위한 소스오픈 정책 도입전략

안 대 진(아카이브랩)

RMS 개발확산을 위한 소스오픈 정책 도입전략

안 대 진(아카이브랩)

1. 개요

1) 오픈소스의 의미와 전략적 이점

개발업체나 기관 담당자들을 만나 이야기해 보면 오픈소스소프트웨어(이하 OSS)에 대한 인식이나 이해의 수준이 다름을 알 수 있다. “OSS는 보안에 취약하다”, “버그가 많고 완성도가 떨어진다” 등의 오해가 있는가 하면, OSS 라이선스의 의무사항에 대한 인식은 매우 부족하다. 이는 논의를 진전시키는 데 걸림돌이 된다. 이유를 설명하기 귀찮을 만큼 오픈소스는 이미 420대세이고, 지난 수십년 동안 소프트웨어 업계에서 투자 대비 효과가 증명되었다. 오픈소스의 전략적 이점은 빨리 좋은 소프트웨어를 만드는 것이다. 표준RMS는 매우 오랜기간 동안 정체되어 있다. 오픈소스는 이를 극복할 수 있는 유력한 대안이 될 수 있다. 오해나 우려보다는 기록시스템의 오픈소스 전략을 실행하기 위한 발전적 논의를 하고자 한다.

2) 문제 진단

기록연구사 대상 만족도 조사 결과는 표준RMS 정책의 문제를 적나라하게 드러내고 있다. 국가기록원의 일괄 보급 방식에 대해 94%가 만족한다는 응답은 해석이 필요하다. 기록연구사들은 좋은 기록시스템을 써 본 적이 없고 부처 내 입지가 약하다. 그리고 연구사가 속한 조직은 현 체제를 바꾸길 원하지 않는다. 이들이 처한 입장이나 인식 수준에서는 표준RMS 정책의 문제를 제대로 파악하기가 쉽지 않을 수 있다. 수년 간 만나본 많은 기록연구사들은 표준RMS를 거의 사용하지 않는다고 했다. 별로 쓸 데가 없고 성능도 맘에

안 든다는 의견이 많았다. 표준RMS의 진짜 문제는 너무 구식이라는 것이다. 토론을 준비하며 표준RMS의 소스코드를 공개해도 쓸 의사가 없다는 업계의 의견을 청취했다. 이러한 상황을 종합해 보면 현 시점은 RMS 2.0의 소스를 공개하기 위한 노력보다는 새로운 구조와 기능을 가진 차세대 시스템을 구상해야 할 때인 것 같다.

2. 제안

1) 표준RMS 새로 만들자

업계에서는 표준 RMS의 설계가 2006년 이후 크게 변경되지 않았다고 한다. 그동안 업무환경과 사람은 크게 바뀌었다. 공공기록물관리법이 수차례 개정되고 결재문서 이외의 다양한 기록유형을 통합 관리할 수 있는 자동화된 시스템이 필요하다는 요구가 누적되어 왔다. 이미 800개 가까운 기관이 표준RMS를 도입했고 여기에 투입된 예산을 감안하면 이를 바꾸기란 쉽지 않을 것이다. 하지만 현재의 표준RMS를 구조부터 바꾸지 않는 한 기록연구사들의 외면은 계속될 것이고 기록관리 업무는 개선되기 힘들다. HP, IBM, Opentext 등 해외 벤더의 상용 RMS와 표준RMS의 수준 차이는 비교하기 힘들 정도이다. 정체된 표준RMS가 기록연구사들의 업무 수준을 낮추고 좋지 않은 경험을 반복 확산시키고 있다. 새로운 RMS를 만들기 위한 ISP부터 시작하길 제안한다.

2) 오픈소스 전략을 실행하자

a. 기술적 구조와 라이선스

- (구조) OSS 프로젝트가 성공하려면 다수의 개발자가 쉽게 소스코드를 이해하고 개작할 수 있어야 한다. 이를 위해 컴포넌트 기반의 설계와 개발이 필수적이다. 모듈화, MVC(모델, 뷰, 컨트롤러) 디자인 패턴 등이다.
- (라이선스) OSS 라이선스 중 GPL 계열(GPLv2, GPLv3, AGPL)과 퍼미시브 라이선스(Apache-2.0, MIT, BSD 등)를 용도에 맞게 활용하는 방안을 제안한다. GPL은 강한 카피레프트 라이선스로 해당 소스코드를 개작하면 결과물을 커뮤니티에 동일한 라이선스로 공개해야 한다. 해외의 아카이브시스템 중 AtoM, Omeka, RODA 등 기억기관이 주도하는 OSS 프로젝트에 주로 적용된다. 아카이브나 도서관 등 커뮤니티 구성원들의 다양한 개발 결과물이 집단지성을 발휘하여 소프트웨어의 발전 속도가 매우 빨라진다. 퍼미시브 라이선스는 상업적 이용을 허용하므로 유능한

개발업체의 참여를 유도할 수 있고, 보다 광범위하게 활용되는 경향이 있다.

- (패키징) 기록시스템 패키지 전체를 오픈소스화하는 것보다는 의미 있는 단위로 분할하여 개별 프로젝트로 발전하도록 하는 것이 효과적이다. 예를 들어 메타데이터를 추출하는 툴, 보존포맷을 생성하는 툴, 기록물 뷰어 등으로 구분하는 것이다. 이러한 단위 프로젝트는 표준RMS와 표준AMS가 공통으로 사용할 수 있어 더욱 현실적이다.
- (배포) 다양한 설치 방법을 제공하여 쉽게 테스트해볼 수 있도록 하는 게 좋다. ZIP 설치파일, 깃허브 클론, 도커 등이 있다.
- (표준AMS 모듈 재사용) 표준AMS가 작년, 올해 개발되고 있다. 표준AMS는 철건 구조를 탈피한 DB 구조를 적용했다고 한다. 이는 대단한 시작이며 큰 변화다. 모듈화된 구조는 오픈소스로 배포하는 데도 훨씬 유리하다. 실제로 국제 기록커뮤니티에서는 Archivematica, e-Ark 등 AMS의 전체 패키지를 오픈소스 프로젝트로 공개한 경우가 많다. 표준AMS의 모듈은 향후 차세대 표준RMS의 모듈로 재사용될 수 있다. 이를 감안하여 사업이 추진되면 좋겠다.

b. 소스공개 절차와 방법

OSS 프로젝트가 아닌 경우 오픈소스화하기 위한 준비작업이 매우 많다. 기록시스템 개발 단계부터 소스코드 공개를 염두에 두고 모듈화, 명료한 주석처리 등이 요구된다. 코드 정제, 배포용 코드 재작성, 유지보수 생산성 향상을 위한 소스코드리팩토링, 라이선스 파일 준비, 문서화, 깃허브 리포지터리 만들기 등 많이 알려진방식들을 참고하길 바란다.

c. 커뮤니티, 비즈니스모델, 거버넌스 작동 방식

국가기록원 깃허브에 표준RMS와 표준AMS에 공통으로 사용할 소프트웨어들이 공개되었다고 상상해 보자. 그중 쓸만한 것들은 표준RMS 벤더들이 재사용할 수 있다. GPL 라이선스가 적용된 ‘SIP Creator’를 서울기록원이 가져가 메타데이터 부분을 변경한 후 결과물을 다시 동일한 라이선스로 공개한다. MIT 라이선스가 적용된 기록물 뷰어는 기록 벤더들이 가져가 자사의 솔루션에 적용하여 판매할 수 있다. 이러한 과정이 반복되며 기록시스템 생태계가 형성된다. OSS의 커뮤니티, 비즈니스모델 거버넌스 작동방식 또한 다양한 레퍼런스를 참조하길 바란다.

d. 기록시스템 생태계 조성

최근 문체부와 더불어민주당 이병훈(문화체육관광위원회, 광주동남구을) 의원실주최, 한국국학진흥원, 한국학호남진흥원, 충남역사문화연구원 주관으로 ‘민간기록문화의 보존 및 활용을 위한 법률’의 제정이 논의되고 있다.

1945년 이전 민간기록문화를 보존한다는 취지이다. 이 내용을 접하고 위기의식을 느꼈다. ‘민간기록문화’는 우리의 키워드이고 차세대 연구(2017)에서도 가칭 아카이브 진흥법에 대한 필요성을 강조했다. 2010년 이후 일상과 커뮤니티, 마을기록 등 민간기록과 관련된 연구가 이어지고 오픈소스 소프트웨어인 오메카(Omeka)가 보급되며 시민사회 영역에서 디지털 아카이브가 높은 인기를 구가하고 있다. 이러한 시점이라서 놀라움은 더 커진다. 우리는 1945년 이전 기록, 민간기록문화, 데이터 라는 키워드를 놓쳤다. 국가기록원은 ‘민간기록문화’란 키워드를 되찾기 위해 기록시스템의 오픈소스 전략을 효과적으로 활용하길 바란다. 표준AMS의 배포대상에 민간영역을 포함하는 등 공공기록관리 바깥의 기관들이 국가기록원의 소프트웨어를 같이 사용하도록 유도하는 것이다. 국가기록원이 시스템을 지원해 준다기 보다는 기록생태계를 조성하는 데 주도적인 역할을 하여 상생을 도모하는 전략이다. 개발업체들은 커뮤니티의 핵심멤버로서 이들을 위한 구체적인 방안이 강구되어야 한다. 오픈소스란 키워드로 논의의 장이 열린 것은 좋은 시작이다. 표준RMS정책의 재정비를 위해 앞으로 실효적인 문제 진단과 논의가 지속되길 바란다.

<참고자료>

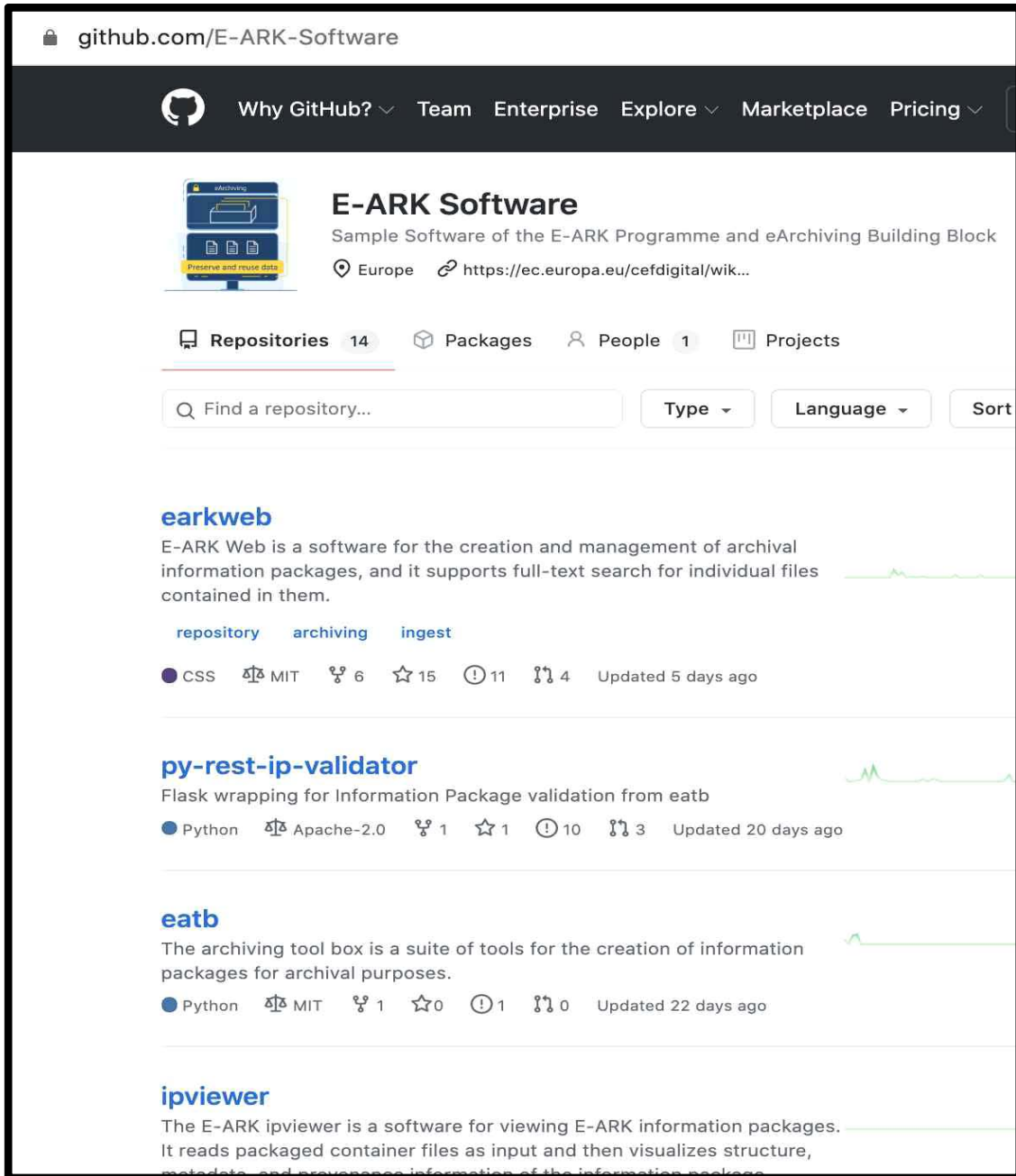


그림 1 E-Ark 소프트웨어 깃허브

전자기록 장기보존에 필요한 14개의 툴로 구분하여 제공

- earkweb : AIP를 생성하고 본문검색하는 소프트웨어(CSS)
- py-rest-ip-validator : 보존패키지 검증도구(Python)
- eatb : AIP 생성도구 (Javascript)
- ipviewer : 보존패키지 뷰어(Java)