

오픈소스 4.0

- 협력과 경쟁을 위한 혁신의 도구 -

김성민·홍아름·최새솔·연승준



본 보고서는 ETRI 기술정책연구본부 주요사업인 “국가 지능화 기술정책 및 표준화 연구”를 통해 작성된 결과물입니다.





핵심 요약	1
I. 오픈소스 개요	5
1. 오픈소스 개념 및 조건	5
2. 오픈소스 라이선스	7
3. 오픈소스 생태계의 현재	9
II. 오픈소스 진화단계 특성	11
1. 분석 배경 및 개요	11
2. 오픈소스 1.0	14
3. 오픈소스 2.0	17
4. 오픈소스 3.0	21
5. 오픈소스 4.0	25
III. 오픈소스, 협력과 경쟁의 진화	30
1. 오픈소스 프로젝트 성장경로 유형	30
2. 저장소의 발전과 협력 방식의 진화	38
3. 오픈소스 생태계 경쟁 구도의 변화	43
IV. 결론 및 시사점	49
참고문헌	55

핵심 요약



연구의 필요성 및 목적

- 오늘날 오픈소스는 소프트웨어 산업을 넘어 ICT 산업에서 보편적 개발방식으로 자리매김하고 있을 뿐 아니라, 기술·산업 혁신의 촉진제로서 역할을 담당함
- 오픈소스가 어떠한 배경과 방식으로 전개됐으며, 그 영향은 무엇인가를 파악하는 것은 오픈소스의 발전 전망과 이해에 필수적임
- 따라서 오픈소스를 단순히 협력개발 기반 소프트웨어로 한정하는 것은 오픈소스가 지닌 ICT 생태계에서의 영향력을 충분히 이해하는데 부족하며, 변화된 오픈소스 생태계의 특성에 따라 인식과 대응 방향 또한 변화되어야 할 것으로 판단됨
- 이에 본 연구는 오픈소스 개요와 현황을 파악하고, 오픈소스 진화단계 특성과 협력 및 경쟁 진화과정을 분석하며, 이에 대한 시사점을 도출함



오픈소스 개요

- (개념) 오픈소스 소프트웨어(Open Source Software, 이하 오픈소스SW)는 프로그래밍 언어로 나타난 설계도인 소스코드를 공개하여 다른 사람들이 이를 볼 수 있으며, 지정된 라이선스 방식에 따라 배포, 수정, 복제, 사용, 재배포를 할 수 있는 SW
 - 오픈소스는 코드에 대한 정보를 다른 개발자가 이해할 수 있는 소스코드로 제공하기 때문에 이를 이해할 수 있는 사용자는 스스로 수정하거나 업그레이드할 수 있음
- (오픈소스 현황) 오픈소스 개발이 증가하고 관련 서비스 시장이 활성화되는 등 오픈소스 생태계는 지속적으로 성장하고 있음
 - 전세계 오픈소스 서비스 시장*은 2022년까지 329억 달러로 성장할 것으로 예상됨
 - ※ 오픈소스 서비스: 오픈소스 SW 관련 지원, 유지보수 및 관리, 구현, 컨설팅 서비스 등을 의미
 - 세계 최대 오픈소스 개발 플랫폼인 깃허브에는 2019년 9월 말 기준 70여 개국의 4천만명 이상의 개발자(사용자 계정수)가 활동하고 있음



오픈소스 진화단계 특성

- 오픈소스와 관련된 연도별 주요 사건 및 생태계를 움직이는 주요항목 중심으로 태동기 부터 현재까지 특징적 시기로 구분하고, 시기별 특징과 진화 양태를 분석을 진행
 - (오픈소스 1.0-태동기) SW 사유화에 저항한 개발자의 자발적 참여로 시작
 - (오픈소스 2.0-확립기) 오픈소스 라이선스의 분화, 프로젝트 커뮤니티의 조직화, 수익화 모델 등 오픈소스의 기틀이 확립
 - (오픈소스 3.0-촉진기) 오픈소스 생태계 자생력의 확보, 일부 기술·시장 우위를 증명
 - (오픈소스 4.0-성장기) 인공지능, IoT, 자율차 등 기술발전이 빠르게 전개되는 산업 영역에서 기술·산업 혁신을 촉진하는 보편적 개발방식으로 자리매김

		오픈소스 진화 단계 특징			
구분		오픈소스 1.0 (태동기)	오픈소스 2.0 (확립기)	오픈소스 3.0 (촉진기)	오픈소스 4.0 (성장기)
시기	구분	GNU선언 이후 (1983~1997)	OSI 창설 이후 (1998~2005)	깃 공개 이후 (2006~2014)	텐서플로우 공개 이후 (2015~ 현재)
	특징	<ul style="list-style-type: none"> • SW상업화에 저항한 자발적 참여운동 	<ul style="list-style-type: none"> • 라이선스 분화 • 커뮤니티 조직화 	<ul style="list-style-type: none"> • 생태계 자생력 확보 • 일부 기술시장 우위 	<ul style="list-style-type: none"> • 빅테크 기업 주도 • 혁신 도구화
변화 동인	기술환경	PC 대중화	인터넷 성장	모바일, 클라우드 확산	인공지능, 데이터 경제 확산
	시장환경	상용 소프트웨어 시장 성장	웹 서버 시장 성장	SW산업 패러다임 전환, 신규 개발 수요 폭증	디지털전환(x+AI) 수요 성장
전개과정		<ul style="list-style-type: none"> • PC대중화-SW상업화, 해커문화위축-개발자 저항운동 • 리눅스공개→자유 SW운동 동력확보 	<ul style="list-style-type: none"> • 라이선스정립→기업 상업적 활용 촉진 • 커뮤니티조직화-지속 가능성 강화 • 상용화 라이선스 등 오픈소스 외형 확립 	<ul style="list-style-type: none"> • 신규개발수요, 빅테크 기업 오픈소스 참가 증대→오픈소스진영 신규개발자면증가 • 오픈소스 개발자 생태계 자생력 확보 • 오픈소스 기술/시장 우위 증명 	<ul style="list-style-type: none"> • 빅테크기업의 시드 기술제공, 커뮤니티 운영 등 적극 참여 • 오픈소스의 ICT산업내 영향력증대 • 다양한 분야로 오픈소스 방식 확대
주요 변화 내용	핵심 참여자	개발자 커뮤니티	재단	재단 및 빅테크 기업	빅테크 기업
	대표 프로젝트	리눅스	LAMP*	안드로이드, 도커, 오픈스택, 하둡 등	텐서플로우, 파이토치, 쿠버네틱스, 이폴로 등
	프로젝트 성격 및 수준	유닉스계열 모방, 대체로 열위	일부대등, 틈새시장	일부 선도 & 시장표준	기술·산업 혁신선도, 보편적 개발방식
	비즈니스 모델	<ul style="list-style-type: none"> • 의도적 수익모델 없음 • 배분사업등 일부개시 	<ul style="list-style-type: none"> • 지원서비스, SW판매 등 다양한 모델출현 	<ul style="list-style-type: none"> • 기업 재단후원 대형화 • 플랫폼기반 수익모델 등장(모바일클라우드) 	<ul style="list-style-type: none"> • 빅테크기업의 플랫폼 수익모델 가속화

* 출처: 저자 작성, LAMP: Linux, Apache, MySQL, PHP



오픈소스, 협력과 경쟁의 진화

- (협력의 대상 - 개발자, 기업, 재단의 선택적 협력) 대표적인 오픈소스 프로젝트의 성장경로를 분석해본 결과, 오픈소스 프로젝트의 목적성과 공개 주체의 목표에 따라 기업, 재단 등 다양한 주체와의 협력 관계를 통해 성장해온 것을 볼 수 있었음
 - 개발자 및 학계 프로젝트는 프로젝트의 목적에 따라 재단 또는 기업 등의 큰 주체의 지원을 받아 성장하고, 기업의 프로젝트는 기업의 전략적인 방향성에 따라 직접 관리, Alliance 구성, 재단 설립 등의 형태로 성장하는 모습을 보임
- (협력 기술/방식 - 저장소의 진화와 협업의 심화) 오픈소스 저장소는 온라인 기반 분산 버전관리시스템으로 진화하고 깃허브가 글로벌 오픈소스 생태계의 중심으로 성장
 - 깃허브 등 오픈소스 저장소는 ① 개방성 ② 투명성 ③ 용이성 ④ 분산과 통제의 균형 ⑤ 네트워크 외부성 등의 특징으로 오픈소스 생태계 성장을 견인함
- (경쟁 구도- 저항세력에서 시장의 주역으로) 오픈소스 주요 기술들의 협력 및 경쟁 구도에 대한 사례를 분석해본 결과, 시대별&기술별로 협력 대상과 경쟁구도가 변화함
 - (협력 주체의 변화) 오픈소스 1.0 시대에는 개발자들이 협력의 주체이자 중심이었으나, 오픈소스 4.0 시대에는 기업들이 협력의 중심에서 오픈소스에 대한 직접지원 확대해감
 - (경쟁구도) 초기에는 오픈소스 진영과 거대 상용SW기업 간의 대결 구도였으나, 오픈소스 진영에 대기업들이 적극 참여하면서 오픈소스 기술이 시장표준으로 자리매김하거나 오픈소스 프로젝트 진영간 경쟁이 형성되는 경우가 많음



시사점

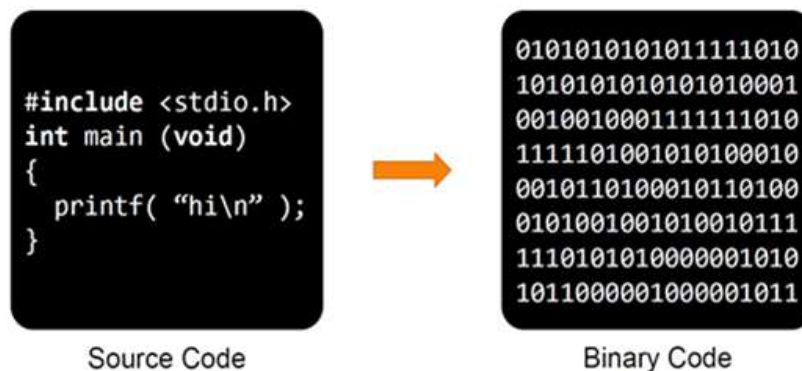
- 오늘날, 오픈소스는 일부 개발자들의 소프트웨어 개발방식이 아니라, 인공지능, 클라우드, 자율주행차 등 4차산업 혁명시대의 ICT 핵심 및 융합 기술 분야의 주된 기술 개발 방식이자 생태계 경쟁 방식임
- 오픈소스 생태계 활성화를 위해서는 최근의 오픈소스 3.0과 4.0 시대의 특성을 고려하여 (1) 오픈소스 프로젝트 성장을 위한 협력 전략 및 지원책 마련, (2) 재단 및 기업 등 오픈소스 투자 주체 유인 방안, (3) 대기업-중소기업, 해외기업-국내기업의 공정하고 안전한 협력을 위한 지원 정책 마련, (4) 제4차 산업혁명 시대 기술력 확보를 위한 국가 R&D 전략으로서의 오픈소스 전략 마련이 필요함을 제안함

I | 오픈소스 개요

1 오픈소스 개념 및 조건

- (개념) 오픈소스 소프트웨어(Open Source Software, 이하 오픈소스SW)는 프로그래밍 언어로 나타낸 설계도인 소스코드를 공개하여 다른 사람들이 이를 볼 수 있으며, 지정된 라이선스 방식에 따라 배포, 수정, 복제, 사용, 재배포를 할 수 있는 SW (이진희, 2019 수정)
 - 상용 SW 저작권자는 저작물 코드를 공개하지 않고 0과 1로 구성된 바이너리 코드를 제공하기 때문에 코드 판독이 불가능하며 개발업체만 버그 수정과 업그레이드가 가능함
 - 반면, 오픈소스는 코드에 대한 정보를 다른 개발자가 이해할 수 있는 소스코드로 제공하기 때문에 이를 이해할 수 있는 사용자는 스스로 수정하거나 업그레이드할 수 있음

그림 1-1 소스코드와 바이너리코드 예시



※ 이미지 출처: 슈어소프트테크 블로그 (2017), 바이너리 분석 기술.

- (OSI의 정의) 오픈소스에서 가장 권위있는 기관인 OSI(Open Source Initiative)에서는 오픈소스는 단순히 소스코드에 대한 접근을 의미하는 것이 아니라 다음의 10가지 배포 조건을 준수해야 한다고 함¹⁾
 - ① 자유로운 재배포 (Free Redistribution) : 여러 소스의 프로그램을 포함하는 통합 소프트웨어의 구성요소인 소프트웨어들을 판매하거나 재배포하는 것을 제한하지 않으며, 이러한 판매에 대해 로열티나 수수료를 요구하지 않아야 함

1) HHMONIKR의 오픈소스 정의에 대한 해설을 참조하여 OSI 'Open Source Definition'의 내용 정리
<https://hamonikr.org/oss/51202>, <https://opensource.org/osd>



- ① 소스 코드 공개 (Source Code) : 프로그램은 소스코드를 포함해야 하며, 어떤 제품이 소스 코드와 함께 배포되지 않은 경우, 합리적인 복제비용 이하로 소스코드를 얻을 수 있는 수단이 있어야 함. 소스코드는 프로그래머가 프로그램을 수정할 때 선호하는 형식이어야 하며 의도적으로 난독화된 소스코드는 허용되지 않음
- ② 파생된(2차적) 저작물 허용 (Derived Works) : 라이선스는 수정 및 파생된 저작물을 허용해야 하며, 원본 소프트웨어의 라이선스와 동일한 조건에 따라 배포해야 함
- ② 저자의 원시코드 원형 유지(Integrity of The Author's Source Code) : 바이너리 코드를 생성할 시점에 프로그램을 수정할 목적으로 패치파일의 배포를 허용한 경우, 변경된 원시코드의 배포를 제한할 수 있음. 그러나 변경된 원시코드가 아닌 그것을 통해 만들어진 결과물의 배포는 허용해야 함. 파생저작물에 대해서는 원본 오픈소스와 다른 이름이나 버전을 적용하도록 할 수 있음²⁾
- ③ 개인이나 단체에 대한 차별 금지 (No Discrimination Against Persons or Groups) : 특정 개인이나 단체를 차별하지 말아야 함
- ③ 사용 분야에 대한 차별 금지 (No Discrimination Against Fields of Endeavor) : 특정 분야에 사용되는 것을 금지하면 안됨. 예를 들어 프로그램이 비즈니스에서 사용되거나 유전자 연구에 사용되는 것을 제한할 수 없음
- ④ 라이선스의 배포 (Distribution of License) : 프로그램에 첨부된 권한은 해당 당사자가 추가 라이선스를 실행할 필요 없이 프로그램을 재배포하는 모든 사람에게 적용되어야 함
- ⑤ 특정 제품에만 유효한 사용 허가의 금지 (License must not be specific to a product) : 프로그램에 첨부된 권리는 특정한 소프트웨어의 배포에 대해 한정되어서는 안되고, 프로그램을 재배포하는 모든 당사자는 원래 소프트웨어 배포와 함께 부여된 것과 동일한 권한을 가져야 함
- ⑥ 다른 소프트웨어에 대한 제한 금지 (License must not contaminate other software) : 라이선스 소프트웨어와 함께 배포되는 다른 소프트웨어에 제한을 두어서는 안됨. 예를 들어 동일한 매체에 배포되는 다른 모든 프로그램이 오픈소스 SW여야 한다고 주장해서는 안됨
- ⑦ 라이선스의 기술적 중립성 (License must be Technology-Neutral) : 개별 기술 또는 인터페이스 스타일에 대한 라이선스 조항은 허용되지 않음

2) 이러한 조치를 통해 소스코드의 원작자와 수정인의 정보를 구분함으로써 사용자가 피드백이 필요한 경우 누구에게 문의해야할지 분명하게 알 수 있음(<https://hamonikr.org/oss/51202>)

2 오픈소스 라이선스

- 현재 소프트웨어는 저작권, 특허권, 상표권, 영업비밀 등의 지식재산권에 의해 보호를 받으며, 이들은 저작권자만 사용할 수 있으나, ‘라이선스’는 권리자가 다른 사람에게 일정한 조건으로 특정 행위를 할 수 있는 권한을 부여하는 것임³⁾
 - 대부분의 오픈소스 라이선스는 개발자 및 기여자에 대한 정보를 표시하고, 코드를 수정한 경우 수정한 정보를 표시해야 하며, 라이선스 정보를 제공해야 함
- 오픈소스는 공개되어 있고, ‘무료’로 사용할 수 있지만, 라이선스에 정의된 사용방법 및 의무사항을 따라야 하며, 이를 위반할 경우 라이선스 위반 및 저작권 침해로 법적 책임을 지고 경제적 손해를 입을 수 있음
 - 소스코드의 사용, 생산, 수정 및 배포 등에 관한 규정으로 소스코드 공개를 강제하기 보다는 오픈소스의 사유화 방지 목적이 강함
- 오픈소스 이니셔티브(OSI)에서 공인된 라이선스는 82개, 소프트웨어 패키지 데이터 익스체인지(SPDX)에서 다루는 라이선스는 350개, 블랙덕 날리지베이스에는 2,600여 개의 라이선스가 쓰이는 것으로 파악됨⁴⁾
 - 가장 많이 쓰이는 라이선스 20개가 실제 사용되는 오픈소스의 98%를 커버함
 - 유명하지 않은 라이선스라도 해당코드 작성자가 적용한 라이선스를 따라야 함
 - 오픈소스 활용이 높아지면서 라이선스가 복잡하여 법적 분쟁과 소송의 우려가 높고, 약성코드로 보안의 취약성도 높을 수 있음
- 법적 관점에서 라이선스가 이용허락 여부 또는 2차적 저작물의 문제에 대해서 어떻게 표현하고 있는가와 관련하여 허용적(Permissive) 라이선스와 제한적(Restrictive) 또는 카피레프트(copyleft) 라이선스로 분류⁵⁾
 - 제한적 또는 카피레프트 라이선스는 사용하고 수정하였을 때 변경된 소스코드를 모두 공개하여 재배포해야하는 경우로 GPL(General Public License)이 대표적인
 - ※ GPL하에서 오픈소스를 사용하거나 그것을 기반으로 제작된 모든 SW는 다시 오픈소스가 되는 순환적인 구조를 갖게 되는데, 이를 바이러스 효과(virus effects)라고 함(이진희, 2019)

3) ETRI 오픈소스 포털의 오픈소스SW 라이선스 소개 참조

4) Synopsys (2020), 2020 오픈소스 시큐리티&리스크 분석보고서(한글본).

5) WhiteSource, ‘Open Source License Comparison,’

<https://resources.whitesourcesoftware.com/blog-whitesource/open-source-licenses-comparison-guide>



- 허용적 라이선스는 사용에 별다른 요구사항을 부여하지 않고 광범위한 권한을 부여하는 아파치(Apache)나 MIT계열의 라이선스로 상업적 용도로 사용이 용이함
- 근래에는 GPL보다는 오픈소스로 개발한 이후 상용화에 용이한 MIT, 아파치 등의 허용적 라이선스가 많이 쓰이는 추세임

표 1-1 주요 오픈소스 라이선스 비교

구분	무료 이용가능	배포 허용가능	소스코드 취득가능	소스코드 수정가능	2차적 저작물 재공개 의무	독점SW와 결합가능
GPL	O	O	O	O	O	X
LGPL	O	O	O	O	O	O
MPL	O	O	O	O	O	O
BSD License	O	O	O	O	X	O
Apache License	O	O	O	O	X	O

※ 출처: NIPA 오픈소스 소프트웨어 통합지원센터 홈페이지 (<https://www.oss.kr>)

- 소스코드 공개 여부와 무료/유료 라이선스는 다른 차원으로, 유료 소프트웨어지만 소스 코드를 공개하는 예도 있고, 무료 소프트웨어지만 소스코드를 공개하지 않는 때도 있음

표 1-2 소프트웨어 공개여부와 유료/무료 라이선스

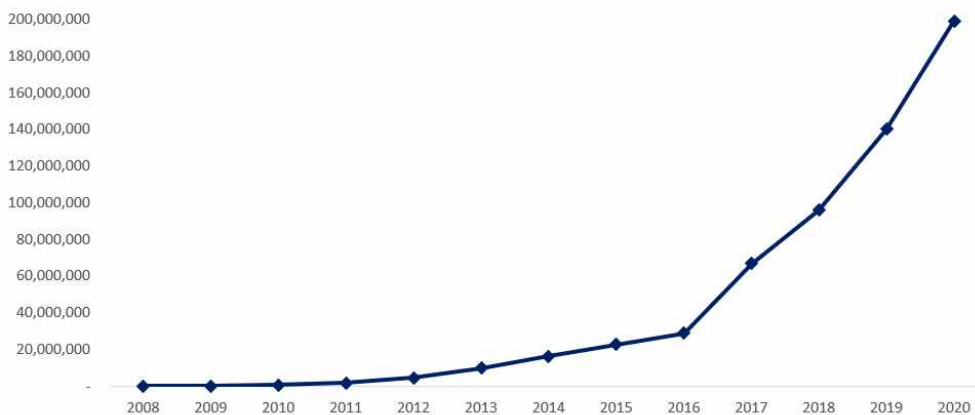
구분	무료 라이선스	유료 라이선스
공개	대부분의 오픈소스	일부 배포한 오픈소스 (Red Hat, MySQL)
비공개	Freeware(Winzip, RealAudio) Shareware(초기에만 무료) Adware(JetAudio)	상용 비공개 소프트웨어 (MS Office, Oracle, Shareware)

※ 출처: NIPA 오픈소스 소프트웨어 통합지원센터 홈페이지 (<https://www.oss.kr>)

3 오픈소스 생태계의 현재

- 전 세계 SW 개발자들은 분산형 협업을 위한 다양한 기능을 제공하는 깃허브, 깃랩 등 오픈소스 코드 저장소를 활용하여 소스코드를 공개하고 협업하여 SW를 개발
- 오픈소스 개발이 증가하고 관련 서비스 시장이 활성화되는 등 오픈소스 생태계는 지속적으로 성장하고 있음
 - 전세계 오픈소스 서비스 시장*은 2022년까지 329억 달러로 성장할 것으로 예상됨⁶⁾
 - ※ 오픈소스 서비스: 오픈소스 SW 관련 지원, 유지보수 및 관리, 구현, 컨설팅 서비스 등을 의미
- 세계 최대 오픈소스 개발 플랫폼인 깃허브에는 2019년 9월 말 기준 70여 개국의 4천만 명 이상의 개발자(사용자 계정수)가 활동하고 있음⁷⁾
 - 2019년 깃허브에는 1천만 개의 새로운 계정이 개설되었으며, 4,400만 개 이상의 신규 저장소가 생성됨
 - 깃허브 내 저장소(Repository)는 2013년에 약 1,000만 개⁸⁾였으나, 2018년에는 1억 개를 돌파⁹⁾하여 2020년 11월 기준 약 1.9억 개의 저장소가 있는 등¹⁰⁾ 급격히 증가
 - 깃허브 내 개발자 수도 지속해서 증가하여 2017년 2,400만 명에서 2019년 4,000만 명으로 증가함¹¹⁾

그림 1-2 깃허브 저장소 추이¹²⁾



※ 출처: 연도별 GitHub Octoverse, GitHub 블로그를 참고하여 작성

6) <https://www.marketsandmarkets.com/PressReleases/open-source-services.asp>

7) GitHub Octoverse 2019를 참고하여 작성 (<http://octoverse.github.com/2019>)

8) <https://github.blog/2013-12-23-10-million-repositories/>

9) <https://github.blog/2018-11-08-100m-repos/>

10) <https://github.com/search?q=&type=>

11) GitHub Octoverse 2017년, 2019년 참고하여 작성

12) 2014~2015년은 추정값



- 평균 41개 국가나 지역에 사는 개발자가 깃허브에 오픈소스 프로젝트를 구축하였고, 깃허브 사용자의 80% 이상은 미국 이외 국가에서 사용하며, 중국, 인도, 독일, 영국, 일본 등의 개발자 비율이 높음

※ 텐서플로우는 깃허브에서 가장 인기 있는 프로젝트 중 하나, 2018년 10월부터 2019년 9월 사이 참가자 2,200명이 개발에 참여하였으며, 텐서플로우와 종속 프로젝트 참가자를 모두 포함하면 2만 5천 명이 개발에 기여했다고 할 수 있음

- 산업분야를 구분하여 볼 때도 다양한 산업에서 오픈소스 사용이 확대되고 있음¹³⁾
 - 시놉시스(2020)년 보고서에 따르면 17개 산업 분야 코드베이스 중 99%가 오픈소스를 포함하고 있으며, 최대 70%까지 오픈소스로 구성됨
 - 특히, 인터넷&소프트웨어 인프라스트럭처(83%), 사물인터넷(82%), 교육기술(79%), 마케팅기술(78%), 사이버시큐리티(78%), 금융서비스&핀테크(75%), 에너지&청정기술(72%), 가상현실, 게이밍, 엔터테인먼트, 미디어(70%) 산업군의 코드베이스에 사용된 오픈소스량은 70% 이상으로 매우 높게 나타남
 - 한국리눅스재단에 따르면, IT 기술에 사용되는 시스템과 서비스 코드는 대부분 실사용 기업의 외부에서 개발되어 평균 30%에서 최대 80%까지 오픈소스로 개발된 코드를 사용하고, 20~70%의 코드를 자체 개발하여 기업의 가치 창출을 위해 차별화함¹⁴⁾

13) 시놉시스(2020), 2020 오픈 소스 시큐리티 & 리스트 분석 보고서

14) 이제웅 (2020), 「Cloud Native Computing Foundation」, 오픈 테크넷 서밋 2020 버추얼 컨퍼런스 발표 자료, (2020.09.16.)

II 오픈소스 진화단계 특성

1 분석 배경 및 개요

- 오늘날 오픈소스는 소프트웨어 산업을 넘어 ICT 산업에서 보편적 개발방식으로 자리매김하고 있을 뿐 아니라, 기술·산업 혁신의 촉진제로서 역할을 담당함
- 오픈소스가 어떠한 배경과 방식으로 전개됐으며, 그 영향은 무엇인가를 파악하는 것은 오픈소스의 발전 전망과 이해에 필수적임
- 따라서 오픈소스를 단순히 협력개발 기반 소프트웨어로 한정하는 것은 오픈소스가 지닌 ICT 생태계에서의 영향력을 충분히 이해하는데 부족하며, 변화된 오픈소스 생태계의 특성에 따라 인식과 대응 방향 또한 변화되어야 할 것으로 판단됨
- (방법 및 기준) 이를 위해, 본 분석에서는 오픈소스와 관련된 연도별 주요 이벤트를 분석하되, 생태계를 움직이는 주요항목(표 참조) 중심으로 태동기부터 현재까지 특징적 시기로 구분하고, 시기별 특징과 진화 양태를 분석하고자 함

표 2-1 구분 기준 및 분석내용

구분	분석 내용
주요특징, 전개과정	- 시기별 주요특징은 무엇이며, 전개과정은 어떠한가?
기술·시장환경 변화	- 시기별 기술·시장환경 변화는 무엇이고, 오픈소스는 어떠한 방식으로 적응해나갔는가?
대표제품 및 성능 수준	- 대표제품은 무엇이며, 이들의 성격과 수준은 어떠한가?
비즈니스 모델	- 시기별 비즈니스 모델의 변천은 어떠한가?
생태계 협력과 조성	- 생태계 내 주체 간 협력과 경쟁, 주도권은 어떻게 변천해왔는가?

※ 출처: 저자 작성



- (오픈소스 1.0-태동기) SW 사유화에 저항한 개발자의 자발적 참여로 시작함
 - 자유주의 소프트웨어 운동을 전개한 MIT 리처드 스톨만이 1983년에 시작한 GNU 프로젝트가 오픈소스SW¹⁵⁾의 기원
 - 자유 소프트웨어 재단(Free Software Foundation; FSF)이 시작되었고, 소프트웨어의 실행, 연구, 공유, 수정의 자유를 최종 사용자에게 보장하고자 한 GPL(General Public License)의 개념이 등장
- (오픈소스 2.0-확립기) 오픈소스 라이선스의 분화, 프로젝트 커뮤니티의 조직화, 수익화 모델 등 오픈소스의 기틀이 확립됨
 - 1998년 오픈소스 이니셔티브 재단의 창설과 오픈소스 개념이 등장
 - 다양한 라이선스 출현과 프로젝트 커뮤니티 조직화를 통한 재단의 설립이 빈번하게 이뤄진 시기이며, 오픈소스SW 지원(레드햇)과 오픈소스 라이선스 및 보안관리(블랙덕) 등의 체계적 수익화 모델이 등장한 시기
- (오픈소스 3.0-축진기) 오픈소스 생태계 자생력의 확보, 일부 기술·시장 우위를 증명함
 - 스마트폰, IT 서비스 등장으로 인한 소프트웨어 산업 패러다임 전환, 분산코드저장 알고리즘 깃(git)의 등장 등은 개발자를 오픈소스 진영으로 모으는 유인책이 되었고, 이는 오픈소스 생태계의 자생력 유지를 위한 ‘임계질량(critical mass)’을 확보하는 계기
 - 일부 프로젝트는 상용제품의 수준을 추월하거나 시장표준(De facto standard)이 되어 오픈소스SW의 기술·시장 우위를 증명한 시기
- (오픈소스 4.0-성장기) 인공지능, IoT, 자율차 등 기술발전이 빠르게 전개되는 산업 영역에서 기술·산업 혁신을 촉진하는 보편적 개발방식으로 자리매김함
 - 신규 사업 분야에서 빅테크 기업들은 시드(Seed) 기술을 공개하고 프로젝트 커뮤니티를 운영·지원 통해 개발을 촉진하며 우호적 생태계를 조성하는 전략적 활용증가

15) 당시에는 오픈소스 소프트웨어라는 용어라는 존재하지 않았고, 자유(Free) 소프트웨어라는 용어가 사용되었다. 엄밀히 보면 이들은 라이선스 측면에서 2차 저작물의 의무적 소스 공개 여부에서 차이가 존재한다. 그럼에도 불구하고, 본 연구에서 자유 소프트웨어를 오픈소스 소프트웨어에 포함한 이유는 자유 소프트웨어가 오픈소스 소프트웨어의 원형(元型)일 뿐 아니라, 현재는 사실상 이들 간 구분 없이 오픈소스 소프트웨어로 통칭하고 있기 때문이다. 참고로 자유 소프트웨어 재단은 현재도 존재하며, 이들은 자유 소프트웨어가 오픈소스 소프트웨어로 불리는 것을 거부하며, FLOSS(Free Libre Open Source Software)로 부르기를 권고하고 있다. (홈페이지: <https://www.gnu.org/philosophy/floss-and-foss.html>)

그림 2-1 오픈소스 시기 구분과 주요사건



※ 출처: 저자작성, 시기별 주요사건은 저자의 집단토의 결과로 도출한 오픈소스 주요사건 20개를 기초로 선정



2 오픈소스 1.0 : SW 사유화에 저항한 개발자의 자발적 참여

「자유 소프트웨어 운동 정신이 개발자들 사이에서 강하게 작동되었던 시기로, 능력 있는 개발자들이 금전과 상관없이 자발적으로 참여한 오픈소스 태동기」

(시기) 1983~1997년. GNU 프로젝트 시작 시점(1983년)에서 오픈소스 이니셔티브가 창설(1998년) 전까지의 기간

(배경) PC 대중화와 상용 SW산업 성장, SW 사유화에 반대하는 개발자들의 저항운동

(전개) 1991년 리눅스 커널이 GPL 라이선스로 공개되면서 자유 소프트웨어 운동의 급성장 계기 마련

□ 자유 소프트웨어 정신에 따른 개발자들의 자발적 참여¹⁶⁾

- 1983년 MIT 리차드 스톨만(Richard Stallman)이 개발자 자유를 저해하는 기업의 소프트웨어 사유화 움직임에 저항하는 활동으로서 상용 OS 유닉스에 대항하는 GNU(GNU's Not Unix)라는 공개프로젝트를 시작한 것을 오픈소스 기원으로 볼 수 있음
 - ※ 1980년대 상용 소프트웨어 산업이 성장하고, AT&T가 유닉스에 대한 지적 재산을 강조하면서, 그간 활발하게 유지되어 온 개발자 간 소프트웨어 소스 공유 문화가 약화되고, 커뮤니티 운영의 중단이 발생. 이에 대한 저항으로 자유 소프트웨어 운동이 시작됨
- 이어 스톨만은 1985년 개발자의 자유로운 소프트웨어 개발과 이용을 장려하기 위해 자유 소프트웨어 재단(FSF: Free Software Foundation)을 설립하고 활동을 시작함
 - 소프트웨어의 실행, 연구, 공유, 수정의 자유를 최종 사용자에게 보장하고자 한 GPL(General Public License) 개념이 정립
 - FSF의 주도 아래, 다수개발자의 자발적 참여로 유닉스계열 상용 소프트웨어에 상응하는 다수의 소프트웨어가 개발
- 능력 있는 다수개발자가 금전과 상관없이 자발적으로 GNU 프로젝트에 참여하였고, GNU 커뮤니티는 단기간 내 빠른 성장을 달성하였음
- 1991년, 리누스 토발스가 IBM-PC용 유닉스 호환 운영체제인 리눅스 커널을 개발하고 이듬해 GPL 라이선스 기반으로 공개하면서 자유 소프트웨어 운동은 더욱 활성화됨
 - ※ 1992년 1월, GPL 라이선스가 적용된 리눅스 0.12버전이 배포

16) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사'의 내용을 참고하여 작성함

〈자유 소프트웨어 운동의 배경¹⁷⁾〉

- 1960년대, 컴퓨터(메인프레임)의 도입 때부터 소프트웨어는 소스코드와 함께 무료 배포되었으며, 누구나 이를 사용하면서 수정할 수 있는 문화가 정착되었음
- 1970년대, AT&T 벨(Bell) 연구소가 다중 컴퓨터 환경에서 구동되는 운영체제인 유닉스(UNIX)를 소스코드와 함께 공개하였고, 이를 내외부의 개발자들이 협업하여 공동개발을 하면서 발전시켜감
- 하지만, 1980년대 초, AT&T가 지적 재산을 강조하며 유닉스를 상용화하기 시작하고, 소프트웨어를 소스코드 형태가 아닌 바이너리 포맷으로 배포하면서 자유로운 ‘해커 문화’에 위기가 찾아옴
 - 초기 유닉스는 카피당 \$99로 소스코드도 함께 사실상 무료 제공되었으나 1984년 AT&T가 여러 회사로 물적 분할되는 과정에서 개별 기업의 이윤추구(저작권활용)가 허용되면서, AT&T는 갑자기 유닉스 가격을 \$99에서 \$250,000로 대폭 인상
- 기업의 소프트웨어 사유화(소스코드 배포금지, 가격인상) 시도로, 축적되어 온 개발자 문화가 고사할 위기에 놓이자, 이에 대한 저항으로 ‘자유 소프트웨어 운동’이 일어남
- MIT 인공지능연구실의 리처드 스톨만이 이에 대표적 인물로, 1983년 스톨만은 상업화된 유닉스에 대응하기 위한 GNU 프로젝트를 시작하고 1985년 자유 소프트웨어 재단을 설립하고 자유 소프트웨어(Free software)¹⁸⁾ 운동을 전개해 나감

□ PC 대중화와 상용 패키지 소프트웨어 시장의 성장

- 개인용 컴퓨터(Personal Computer; PC)가 대중화되면서, 상용 운영체제 및 패키지 소프트웨어가 활발하게 개발된 시기로, 개발자에서 일반 사용자로 저변이 확대되었음
 - 1970년대 처음 도입된 개인용 컴퓨터는 1980년대 이후, IBM PC, 애플의 매킨토시 등이 출시되면서 본격적으로 대중화 시대가 전개
 - PC 산업의 성장은 상용 패키지 소프트웨어 시장을 견인하였는데, 이는 컴퓨터가 개발자만이 사용하던 개발 장비에서 일반인도 활용하는 도구로 변모하였기 때문에 소프트웨어 역시 상품성이 강조될 필요가 있음
- (대표제품) 이 시기의 대표적인 소프트웨어로는 GNU, 리눅스 등 들 수 있음
 - (GNU) 상용화된 유닉스에 대항하여 만들어진 OS 커널이자 소프트웨어의 모음집으로, 완전한 자유 소프트웨어이며, 현재는 리눅스와 결합¹⁹⁾되어 많이 사용

17) 크리스토퍼 토찌(2017), ‘프리-오픈소스 소프트웨어 혁명의 역사’, 김종배·김광용(2017), “오픈소스 소프트웨어 비즈니스 모델”을 참고하여 작성함

18) 자유소프트웨어는 소스코드의 배포와 수정이 자유로운 소프트웨어로 GPL(General Public Licence) 라이선스를 적용하여 소프트웨어의 사유화를 방지



- (리눅스) 1991년 리누스 토발스가 공개한 개인용 유닉스 호환 운영체제 커널로 고가의 유닉스 운영체제를 PC로 사용할 수 있게 한 대표적 오픈소스SW
 - ※ 그 외에도 R, 파이썬, MySQL, PHP, 아파치 웹서버 등 현재까지도 많이 사용되고 있는 오픈소스 프로젝트들이 시작된 시기
- (제품성격 및 수준) 이 시기의 오픈소스SW는 상용제품의 기능을 재현하는 것이 프로젝트의 주된 목적으로, 성능 측면에서는 상용대비 열위로 성능구현과 안정성 측면에서 상용 프로그램을 대체할 수는 없는 수준으로 평가할 수 있음

□ 이윤 추구가 목적은 아니었으나, 일부 수익 모델 시도

- 이 시기 대다수 참여자는 이윤 추구를 목적으로 하지 않았고 기업에 종속되거나 사유화되는 것에 반대하였기 때문에 수익화 모델은 크게 발달하지 못하였고, 프로젝트 유지를 위한 경비를 개인 후원으로 마련하는 정도의 활동이 진행됨
 - 당시 개발자들의 프로젝트 주된 참여 동기는 이윤 추구 활동이 아니라 소프트웨어의 상업화에 맞서는 자발적 참여운동으로 이해하는 것이 타당
 - (개인후원) 프로그램 소스코드는 인터넷을 통해 받을 수 있었지만, 개인들은 테이프 또는 디스크를 통해 GNU를 구매함으로써 GNU 프로젝트에 후원이 가능²⁰⁾
 - ※ GNU를 활용하는 기업들이 늘어나면서 휴렛 패커드, 리스프 머신 등의 일부 기업들이 GNU 프로젝트를 후원하는 기원후원 사례도 존재
- 그러나, 오픈소스SW의 기본적 비즈니스 모델들이 일부 시작된 시기기도 함
 - (배포 사업) 최초의 대중적 리눅스 패키지인 슬랙웨어(1993)²¹⁾, 최초의 커뮤니티인 데비안 프로젝트(1993)²²⁾가 조직되었고, 같은 시기 리눅스 패키징 서비스를 사업화한 최초의 기업 레드햇(1994)²³⁾이 설립
 - (기술지원) 시그너스 솔루션즈 등이 자유 소프트웨어를 위한 기술지원 사업을 시작²⁴⁾
 - (리눅스 엑스포) 1995년 리눅스 진영이 첫 번째 상업 행사인 '리눅스 엑스포'를 개최²⁵⁾
 - ※ 리눅스를 활용한 여러 제품과 사업모델이 등장하고 판매자와 구매자가 한데 모인 첫 상용 이벤트, 노스캐롤라이나 주립대에서 개최

19) GNU는 산업적으로 사용 가능한 운영체제 커널이 없어서 리눅스 커널과 함께 자주 사용됨

20) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.97~98

21) <https://zdnet.co.kr/view/?no=20200826135027>

22) <https://www.debian.org/doc/manuals/project-history/>

23) <https://www.redhat.com/en/events/레드햇에-대해-알아야-할-25가지-사실>

24) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.99

25) <https://www.linuxjournal.com/article/9065>

3 오픈소스 2.0 : 라이선스 정립, 커뮤니티 조직화 등 오픈소스 기틀 마련

「오픈소스SW의 기업 활용 확대, 프로젝트 커뮤니티의 지속 가능한 성장 및 체계적 지원이 가능해지면서 오픈소스의 기틀을 마련한 확립기」

(시기) 1998~2005년. 오픈소스 이니셔티브 창설(1998년)에서 분산코드저장 알고리즘 깃(git)의 개발(2006년) 전까지의 기간

(배경) 오픈소스 생태계 내 기업 참여 제고와 체계적인 커뮤니티의 지원 필요성 증대, 인터넷 산업 성장으로 서버용 OS 및 시스템 SW 수요 증가는 기회 요소

(전개) 완화된 라이선스의 허용은 오픈소스의 상업적 활용과 기업 참여를 견인하였고, 활성화된 프로젝트 커뮤니티의 재단설립이 이뤄지며, 오픈소스SW 기반 수익화 모델이 안착

□ 오픈소스 개념 등장과 오픈소스 라이선스 정립 및 분화

- GNU의 철학과 GPL 라이선스는 기업의 자유 소프트웨어 수용을 어렵게 하는 요인이 되었고, 이는 자유 소프트웨어 대중적 확산을 더디게 한 측면이 있음
 - 독점적 소프트웨어 방지를 위해 개발코드를 의무공개하는 것은 수익창출에 부정적 인식요소로 작용
- 이에 자유 소프트웨어의 라이선스보다 유연한 라이선스를 포함하는 ‘오픈소스’ 개념이 정의되고, 이에 부합하는 다양한 라이선스들이 등장하기 시작함
 - 1997년 브루스 페렌스와 에릭 레이몬드는 오픈소스 이니셔티브(Open Source Initiative)를 설립하고 자유 소프트웨어 대신 오픈소스SW라는 용어를 사용하기 시작²⁶⁾
 - 이 시기, OSI가 오픈소스 정의(Open Source Definition; OSD)를 개발하고 OSD에 부합하는 라이선스를 인증하면서 다양한 오픈소스 라이선스가 등장
 - ※ MIT 라이선스, 아파치 라이선스 등이 현재 주류 오픈소스 라이선스가 등장하였으며, 이들은 2차 저작물에 대한 재배포 의무가 없고 독점 소프트웨어와 함께 사용할 수 있는 허가적 라이선스(Permissive License)의 적용으로 기업이 2차 저작물의 소스코드는 공개하지 않고 사용료를 받을 수 있는 길을 열어 줌
- 오픈소스의 허가적 라이선스는 소프트웨어의 상업적 이용을 허용해줌으로써 기업의 오픈소스 참여 및 활용을 증대시키고, 오픈소스의 외연을 확대하는 데 이바지함
 - 기업은 다양한 라이선스 중에서 자사 전략에 부합하는 라이선스를 선택하여 활용 가능

26) 김종배·김광용(2017), “오픈소스 소프트웨어 비즈니스 모델”, p.75



□ 커뮤니티의 조직화 및 재단설립

- 오픈소스 라이선스가 분화되고, 프로젝트들이 빠르게 성장함에 따라, 이로 인해 발생할 수 있는 법률적 문제(저작권)와 체계적인 커뮤니티 운영지원의 필요성이 증대됨
- 이에, 개발자의 권익을 보호하고, 커뮤니티의 체계적인 관리 및 지원을 하기 위한 재단이 대형화된 주요 프로젝트별로 설립되기 시작함
 - 리눅스 재단, 아파치 재단, 이클립스 재단 등이 해당 프로젝트를 기반으로 설립되었고, 오픈소스 이니셔티브와 같이 생태계 자체를 지원하기 위해 재단도 창설
- 이들은 법률, 교육, 커뮤니티 관리, 이벤트 운영, 라이선스 관리 등 다양한 지원을 통해 오픈소스 생태계를 활성화하는 역할을 담당하며 개인 또는 기업의 후원을 받아 운영됨

표 2-2 주요 오픈소스 재단 개요

구분	개요
리눅스 재단	- 많은 오픈소스SW의 토대가 되는 리눅스 커널을 후원하는 재단으로 2000년에 설립27) ※ 현재 리눅스를 포함하여 클라우드, 인공지능, 네트워크 등의 다양한 분야에서 100개 이상의 오픈소스 프로젝트를 지원
아파치 재단	- 아파치 웹서버 및 여러 소프트웨어를 협업하여 개발하던 커뮤니티인 아파치 그룹이 급격히 성장하면서 1999년 아파치 재단을 설립28) ※ 세계적으로 가장 많이 사용되는 웹서버 중 하나인 아파치 웹서버, 빅데이터 분석 플랫폼 하둡(Hadoop) 등 다수 오픈소스 프로젝트를 운영하며, 법률 및 재무 등 지원조직을 운영29)
이클립스 재단	- IBM이 개발한 이클립스 프로젝트를 오픈소스 기술로 전환하면서 2004년에 설립된 재단30) ※ 소프트웨어 개발, 배포, 관리용 소프트웨어를 개발하는 오픈소스 커뮤니티를 지원

※ 출처: 저자 작성

□ 인터넷과 웹서버 시장 성장은 새로운 기회 요소

- (기술환경 변화) 2000년대에 들어서면서 초고속통신망과 이용하기 쉽고 멀티미디어에 최적화된 윈도 OS가 대중화되면서 웹(web) 기반 인터넷 산업이 폭발적으로 성장함
 - IT 산업에 막대한 투자금이 유입되면서 전자상거래, 인터넷 포털, 스트리밍, 소셜 네트워킹 등 다양한 신규 서비스가 시도·활성화

27) 리눅스 재단 홈페이지, 위키백과 '리눅스 재단' 참고하여 작성

28) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.230

29) <https://www.ciokorea.com/slideshow/26664>

30) <https://terms.naver.com/entry.nhn?docId=3580755&cid=59088&categoryId=59096>

- 이 시기의 라이선스 완화와 인터넷 산업 성장은 기업의 오픈소스 채택과 활용을 증가시키는 계기를 만들
 - 인터넷 산업의 활황은 웹 기반의 다양한 신규 개발 수요를 촉진하였고, 이는 라이선스 완화와 함께 오픈소스 진영의 기회 요소로 작용
- 특히, 리눅스는 서버용 운영체제로 빠르게 재편·활용되면서, 기타 오픈소스 시스템 소프트웨어(Apache, PHP), 데이터베이스 관리 시스템(MySQL) 확산을 견인함
 - 리눅스는 MS의 윈도우NT 뿐 아니라 당시 서버용 주류 OS인 솔라리스 등과도 경쟁할 수준으로 성장하였고, 오늘날 웹서버와 클라우드 OS 시장에서 선도적 지위를 지님
 - ※ 리눅스는 전 세계 상용 서버 운영체제 배포의 51% 이상 차지하고 있으며, 비상용 서버 시장까지 고려할 경우 70% 이상을 차지하고 있음³¹⁾ ('18년기준)
- **(대표제품)** 오늘날 대표적 오픈소스SW로 인정받고 있는 리눅스, 아파치 웹서버, MySQL, PHP (Linux, Apache, Mysql, PHP; LAMP)가 이 시기 웹 서버 수요의 확산과 함께 활성화된 대표적 프로젝트임
 - ※ 이들은 1.0시기에 개발이 시작됐으나, 2.0시기에 성장하여 대표적인 오픈소스SW로 자리매김

표 2-3 오픈소스 2.0 시기 대표제품 (LAMP)

구분	개요
리눅스 (Linux)	- IBM PC용 유닉스 호환 OS로 개발되었으나, 인터넷 산업 성장과 함께 서버/엔터프라이즈용 OS로 활용되면서 급성장하여, 오늘날까지 선도적 지위 확보 ※ IBM은 2000년 리눅스 지지를 선언하고, 약 10억 불을 투자계획을 발표. IBM의 리눅스 투자 선언으로 엔터프라이즈 시장에서 리눅스 사용 분위기가 조성 ³²⁾
아파치 웹서버 (Apache Web server)	- 아파치 재단에서 관리하는 HTTP 웹서버로 현재까지도 세계에서 가장 인기 있는 웹서버 프로그램 중 하나 ※ 1995년 2월 8명의 개발자들이 아파치 그룹을 조직하고 아파치 웹서버를 개발한 것이 시초로, 자체적인 아파치 라이선스를 도입하여 배포한 것이 특징 ³³⁾
MySQL	- 현재까지도 널리 쓰이는 관계형 데이터베이스 관리 시스템, 전체 소스코드를 포함한 리눅스용 버전이 1996년 11월에 공개됨 ※ 1994년 마이클 위데이우스와 데이빗 액스마크가 개발하였으며, 2008년 썬마이크로시스템즈에 인수되었다가 2010년 오라클에 인수됨 ³⁴⁾³⁵⁾
PHP	- 1995년에 공개된 동적 웹 프로그래밍 스크립트 언어로, 윈도를 지원하면서 MS의 ASP, 썬마이크로시스템즈의 JSP와 함께 활발하게 사용되었음 ※ 1995년 라스무스 러도프가 개발하여 공개한 언어로 ³⁶⁾ , 현재는 The PHP Group이라는 단체에서 개발 및 관리를 맡고 있음

※ 출처: 저자 작성

31) <https://www.redhat.com/ko/blog/red-hat-leading-enterprise-linux-server-market>



- (제품성격 및 수준) 이 시기의 오픈소스SW는 리눅스 등 일부 제품의 경우 웹서버 시장에서 대등한 성능과 가격 경쟁력으로 상용제품을 대체할 수 있는 수준에 근접했으나, 안정성과 유지보수 측면에서의 우려로 주류로 편입되지는 못한 것으로 평가됨

□ 다양한 오픈소스SW 비즈니스 모델 출현

- 기업의 오픈소스SW 활용이 늘어나면서, 본격적인 수익화 모델이 등장함
- (지원서비스) 컨설팅, 기술지원, 법률 지원, 교육, 오픈소스 개발용 호스팅 서비스 등 다양한 오픈소스 지원서비스를 통해 수익을 창출하는 대표적 수익 모델 중 하나
 - 소스코드가 공개되어 있어도 사용자가 이를 설치·활용하기가 어려우므로, 최신·최적의 오픈소스SW를 패키징하여 판매하거나, 관련 기술교육 및 훈련, 법률 지원, 컨설팅 등을 제공하는 서비스 모델
 - ※ (레드햇) 기업용 오픈소스의 사용이 본격화되면서 패키징된 리눅스 배포판을 판매하던 레드햇은 1999년 주식을 상장하였으며, 지원 서비스 기업인 시그너스 솔루션즈를 합병³⁷⁾
 - ※ (블랙덕) 2002년 설립된 블랙덕³⁸⁾은 오픈소스 기술 컨설팅, 보안 취약점 검사, 구성요소 분석 등을 수행하는 BlackDuck 소프트웨어를 판매
 - ※ (소스포지) 소스포지(SourceForge)는 오픈소스 소프트웨어 개발용 호스팅 서비스를 운영
 - 기업들이 쉽게 오픈소스를 활용할 수 있도록 하여 오픈소스에 대한 진입장벽을 낮춤
- (소프트웨어 판매) 듀얼 라이선스, 결합 판매, 허가적 라이선스 등 완화된 오픈소스 라이선스를 활용한 소프트웨어 판매 모델이 가능해짐
 - (듀얼 라이선스) 하나의 소프트웨어에 오픈소스SW 라이선스와 상용 라이선스를 동시에 부여하여, 사용자가 사용료를 낼지 2차 저작물의 소스코드를 공개할지를 선택할 수 있도록 함 (예: MySQL)
 - (결합 판매) 오픈소스로 개발한 기본 버전과 기본 버전에 결합할 수 있는 추가기능을 갖춘 상업적 버전을 개발하여 상업용 버전에 대한 판매수익을 창출 (예: 샌드메일)
 - (허가적 라이선스) 2차 저작물에 대한 코드 공개 조항이 없는 라이선스로, 원본의 소스코드는 공개하되 2차 저작물의 소스코드는 공개하지 않고 사용료를 받음

32) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.240~242

33) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.224~231

34) <https://www.w3resource.com/mysql/mysql-tutorials.php>

35) <http://www.bloter.net/archives/142487>

36) <https://www.php.net/manual/en/history.php.php>

37) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사', p.237

38) http://www.dt.co.kr/contents.html?article_no=2013052402019960718003

- (하드웨어 보조모델) VA리눅스 등의 기업은 리눅스를 설치한 컴퓨터를 판매하는 하드웨어 보조모델(위젯 프로스팅)을 통해 사업을 진행하기도 함
 - ※ 위젯 프로스팅: 자사의 하드웨어에 맞는 디바이스 드라이버 등의 소프트웨어 공개를 통해 개발자나 사용자의 참여를 유도하는 방식

4 오픈소스 3.0 : 생태계 자생력 확보, 일부 기술·시장 우위 증명

『오픈소스 개발자 생태계 자생력을 확보하고, 빅테크기업 의 참여와 일부 기술·시장에서 상용대비 우위를 증명한 오픈소스 촉진기』

- (시기) 2006~2014년, 분산버전 관리 알고리즘 깃(git)의 개발(2006년)에서 구글의 텐서플로우 공개(2015년) 전까지의 기간
- (배경) 모바일 시대의 도래와 IT서비스(SaaS) 확산, 분산버전관리 알고리즘 깃(git) 등장, 빅테크 기업의 오픈소스 진영 참여 증가
- (전개) 소프트웨어 시장에서의 신규 개발 수요 폭증과 새로운 원격 협업개발 기술의 보급으로 오픈소스 개발자 생태계는 자생력을 위한 ‘임계질량’을 확보하였고, 일부 제품은 상용성능을 추월하고 시장표준이 되는 등 오픈소스SW의 기술 우위성을 증명

□ 소프트웨어 산업 패러다임 전환과 신규 개발 수요의 폭증

- (기술환경 변화) 이 시기는 ICT 산업의 중대한 패러다임 변화가 있었던 시기로, 소프트웨어 산업은 패키지 소프트웨어에서 서비스 소프트웨어로, PC에서 스마트폰으로, 웹에서 앱으로의 대전환이 진행됨
 - 전 세계 기업의 IT시스템이 자체 데이터센터와 서버에서 클라우드 컴퓨팅으로 본격적으로 옮겨가기 시작한 시기로, 아마존의 AWS(Amazon Web Service), 마이크로소프트 애저(Azure) 등 클라우드 서비스가 본격화
 - 2007년 애플 아이폰 출시로 스마트폰, 앱 중심으로 ICT 생태계가 재편
- 이러한 산업 패러다임의 변화로 인해 스마트폰, 태블릿 PC 등 모바일 기기들과 클라우드 등 IT 솔루션 서비스(SaaS)가 다양하게 출시되었으며 이는 신규 개발 수요를 창출하였음
 - 오픈소스 진영은 이러한 기술수요에 민첩하게 반응하며 새로운 기회를 획득함



- ※ IT 서비스 SW는 패키지 SW와 달리, 지속적이고 신속한 개발-수정-운영의 사이클을 가지고 있으므로, 커뮤니티 기반의 유연성과 다양성을 지는 오픈소스 방식이 유리
- ※ 개인 단말인 스마트폰과 앱의 등장은 개인 개발자가 오픈소스SW로 공개된 개발도구와 앱 스토어를 통해서 비교적 간단하게 아이디어 기반의 앱개발로 수익을 창출하는 새로운 경로를 형성

□ 분산협업 활성화와 개발자 생태계 자생력 확보

- 2006년 리누스 토발즈의 분산 버전관리 시스템인 깃(Git) 개발³⁹⁾과 2008년 깃을 활용한 오픈소스 저장소인 깃허브(GitHub)⁴⁰⁾가 등장하면서 오픈소스 커뮤니티는 급격히 성장함
 - ※ SW 버전관리 시스템: 기능은 파일의 변경 내역을 기록하고, 추적할 수 있게 하며, 분기하거나 병합이 가능하게 하며 동시 작업을 지원
 - 깃(Git)이 등장하면서 중앙관리형 대비 위험분산 및 로컬 개발의 효율성이 제고되어 원격협업과 개발자 유치가 용이
- (깃허브) 2008년에 설립된 세계 최대 오픈소스 저장소로, 개발자들이 깃(Git)을 통해 오픈소스 버전을 관리할 수 있도록 저장공간 및 관리 기능들을 제공하며, 공급자(개발자)와 수요자(기업)이 만나는 장(場)으로서의 역할을 담당함
 - 개발자 코드 기여분에 대해 철저한 버전 관리는 커뮤니티 내 투명성과 공정성을 확보하고 프로젝트 관리 용이성을 제고 하여 대규모 프로젝트의 분산 협업개발이 가능케 함
 - 개발자들은 깃허브에 공개된 기업 프로젝트 참여를 통해 자신의 개발 능력을 홍보할 수 있고, 기업들은 유능한 개발자를 발굴하고 채용하는 루트로 활용
- (생태계 자생력 확보) 분산협업 시스템 발달과 앱(app) 플랫폼의 활성화로 개인 개발자들이 대거 오픈소스 개발진영으로 유입되었고, 빅데이터, 클라우드 등 IT서비스로의 소프트웨어 산업 전환에 따른 신규 개발 수요를 수용함으로써, 오픈소스 진영은 최대 SW 개발자 생태계로서 지속 가능한 ‘임계질량(critical mass)’을 확보함
 - 오픈소스 커뮤니티 내 개발자들을 활용하려는 기업들이 더 많은 프로젝트를 공개하는 등 네트워크 외부성이 생기면서 더 많은 신규 개발자의 유입을 견인
 - ※ 깃허브 내 저장소(Repository)는 2013년에 약 1,000만 개였으나, 2018년에는 1억 개를 돌파하여 2020년 11월 기준 약 1.9억 개의 저장소가 있는 등 최근 급격히 증가⁴¹⁾
 - ※ 깃허브 내 개발자 수도 지속해서 증가하여 2017년 2,400만 명에서 2019년 4,000만 명으로 증가⁴²⁾

39) http://www.tta.or.kr/data/weeklyNoticeView.jsp?pk_num=5775

40) <https://www.samsungsds.com/kr/insights/github.html>

41) GitHub 블로그, GitHub 사이트를 참고하여 작성

42) GitHub Octoverse 2017년, 2019년을 참고하여 작성

□ 빅테크 기업의 오픈소스 참여와 지원 본격화

- 이전 시기까지 개발자 커뮤니티, 재단이 오픈소스 진영의 주도적인 세력이었다면, 이 시기부터는 기업의 역할이 주요하게 부각된 시기임
 - 오픈소스SW의 가능성이 확인되자, 기업은 오픈소스를 채택하여 활용하고, 관련 생태계를 후원하게 되면서 영향력을 확보
 - ※ 기업들의 후원을 받아 재단도 지속적으로 성장하였으며, 재단들은 오픈소스 프로젝트 및 개발자를 지원하고 기업, 오픈소스 커뮤니티, 그리고 개발자들 사이의 연결고리 역할을 수행
- 특히, 빅테크 기업들은 유망한 오픈소스 프로젝트를 발굴·채택하여 이를 인수하거나 관련 재단 및 커뮤니티를 적극적으로 후원하면서 오픈소스 생태계에 참여함
 - ※ 구글은 2005년 안드로이드社를 인수한 후, 2008년 안드로이드의 기본 소스코드를 ‘안드로이드 오픈소스 프로젝트’로 공개하고 여러 기업들과의 협력 관계 구축을 통해 모바일 생태계를 장악
 - ※ 리눅스 재단의 경우, 2008년 기준 약 674만 달러였던 수익이 2014년 기준 약 3,145만 달러에 달하는 등 매우 빠른 성장세를 보이는데 이들의 수익의 상당 부분이 기업 연회비와 기부금임⁴³⁾
- (대표제품 및 수준) 모바일OS인 안드로이드, 클라우드 구축을 위한 오픈스택과 도커, 빅데이터 처리를 위한 하둡 등이 대표적인 프로젝트로 이들은 기술적 성능뿐 아니라, 시장 선호에서도 상용제품 대비 우위를 점하며 시장 표준으로 자리매김함
 - 안드로이드는 리눅스를 기반으로 구축되어 모바일OS의 시장표준으로 자리 잡음
 - 특히 클라우드 시장은 오픈스택, 클라우드 스택, 도커, 클라우드 파운드리 등 오픈소스 기반 기술들이 시장을 선도

표 2-4 오픈소스 3.0 시기 대표제품

구분	개요
안드로이드 (Android)	- 2008년 오픈소스로 공개된 모바일 OS로, 2020년 6월 기준 스마트폰 OS 점유율 74.2%를 차지 ⁴⁴⁾ , 스마트폰뿐만 아니라 다양한 모바일 기기에 사용
오픈스택 (OpenStack)	- 데이터센터의 컴퓨팅, 스토리지, 네트워킹 자원들을 제어하는 클라우드 운영 체제로, 클라우드 오픈소스 프로젝트 중 가장 인기 있는 프로젝트 중 하나 ⁴⁵⁾
도커 (Docker)	- 닷클라우드社(現 도커社)에서 2013년 발표한 오픈소스 프로젝트로, 리눅스 컨테이너 ⁴⁶⁾ 자동화 기술
하둡 (Hadoop)	- 대용량 데이터 분석 처리를 위한 분산 컴퓨팅 프레임워크로 2011년에 발표 ⁴⁷⁾

※ 출처: 저자 작성

43) <https://projects.propublica.org/nonprofits/organizations/460503801>

44) <https://moneys.mt.co.kr/news/mwView.php?no=2020070111268036630>



□ 플랫폼 기반의 수익화 전략 등장과 기존 수익모델의 대형화

- 빅테크 기업들은 오픈소스 기술을 채택하여 모바일 앱, 클라우드 플랫폼을 구축하고 플랫폼 서비스를 통해 수익을 창출하기 시작함
- (모바일 플랫폼) 구글은 2008년 공개한 오픈소스 모바일OS인 안드로이드를 기반으로 구글 플레이스토어 등의 플랫폼 비즈니스를 통해 수익을 창출하고 있음
 - 구글은 안드로이드 기술을 인수한 후, 안드로이드의 기본 소스코드를 ‘안드로이드 오픈소스 프로젝트’로 공개하고 여러 기업들과의 공동개발 및 협력 관계 구축을 통해 모바일 생태계를 장악
 - ※ 오픈소스로의 공개 결정은 애플과의 경쟁에서 뒤쳐져 당시 개발자 대부분이 아이폰 진영으로 쏠리고 있던 것을 만회하기 위한 전략이었다는 것이 중론
 - 안드로이드 OS 위에 앱 마켓인 구글 플레이스토어를 탑재하여 앱 거래 수수료와 스마트폰 광고를 통해 수익 창출
 - 안드로이드 오픈소스 프로젝트에 구글 인증 및 플레이스토어를 비롯한 독점기능 등을 포함한 구글 모바일 서비스(Google Mobile Service)를 탑재하여 ‘안드로이드’⁴⁸⁾를 출시
 - ※ 즉, 안드로이드 오픈 소스 프로젝트(Android Open Source Project)는 오픈소스SW이지만, 안드로이드는 구글의 제품임
- (클라우드 플랫폼) 클라우드 기업들은 오픈소스SW를 일부 활용하여 클라우드 플랫폼을 구축하고 서비스를 제공하는 수익모델을 채택하기 시작함
 - ※ 2006년 아마존 웹 서비스(Amazon Web Service: AWS)⁴⁹⁾, 2010년 마이크로소프트 애저(Azure)⁵⁰⁾, 2011년 구글 클라우드 플랫폼 발표⁵¹⁾, 2014년 IBM 블루믹스(Bluemix)⁵²⁾ 등 주요 빅테크 기업들이 자사의 클라우드 플랫폼을 출시
- (기존 수익모델) 오픈소스 생태계가 더욱 활성화되면서 기업용 오픈소스SW 판매와 지원·교육 서비스 시장이 대형화됨
 - ※ (레드햇) 기업용 오픈소스 판매와 지원·교육 서비스 등을 통해 수익을 지속적으로 창출하여 2012년 매출 1조 원을 달성⁵³⁾

45) <https://www.zdnet.com/article/most-popular-open-source-cloud-projects-of-2014/>

46) 전체 런타임 환경에서 애플리케이션을 패키지와 분리하는 기술로 이를 통해 전체 기능을 유지하면서 컨테이너화된 애플리케이션을 환경(개발, 테스트, 생산 등) 간에 쉽게 이동할 수 있음

47) <https://intellipaat.com/blog/hadoop-creator-goes-to-cloudera/>

48) 구글이 안드로이드 저작권 및 상표권을 보유하고 있으므로 안드로이드 오픈소스 프로젝트를 활용하더라도 구글인증(CTS)을 받고 구글 모바일서비스를 탑재하지 않으면 ‘안드로이드’란 이름을 사용 못함

49) <https://aws.amazon.com/ko/about-aws/>

50) <https://blogs.microsoft.com/blog/2010/02/01/windows-azure-general-availability/>

51) <https://www.infoworld.com/article/3254749/google-cloud-platform-services-guide-the-right-tools-for-the-job.html>

52) <https://www-03.ibm.com/press/us/en/pressrelease/43257.wss>

53) http://ddaily.co.kr/m/m_article/?no=89646

5 오픈소스 4.0 : 빅테크 기업 주도, 오픈소스의 혁신도구화

『빅테크 기업의 오픈소스 생태계를 주도하고, 기술·산업 혁신을 선도하는
보편화된 개발방식으로 정착하는 오픈소스의 성장기』

(시기) 2015~현재, 구글의 텐서플로우 공개(2015년)에서 현재까지의 기간

(배경) 오픈소스의 ICT 산업 내 영향력 증대, 빅테크 기업의 전략적 참여 및 활용 증가

(전개) 빅테크 기업들은 시드(Seed) 기술을 공개하고 프로젝트 커뮤니티를 운영·지원 통해 기술개발을 촉진하며 우호적 생태계를 조성하는 전략을 구사하는 한편, 오픈소스는 소프트웨어 산업을 넘어 인공지능, IoT, 자율주행차 등 기술발전이 빠른 산업영역에서 기술·서비스 혁신을 촉진하는 보편적 개발방식으로 자리매김

□ 빅테크 기업의 전략에 의한 오픈소스 주도

- 오픈소스가 기술 및 서비스 혁신의 주도적 개발방식으로 부상함에 따라, 구글, 아마존, 마이크로소프트, 삼성 등의 빅테크 기업들은 오픈소스를 적극적으로 수용하며, 자사 주도의 혁신 생태계 구축을 시도하고 있음

※ 깃허브에 따르면, 글로벌 Fortune 50 기업 중 70%인 35개 기업이 2019년에 오픈소스에 기여⁵⁴⁾

- (전략적 활용) 이들은 단순히 오픈소스 활용에만 그치는 것이 아니라, 시드기술 제공이나 커뮤니티 및 재단 지원 등 자사 우호적 생태계 조성을 위한 전략을 구사하기 시작함

- 시드기술 제공: 빅테크 기업들은 핵심(Seed) 기술을 오픈소스로 공개하여, 관련 개발자 생태계의 주도권을 확보하고, 외부 개발자들을 통한 기술개발을 촉진
- 재단 후원: 빅테크 기업들은 주요 오픈소스 재단들을 후원하고 자사 개발자의 주요 오픈소스 프로젝트 참여를 통해 재단 및 커뮤니티에 대한 영향력을 증대
- 일레로, 빅테크 기업들은 아파치 재단, 리눅스 재단 등 빅데이터, 인공지능 분야의 핵심 기술을 다루는 프로젝트들을 운영하는 재단과 파트너십을 맺고 운영자금을 지원

※ 대표적으로 리눅스 재단은 현재 구글, 페이스북, 인텔, 퀄컴, 삼성 등 900개 이상의 기업 회원이 후원⁵⁵⁾하고 있으며, 2018년 기준 리눅스 재단의 수익은 약 9,658만 달러에 달함⁵⁶⁾

54) <http://octoverse.github.com/2019>

55) <https://lf-landscape.netlify.app/category=lf-members&format=members&grouping=category>



표 2-5 오픈소스 4.0 시대, 빅테크 기업의 오픈소스 전략적 활용

구분	개요
구글	<ul style="list-style-type: none"> - (오픈소스 참여) 정규직원 중 약 9%가 오픈소스 프로젝트에 기여하고 있으며, 약 2,000개의 오픈소스 프로젝트를 운영 중 ('19년 기준) - 구글 직원은 사내 운영하는 프로젝트뿐만 아니라 외부 프로젝트에도 적극적으로 기여 <ul style="list-style-type: none"> ※ '19년 구글 직원은 깃허브 내 70,000개 이상의 프로젝트에서 활동하면서 40,000개 이상의 프로젝트에 참여(commit, Pull Request)하였고, 이들 참여의 75%는 구글 외부 프로젝트였음 - (재단 후원) 아파치 재단, 클라우드 파운드리 재단, 리눅스 재단, 이클립스 재단 등 31개 오픈소스 관련 재단 및 단체들을 후원 중 - (대표 프로젝트) 딥러닝 프레임워크 Tensorflow, 컨테이너 애플리케이션 배포·관리 시스템 Kubernetes, 모바일 플랫폼 Android, 웹브라우저 프로젝트 Chromium 등 <ul style="list-style-type: none"> ※ 텐서플로우(Tensorflow)와 쿠버네티스(Kubernetes)는 '19년 기준 깃허브 기여자 수 Top 10 안에 드는 프로젝트로, 각각 5위, 7위를 차지⁵⁷⁾
마이크로소프트	<ul style="list-style-type: none"> - (오픈소스 참여) 과거, MS는 오픈소스에 적대적이었으나, '14년 사티아 나델라 CEO가 취임하면서 오픈소스 활용정책으로 전환 - '17년 기준 약 1,300명의 직원들이 깃허브 탭 저장소 825개에서 활동⁵⁸⁾하고 있으며, 현재 깃허브에 약 3,700개의 저장소를 운영 중 - 오픈소스에 기여 중인 직원들이 선정한 외부 프로젝트에 대해 총 10,000달러를 후원하는 FOSS 펀드를 운영 중 <ul style="list-style-type: none"> ※ MS에서 쓰이지만 MS 직원이 소유하지 않은 오픈소스 프로젝트를 대상으로 하며, 현재 3개 프로젝트를 후원하였고, 네 번째 대상의 선정절차가 진행 중 - (재단 후원) 리눅스 재단, 이클립스 재단을 비롯한 22개 재단 또는 이니셔티브를 후원 - '18년 깃허브를 75억 달러(8조 원)에 인수하기도 하였음 - (대표 프로젝트) 2019년 기준, 기여자 수 1위인 vscode와 2위인 azure-docs를 운영 중이며, DotNET, Powershell Core, CNTK 등도 대표적 프로젝트
아마존	<ul style="list-style-type: none"> - (오픈소스 참여) 2017년 기준 약 134명의 직원들이 158개의 깃허브 탭 프로젝트에서 활동⁵⁹⁾하고 있으며, 깃허브에 약 2,500개의 프로젝트를 운영 - (재단 후원) 아파치 재단, 리눅스 재단, 오픈소스 이니셔티브 등 24개 오픈소스 관련 재단 및 단체를 후원하며, Pycon, OSCON 등의 컨퍼런스도 후원 - (대표 프로젝트) 클라우드 가상화 도구 AWS, 음성인식 서비스를 위한 개발자 도구 Alexa, 딥러닝 프레임워크 MXnet 등이 대표적

※ 출처: 각 사의 오픈소스 및 블로그 사이트 내용을 참고하여 저자 작성

56) <https://projects.propublica.org/nonprofits/organizations/460503801>

57) <http://octoverse.github.com/2019>

58) Hoffa, F. (2017), "Who contributed the most to open source in 2017 and 2018? Let's analyze GitHub's data and find out.", <https://www.freecodecamp.org/news/the-top-contributors-to-github-2017-be98ab854e87/>

- 빅테크 기업의 오픈소스 참여와 관심은 오픈소스 생태계 유지와 발전에 도움이 되나, 무임승차, 독점, 다양성 저하 등에 대한 우려가 상존하는 것도 사실임
 - ※ 빅테크 기업들이 오픈소스SW를 활용한 플랫폼 사업을 시작하면서 일부 오픈소스 프로젝트 개발자들은 이들이 무임승차하고 있다고 비판하며, 자사의 프로젝트를 보다 제한적인 라이선스로 변경⁶⁰⁾

□ 오픈소스의 혁신도구화

- 오늘날, 제4차 산업혁명의 핵심기술 중 많은 부분이 오픈소스로 개발되고 있는 등 오픈소스는 기술 혁신을 이끄는 보편적 개발방식으로 활용되고 있음
 - 오늘날 소프트웨어 기술수명 짧아지고, 개발 규모와 과정이 거대화, 복잡화되면서, 오픈소스 개발방식이 갖는 다양성, 유연성, 협업 용이성 등이 장점으로 작용
 - 참여하는 기업으로서도 자사 기술 중심의 개발자 및 파트너 생태계를 구축할 수 있기 때문에 적극적
- 대부분의 인공지능 기술이 오픈소스로 공개되고 있으며, 인공지능 관련 오픈소스 프로젝트는 2017년(14,000개)을 기점으로 급격히 증가함⁶¹⁾
 - ※ 텐서플로우(Tensorflow), 파이토치(PyTorch), 케라스(Keras), 카페(Caffe), 테아노(Theano), CNTK, MXNet, Deeplearning4j 등 대표적인 머신러닝 프레임워크들이 오픈소스로 개발
 - ※ 대다수의 이미지 처리, 음성 처리, 자연어 처리, 감성처리 등 인공지능 주요 분야별 최신 알고리즘(State Of The Art; SOTA)들이 오픈소스로 공개됨⁶²⁾
- 빅데이터 및 클라우드 분야는 오픈소스 프로젝트들이 높은 시장점유율을 보이며 사실상 시장표준으로 활용되고 있음
 - ※ (빅데이터) 아파치 하둡(Hadoop), 스파크(Spark), 카산드라(Cassandra), MongoDB 등 빅데이터 관리 및 처리를 위한 오픈소스가 시장에서 높은 점유율을 보임
 - ※ (클라우드) 컨테이너 오케스트레이션 소프트웨어인 쿠버네티스(Kubernetes), 컨테이너 자동화 기술 도커(Docker) 등 클라우드 핵심기술들이 오픈소스를 기반으로 함
- 뿐만 아니라, 오픈소스는 블록체인, IoT, 빅데이터, 클라우드, 5G, 엣지 컴퓨팅 등 새롭게 부상하는 IT 기술 분야에서도 활발히 활용되고 있음
 - ※ (블록체인) 비트코인(Bitcoin), 이더리움(Ethereum), 하이퍼레저(Hyperledger), 오픈체인(OpenChain) 등 다양한 블록체인 플랫폼이 오픈소스로 개발 중

59) 58)과 출처 같음

60) <http://www.bloter.net/archives/365067>

61) 이진휘(2019), “오픈소스 중요성과 시사점”, NIPA

62) <https://www.paperswithcode.com/>



- ※ (IoT) 엣지X 파운드리(EdgeX Foundry), 이클립스 쿠라(Kura), IoTivity 등 IoT 관련 오픈소스 프로젝트들이 진행 중
- ※ (5G & 엣지컴퓨팅) 5G의 핵심기능인 네트워크가상화와 MEC(Multi-access Edge Computing) 구현을 위한 LF edge 산하 AKRAINO 프로젝트, OSF 산하 Airship, StarlingX 프로젝트 등에 통신업체와 클라우드 업체가 참여 중
- 오픈소스는 자율주행차&로봇 같은 버티컬 산업 분야에서의 기술 혁신, 생태계 조성을 위한 협력 방식으로 활발히 활용되고 있음
 - (자율주행차) 자율주행 분야 오픈소스 프로젝트는 기업 위주로 진행이 되고 있는데, 이는 신생분야에서 자사 기술 중심의 생태계 조성을 위한 전략으로 풀이
 - ※ 바이두의 아폴로(Apollo), Apex.AI의 '아펙스.OS', GM 자회사 크루즈의 '월드뷰' 등 다양한 업체가 오픈소스 프로젝트를 공개
- (대표제품) 딥러닝 프레임워크인 텐서플로우, 파이토치, 클라우드기술의 쿠버네티스는 시장과 기술을 선도하는 대표적 프로젝트이며, 자율주행 등 신산업 분야에서도 민간중심의 오픈 소스 프로젝트들이 활발히 활동 중임

표 2-6 오픈소스 4.0 시기 대표제품

구분	개요
텐서플로우	- 가장 널리 사용되고 있는 오픈소스 머신러닝 프레임워크로 구글의 머신인텔리전스 연구 조직인 구글 브레인팀이 머신러닝과 심층신경망 연구를 위해 처음으로 개발하였으며, 2015년 11월에 배포됨 ⁶³⁾⁶⁴⁾ ※ 2011년 개발한 딥러닝 인프라스트럭처 디스트빌리프(DistBelief)를 개선한 2세대 머신러닝 프레임워크임
파이토치	- Torch를 기반으로 한 오픈소스 머신러닝 프레임워크로 텐서플로우에 비해 직관적이고 쉬워 많이 사용되고 있음 ※ 페이스북 AI 리서치 랩에서 개발하여 2016년에 최초로 배포되었으며, 2018년에 머신러닝 프레임워크 Caffe2가 pytorch로 합병 ⁶⁵⁾
쿠버네티스	- 구글에서 2014년 발표한 ⁶⁶⁾ 리눅스 컨테이너 오케스트레이션 소프트웨어로, 컨테이너화된 애플리케이션을 여러 대의 호스트에 배치 및 관리하는 작업을 자동화하는 기술
아폴로	- 2017년 바이두가 오픈소스로 공개한 자율주행차 기술 플랫폼으로 100개의 글로벌 기업이 참여 중이며, 자율주행 개발에 관련한 클라우드 데이터 서비스, SW, HW, 테스트 플랫폼을 포함한 풀 스택(full stack)을 제공 ⁶⁷⁾

※ 출처: 저자 작성

- (제품성격 및 수준) 오늘날 오픈소스는 소프트웨어 산업을 넘어 ICT 산업에서 보편적 개발방식의 하나로 자리매김하고 있으며, 혁신을 선도하는 주류 역할을 담당함
 - 근래 대다수 소프트웨어 알고리즘과 프레임워크는 오픈소스로 공개되며, 기술적 성능과 시장 선호 역시 상용제품 대비 우위를 점하며 혁신을 선도

□ 빅테크 기업의 플랫폼기반 수익모델 가속화

- 아마존, 구글, 마이크로소프트 등 빅테크 기업들은 컴퓨팅 인프라 및 프레임워크, 응용서비스를 구축하고, 고객들이 해당 자원을 활용하여 데이터를 가공하고 AI 모델을 구축할 수 있도록 하는 AI 플랫폼 사업을 적극적으로 전개 중임⁶⁸⁾
 - 음성인식, 번역, 이미지 분석 서비스 등 데이터만 적용하면 결과를 도출할 수 있도록 하는 API 형태의 AI 서비스를 제공하기 때문에 AI 개발 역량이 낮은 고객들도 쉽게 이용
 - ※ 비용은 대부분 각 서비스별 이용량 당 지불하며, 측정단위는 각 서비스에 따라 이용시간·문자길이·음성시간 등 차이가 존재
 - 구글 AI Platform, 마이크로소프트 Azure Machine Learning, 아마존 AWS Sage Maker 등이 대표적 서비스
- 이들은 오픈소스 방식을 적극적으로 수용하여 빠르게 자사의 서비스 플랫폼을 구축하는 한편, 플랫폼 구축 및 운영에 필요한 기술을 오픈소스로 공개하여 공동개발함으로써 플랫폼 비즈니스를 확장해나가고 있음
 - 특정 핵심모듈은 공개하여 다수의 개발자가 사용하도록 유도하여 시장을 확대하되, 머신러닝 모델 등 중요한 기술은 비공개로 하고 AI 플랫폼에서 유료로 제공
 - ※ AI 프레임워크: 구글의 텐서플로우, 아마존의 MXNet, 마이크로소프트의 CNTK 등 주요 AI 프레임워크는 오픈소스로 개발
 - ※ 클라우드: 구글은 클라우드 기술인 쿠버네티스를 오픈소스로 공개하여 개발하고 있으며, 마이크로소프트와 아마존도 각각 Azure 프로젝트와 AWS 프로젝트 운영을 통해 클라우드 운영에 필요한 기술을 오픈소스로 개발

63) 금융보안원(2017), “머신러닝 활용을 위한 오픈플랫폼 현황”

64) 남충현(2016), “오픈소스 AI : 인공지능 생태계와 오픈 이노베이션”, KISDI

65) <https://medium.com/@Synced/caffe2-merges-with-pytorch-a89c70ad9eb7>

66) 쿠버네티스가 클라우드 네이티브 컴퓨팅 재단을 졸업해 본격적인 오픈소스 프로젝트가 되다, 클라우드 이슈리포트 (2018.04.19.)

67) <https://apollo.auto/>

68) 아마존, 구글, 마이크로소프트의 클라우드 및 AI 플랫폼 사이트를 참고하여 작성



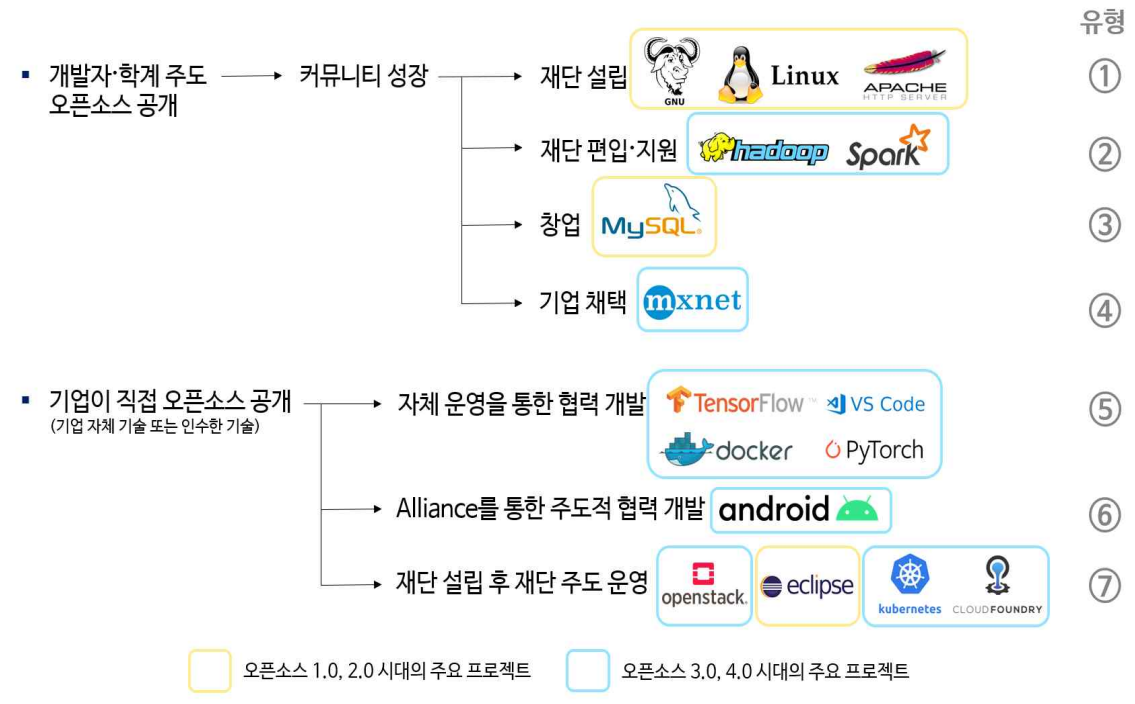
III 오픈소스, 협력과 경쟁의 진화

1 오픈소스 프로젝트 성장경로 유형

□ 분석의 개요

- 여기에서는 오픈소스 프로젝트들이 성장하는 경로를 대표적인 오픈소스 프로젝트 사례들을 통해 분석함
 - 약 15개의 주요한 오픈소스 프로젝트 사례를 분석하였으며, 사례별 오픈소스 공개 주체가 누구인지와 어떠한 활동을 통해 지속적으로 성장하였는지에 따라 유형을 분류
- 오픈소스 프로젝트는 주로 기업 주도 또는 개발자 및 학계 주도로 공개되며, 운영 및 성장 방식에 따라 그림 3-1과 같이 7가지 유형으로 나뉨

그림 3-1 오픈소스 프로젝트 성장 경로



※ 출처: 저자 작성

□ (사례 분석) 개발자·학계 주도 오픈소스 공개와 커뮤니티 성장

- 개인 개발자, 개발자 그룹, 또는 학계 등이 초기 기술을 개발하여 이를 오픈소스로 공개하면 커뮤니티를 중심으로 개발자들의 협력이 확대되면서 성장함
 - 오픈소스 1.0, 2.0 시대의 주요 프로젝트들 중에는 개발자 또는 학계가 주도하여 오픈소스로 공개한 기술이 많았으며, 오픈소스 3.0, 4.0 시대의 주요 프로젝트들 중 일부도 이러한 방식으로 공개됨
- 커뮤니티가 일정 규모로 성장하고 난 후, (유형 1) 직접 재단을 설립하거나, (유형 2) 재단에 편입되어 지원을 받거나, (유형 3) 창업을 하거나, (유형 4) 기업에 채택되는 등의 방식을 통해 지속적으로 성장하게 됨
- (유형 1: 재단 설립) 오픈소스 프로젝트가 일정 규모 이상 성장하고 나면, 이를 효율적으로 관리하기 위해 재단을 설립하여 프로젝트를 운영하며, 대표적인 예로 GNU, 리눅스, 아파치 웹서버 등이 있음
 - 오픈소스 2.0 시대 초반까지는 오픈소스 프로젝트를 지원할 단체나 기업이 거의 없었기 때문에 자체적으로 단체를 형성하여 개발을 지원하고 후원을 받아 커뮤니티를 성장시키기 위해 이러한 방식을 택하는 경우들이 나타났음

표 3-1 주요 오픈소스 프로젝트 예시(유형 1)

프로젝트	개 요
GNU	- 리처드 스톨만이 시작한 자유소프트웨어 프로젝트 (1983년) - GNU 개발을 지원하고 자유소프트웨어를 지지하기 위해 자유소프트웨어 재단을 설립하여 프로젝트 운영(1985년)
리눅스	- 리눅스 토발즈가 개발한 운영체제 커널 (1991년) - 커뮤니티가 크게 성장하면서 리눅스 재단이 설립되었으며(2000년), 현재까지도 리눅스 개발을 지원
아파치 웹서버	- 8인의 개발자들이 아파치 그룹을 형성하여 개발 시작(1995년) - 웹서버 프로젝트와 아파치 그룹이 성장하면서 이를 법적·재정적으로 지원하기 위해 아파치 소프트웨어 재단이 설립

※ 출처: 저자 작성

- (유형 2: 재단 편입 및 지원) 오픈소스 재단에 기술을 제공하여 재단에서 유지 보수 및 운영을 담당하기도 하며, 대표적인 예로 하둡, 스파크 등이 있음



- 개발자 또는 학계에서 비영리적인 목적으로 개발한 프로젝트의 직접 운영이 어려워질 때, 주로 유형 2의 방법을 택할 수 있으며, 대표적인 오픈소스 재단들이 설립된 2.0 시대 이후 많이 등장

표 3-2 주요 오픈소스 프로젝트 예시(유형 2)

프로젝트	개요
하둡	- 야후 소속 개발자 더그 커팅과 마이크 캐퍼렐라가 개발한 빅데이터 분산 처리 프레임워크 (2011년) - 아파치 재단에서 이 기술을 인수하여 운영 ⁶⁹⁾
스파크	- UC버클리의 AMPLab에서 개발된 클러스터 컴퓨팅 프레임워크 - 아파치 재단에 기부되어 유지 보수 ⁷⁰⁾

※ 출처: 저자 작성

- 그 외에도 아파치 인큐베이터(Apache Incubator)⁷¹⁾와 같이 재단이 초기 단계의 유망한 오픈소스 프로젝트들을 선정하여 성장을 지원하는 프로그램도 존재
- (유형 3: 창업) 오픈소스 기술을 기반으로 창업을 하여 타 기업에 인수되거나, 벤처 캐피탈 등의 지원을 받아 자체적으로 프로젝트를 운영함
- 오픈소스 프로젝트를 통해 수익을 창출하기 위해 창업을 선택할 수 있으며, 오픈소스 프로젝트 기반의 다양한 스타트업들은 자금을 유치하여 프로젝트를 운영하고, 기업에 인수되거나, 자체 사업을 통해 주식상장을 하기도 함

※ Crunchbase에 따르면, 2020년 12월 기준 오픈소스 스타트업(Open Source Startup)으로 분류된 기업은 총 229개이며, 총 자금 유치 횟수는 551회, 총 유치 자금은 66억 달러임⁷²⁾

표 3-3 주요 오픈소스 프로젝트 예시(유형 3)

프로젝트	개요
MySQL	- 마이클 위데니우스와 데이빗 액스마크가 개발(1994년) - MySQL AB를 창업하여 성장하다 썬 마이크로시스템즈에 인수되고(2008년) 오라클에 인수(2010년) ⁷³⁾⁷⁴⁾

※ 출처: 저자 작성

69) <https://intellipaat.com/blog/hadoop-creator-goes-to-cloudera/>
 70) https://en.wikipedia.org/wiki/Apache_Spark
 71) <http://incubator.apache.org/>
 72) <https://www.crunchbase.com/hub/open-source-startups>
 73) <https://www.w3resource.com/mysql/mysql-tutorials.php>
 74) <http://www.bloter.net/archives/142487>

- (유형 4: 기업 채택) 기업이 필요한 기술이 오픈소스로 공개되어 있을 경우, 기업은 그 기술을 채택하여 후원 및 협력 개발을 함
 - 기업이 오픈소스 프로젝트를 직접 인수하는 것이 아니라, 개발 참여 및 후원을 통해 해당 프로젝트에 대한 영향력을 강화

표 3-4 주요 오픈소스 프로젝트 예시(유형 4)

프로젝트	개 요
MXNet	<ul style="list-style-type: none"> - 카네기 멜론 대학교와 워싱턴 대학교 등 학계가 주축이 되어 개발⁷⁵⁾ - 아마존이 AWS를 위한 딥러닝 프레임워크로 채택(2017년)⁷⁶⁾ - 아파치 재단의 인큐베이터 프로그램에도 등록되었으며, 아마존에서는 주도적으로 프로젝트를 운영하고 개발에 참여

※ 출처: 저자 작성

□ (사례 분석) 기업이 직접 오픈소스 공개

- 오픈소스 3.0, 4.0 시대에는 기업이 직접 오픈소스 기술을 공개하는 일이 많아지고 있으며, 조사한 주요 프로젝트 중 오픈소스 3.0, 4.0 시대에 해당하는 프로젝트들은 대부분 이 경우에 해당함
- 기업들은 기술을 혁신하고 우호적 생태계를 조성하기 위해 자사 기술을 오픈소스로 공개하고 협업하여 개발하고 있음
 - 우수한 개발자들이 오픈소스 개발에 참여하여 기술 혁신 속도가 빨라질 수 있음
 - 기업들이 기술을 오픈소스 프로젝트로 공개하면, 다른 소프트웨어와의 호환성을 높여 우호적인 생태계를 조성할 수 있으며 기술 확산이 보다 쉬워짐
 - 오픈소스 프로젝트에 참여하는 개발자들을 채용하여 우수인력을 확보할 수 있음
- 기업이 기업 자체 기술 또는 인수한 기술을 직접 오픈소스 기술로 공개를 할 경우, (유형 5) 자체 운영을 통한 협력 개발, (유형 6) Alliance를 통한 주도적 협력 개발, (유형 7) 재단 설립 등의 방식을 통해 프로젝트가 운영될 수 있음

75) <https://www.infoworld.com/article/3149598/mxnet-review-amazons-scalable-deep-learning.html>

76) <https://aws.amazon.com/ko/blogs/aws/excited-about-mxnet-joining-apache/>



- (유형 5: 자체 운영을 통한 협력 개발) 기업은 자체적으로 오픈소스 프로젝트를 운영하여 다른 개발자 및 기업들과의 공동개발을 자사가 직접 주도할 수 있음
 - 기술을 오픈소스로 공개 후 프로젝트를 직접 재정적으로 지원하며 자사 개발자들이 프로젝트를 이끌도록 하는 등을 통해 외부와 협업하되 주도권을 유지하도록 함

표 3-5 주요 오픈소스 프로젝트 예시(유형 5)

프로젝트	개요
텐서플로우	- 구글이 오픈소스로 공개하여 자체적으로 운영하는 머신러닝 프레임워크 프로젝트(2015년) ⁷⁷⁾ - 2019년 기준 깃허브 기여자수 5위를 차지 ⁷⁸⁾
vscode	- 마이크로소프트의 자체 기술인 비주얼 스튜디오 코드를 오픈소스로 공개 ⁷⁹⁾ 하였으며, 직접 프로젝트를 운영(2015년) - 2019년 기준 깃허브 내 기여자수 1위 프로젝트에 해당
도커	- 닷클라우드사에서 공개한 컨테이너 자동화 기술(2013년) ⁸⁰⁾ - 자사 주도로 프로젝트를 운영하고 관련 사업을 운영
PyTorch	- 페이스북 AI 리서치 랩에서 개발한 머신러닝 프레임워크(2016년) ⁸¹⁾ ⁸²⁾ - 페이스북에서 직접 운영하고 있으며, AWS, 구글 클라우드, 마이크로소프트 Azure 등 여러 클라우드에서 서비스 제공

※ 출처: 저자 작성

- (유형 6: Alliance를 통한 주도적 협력 개발) 자사에서 개발을 주도하되, 오픈소스 프로젝트의 이해관계자 기업들과 Alliance를 구성하여 필요한 부분을 협력하여 개발

표 3-6 주요 오픈소스 프로젝트 예시(유형 6)

프로젝트	개요
안드로이드	- 구글이 안드로이드사 인수 후 공개한 모바일 OS (2008년) ⁸³⁾ - 통신사, 스마트폰 제조사, 반도체 회사, 응용프로그램 개발사 등으로 구성된 오픈핸드셋 얼라이언스(Open Handset Alliance)를 구성 ⁸⁴⁾ 하여 안정화와 신규 기능 개발에 참여하도록 함

※ 출처: 저자 작성

77) <https://www.wired.com/2015/11/google-open-sources-its-artificial-intelligence-engine/>
 78) <https://octoverse.github.com/2019/>
 79) <https://arstechnica.com/information-technology/2015/11/visual-studio-now-supports-debugging-linux-apps-code-editor-now-open-source/>
 80) <https://www.infoq.com/news/2013/03/Docker/>
 81) <https://github.com/pytorch/pytorch/releases/tag/v0.1.1>
 82) <https://www.oreilly.com/content/when-two-trends-fuse-pytorch-and-recommender-systems>

- (유형 7: 재단 설립 후 재단 주도 운영) 기업이 직접 운영하기 어려운 프로젝트이거나 다른 기업들의 적극적인 참여가 필요한 프로젝트의 경우, 기업은 재단을 설립하여 프로젝트를 이관하고 재단이 주도하여 운영하도록 함
 - 기반이 되는 기술들은 많은 기업들에서 사용되고 사실표준(De Facto Standard)이 되어 관련 생태계가 조성되는 것이 매우 중요하기 때문에 외부기업들의 보다 적극적인 개발 참여가 필요함
 - 재단을 설립하여 운영하면 보다 중립성과 독립성을 유지할 수 있어 다른 기업들의 참여를 유도할 수 있음

표 3-7 주요 오픈소스 프로젝트 예시(유형 7)

프로젝트	개요
쿠버네티스	<ul style="list-style-type: none"> - 구글이 시드기술을 발표(2014년)하고 리눅스 재단과 함께 클라우드 네이티브 컴퓨팅 재단(CNCF)을 설립(2015년) - 재단을 중심으로 빠르게 성장하여 시장 표준으로 자리매김하고 CNCF를 졸업⁸⁵⁾
오픈스택	<ul style="list-style-type: none"> - 미 항공우주국과 렉스페이스社가 발표한 클라우드OS (2010년) - 2012년 오픈스택 재단(現 오픈인프라스트럭처 재단)을 설립하여 운영⁸⁶⁾
클라우드 파운드리	<ul style="list-style-type: none"> - VM웨어에서 처음 개발한 멀티 클라우드 애플리케이션 PaaS - 피보탈 소프트웨어로 이전되었다가 2015년 리눅스 재단의 협업 재단인 클라우드 파운드리 재단이 만들어지고 소유권이 이전⁸⁷⁾
이클립스	<ul style="list-style-type: none"> - 1990년대 IBM이 처음 개발한 개발환경으로, 2001년 여러 소프트웨어 개발 업체들과 컨소시엄을 구성하여 기술 개발 진행 - 더 많은 기업과 개발자들이 참여하도록 하기 위해 오픈소스로 전환하고 재단을 만들어 운영(2004년)⁸⁸⁾

※ 출처: 저자 작성

83) http://www.openhandsetalliance.com/press_102108.html

84) http://www.openhandsetalliance.com/press_110507.html

85) “쿠버네티스가 클라우드 네이티브 컴퓨팅 재단을 졸업해 본격적인 오픈소스 프로젝트가 되다”, 클라우드 이슈리포트 (2018.04.19.),

86) <https://www.businesswire.com/news/home/20120919005997/en/OpenStack-Launches-Independent-Foundation-Begins-Work-Protecting>

87) <https://www.forbes.com/sites/benkepes/2014/12/09/cloud-foundry-foundation-matures-becomes-a-linux-foundation-collaborative-project/?sh=700400fb452b>

88) <https://terms.naver.com/entry.nhn?docId=3580755&cid=59088&categoryId=59096>



□ 성장 유형의 특징 : 개발자, 재단, 기업의 선택적 협력

- 개발자 또는 학계에서 공개한 오픈소스 프로젝트의 경우에는 보다 큰 주체의 도움을 받아 오픈소스 프로젝트 및 커뮤니티를 더욱 성장시키는 방법을 채택한 것으로 분석됨
 - 오픈소스 커뮤니티가 일정 규모 이상 성장하게 되면, 개인 개발자, 개발자 그룹, 학계 단위에서는 지속적인 관리와 운영이 어려울 수 있으며, 보다 체계적인 관리와 법적·재정적 지원이 필요하기 때문
- 재단을 통하는 방식(유형 1 & 2)과 기업을 통한 방식(유형 3 & 4)를 택할 수 있으며, 프로젝트의 성격 및 방향성에 따라 유형을 선택할 수 있음
 - 유형 1, 2와 같이 재단을 직접 설립하여 공동으로 운영하거나, 재단에 편입하여 지원을 받는 등 재단을 통한 성장 방식이 있으며, 주로 비영리적인 프로젝트에서 이러한 방식을 택할 수 있음
 - 또는, 유형 3, 4와 같이 창업 후 기업에 인수되거나, 프로젝트 자체가 기업에 채택되는 등 기업의 지원을 받는 방식으로 성장할 수도 있으며, 이윤을 추구하거나 기업에서의 기술의 실제적인 활용을 원하는 경우 유리할 수 있음
 - ※ 특히, 유형 3, 4와 같이 기업의 영향권 안으로 들어가는 경우에는 추후 해당 기업의 전략에 따라 유형 5, 6, 7과 같은 기업의 성장 방식을 통해 성장할 수도 있음

그림 3-2 개발자·학계 오픈소스 프로젝트의 성장 유형 분류



※ 출처: 저자 작성

- 또한, 기업이 직접 공개한 오픈소스 프로젝트의 경우, 협력의 정도와 주도권의 정도에 따라 성장의 유형이 나뉘어 알 수 있었음
 - 그림 3-2와 같이 유형 5에서 유형 7로 갈수록 협력의 용이성은 보다 높아지나, 기업의 기술 주도권은 다소 낮아지게 됨

- 기업이 오픈소스 프로젝트를 공개하고 성장시킬 때에는 얼마만큼의 협력이 필요한 기술인가, 얼마만큼의 기술 주도권이 필요한 기술인가에 따라 성장 방식 유형에 대한 전략적인 선택이 필요함
 - 다른 기업 및 개발자들은 기업이 오픈소스 프로젝트를 공개하더라도 해당 기술을 사유화하거나 경쟁 상품 및 서비스에 활용할 것이라는 우려 때문에 개발 참여에 거부감을 느낄 수 있음
 - 외부 기업들의 기업 오픈소스 프로젝트에 대한 참여를 보다 높이기 위해서는 이러한 우려를 낮추고 Alliance 구성 또는 재단 설립 등의 방법을 통해 협력을 용이하도록 만드는 것이 중요할 것
 - 한편, 기업은 오픈소스 프로젝트를 통해 수익을 창출하여야 하기 때문에 오픈소스로 기술을 공개하더라도 이에 대한 주도권을 유지하는 것도 중요하며, 따라서 전략 프로젝트의 성장 방식을 선택할 필요가 있음

그림 3-3 유형별 협력의 용이성 및 기술 주도권 변화



※ 출처: 저자 작성



2 협업개발 플랫폼의 발전과 협력 방식의 진화

□ 분석의 개요

- 본 파트에서는 오픈소스 생태계에서 협력 개발의 중심이 된 소스코드 저장소의 발전 과정과 최근 오픈소스 생태계의 중심이 된 깃허브에 대해 살펴보고, 저장소의 발전이 오픈소스 생태계의 발전에 어떠한 영향을 미치고 있는지 분석해 봄

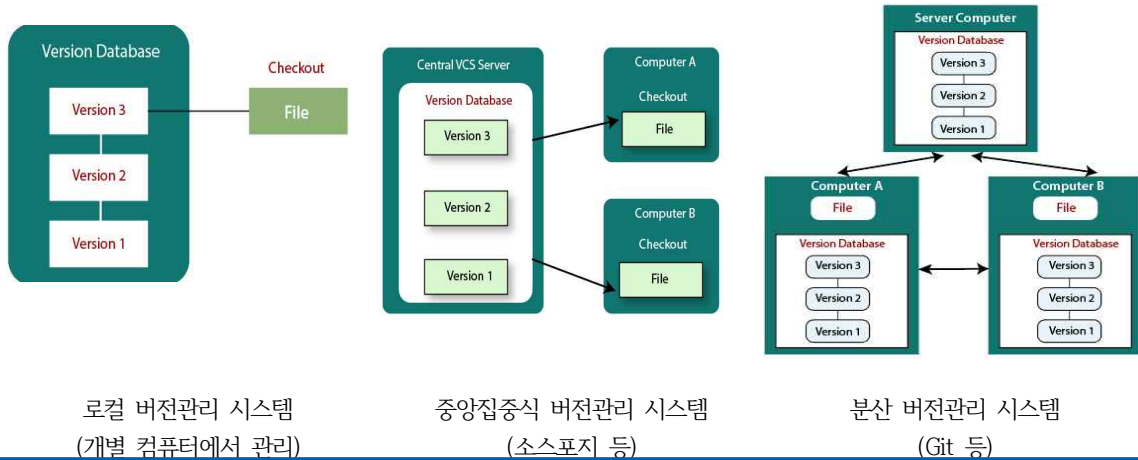
□ 협업 기술의 발전 - 버전관리 시스템⁸⁹⁾

- (협업개발 플랫폼-저장소의 발전) 소스코드를 공유하고 프로젝트를 관리하는 방식이 중앙집중식에서 온라인기반 분산버전관리 시스템으로 발전하면서 안전하고 효율적인 협업관리와 프로젝트의 확장이 용이해짐
 - (~오픈소스 1.0 시대) 1960~70년대에는 유닉스 소스코드 설명서가 책자나 디스크로 배포되다가, 인터넷과 개인용 컴퓨터가 보급되면서 인터넷을 통해 유포됨
 - (오픈소스 2.0 시대) 2000년대 초반까지 소스포지라는 중앙집중식 저장소가 가장 인기가 있었고 구글과 마이크로소프트 등의 대기업들은 자체저장소를 운영함
 - (오픈소스 3.0~4.0 시대) 2006년 깃이 개발되고 2008년 깃허브가 온라인 기반 으로 제공되자, 깃허브를 중심으로 개발자들과 기업이 모이는 네트워크 효과가 커지면서 SW의 메카로 자리매김
- (시스템별 특징) SW개발 버전관리 시스템⁹⁰⁾은 개별 컴퓨터에서 개발자별로 운영될 수도 있으나, 협업을 위해 중앙에서 관리하다가 위험분산 및 로컬 개발의 효율성을 높여주는 분산관리 방식으로 진화함
 - 중앙 시스템에 모든 작업의 정보를 모으고 개발자들을 관리하는 중앙집중식 버전관리 시스템은 관리는 편리하나 중앙 서버에 이상이 생길 경우, 프로젝트 운영이 어려워지며 중앙의 관리를 받아야 하므로 로컬 개발에 제약이 있음
 - 분산버전관리 시스템은 모든 사람이 자신의 저장소를 가지고, 모든 사항에 대해 중앙 서버와 상호작용할 필요가 없기 때문에 로컬 작업과 공동작업을 모두 효율화 할 수 있으며, 위험요인도 분산시킬 수 있음 (ex, Git, Mercurial)

89) Javatpint.com 홈페이지 Git 튜토리얼, 조만석(2020) ETRI 발표자료 등을 기반으로 정리

90) 버전관리시스템의 주요 기능은 파일의 변경 내역을 기록하고, 추적할 수 있게 하며, 분기하거나 병합이 가능하게 하며 동시 작업을 지원함

그림 3-9 SW 버전 관리 시스템의 진화



※ 출처 : Javatpint.com 홈페이지, Git 튜토리얼 자료 이미지 인용

□ 협업을 위한 노력 - 소스코드 저장소들⁹¹⁾

- 버전관리 시스템의 발전과 함께 1) 상용 소스코드 저장소가 급성장하고 있으나 2) 기업별로 자체 저장소를 운영하거나, 3) 국가에서 정책적으로 저장소를 지원하기도 함
- (상용 저장소의 발전) 깃랩, 비트버킷 등의 상용저장소들은 깃허브와 차별적인 기능을 지원하고 있어 개발 프로젝트의 목적 및 성격에 따라 깃허브와 깃랩 등을 병행하여 사용하는 경우가 많음
 - 중앙집중식 버전관리시스템을 이용한 소스포지가 2000년대 초반에는 가장 인기 있는 저장소였으나,
 - 분산버전관리 시스템을 온라인으로 구현하는 깃허브, 깃랩, 비트버킷 등이 출현하고 개발자 협업을 지원하는 새로운 기능들이 추가되면서 발전하고 있음
- ※ 소스포지는 2008년 약 24만개의 프로젝트가 등록되며 가장 인기 있는 저장소였으나, 수익원 확보를 위해 정크웨어 끼워 넣기 등 무리한 정책으로 개발자들이 떠남(2015년)
- (상용저장소 -깃랩) 깃랩社(GitLab Inc.)가 2011년 선보인 오픈소스 저장소로 깃을 활용하며, 이슈 추적, 보안성 테스트 등의 기능을 갖춘 웹기반 데브옵스 플랫폼임
 - 비공개 소스인 깃허브와 달리 깃랩 자체가 MIT 라이선스의 오픈소스로 공개
 - 깃랩은 유럽의 깃허브로 불리우며 유럽 입자물리연구소(CERN), 유럽 우주기구 등에서는 내부 코드 관리는 깃랩을 이용하고, 외부 공개 코드 및 데이터는 깃허브에 배포함

91) 위키백과 깃랩, 소스포지, 비트버킷 정보 및 각사 홈페이지 정보 기반 정리



- ※ 깃랩社は 2020년 5월 기준, 66개국 1300여명의 직원과 2,000명 이상의 오픈소스 기여자들이 있음. 골드만삭스, IBM, 소니, NASA, 알리바바, 스페이스X, 유럽 입자 물리 연구소 등 100,000개 이상의 단체에서 사용되고 있음. 현재 구글 벤처스, 골드만 삭스 등이 주요 투자자로 있으며 기업 가치 27억 달러로 평가 받는 유니콘 기업임
- (국가 지원 저장소-중국) 중국에서는 깃허브도 많이 사용하나 자국내에서는 기티(Gitee), 코딩닷넷(Coding.net) 등의 자국의 오픈소스 저장소를 많이 사용하고 있음
 - 기티는 중국 최대 오픈소스 커뮤니티인 오픈소스 차이나(Open Source China, OSChina)에서 운영하고 있으며, 현재 5백만 명 이상의 개발자와 10만여 개의 기업 고객이 기티를 사용하고 있음
 - ※ 깃허브가 미국의 바뀐 수출 관리 규정(EAR)으로 인해 쿠바, 북한, 크림반도, 시리아 등에 거주하는 사용자들의 일부 권한을 사전 고지 없이 제한함. 위기의식을 느낀 중국 기업들이 중국 중심의 오픈소스 저장소의 필요성을 제기, 이에 따라 중국 산업정보기술부는 '2020년 오픈소스 호스팅 플랫폼 프로젝트'를 시작하고 기존에 운영되고 있던 기티를 공식 호스팅 서비스로 지정함
- (기업 자체저장소) 기업들은 대형 프로젝트로 육성을 위해서는 외부 개발자들의 접근성이 좋은 깃허브 등의 저장소를 사용하지만, 필요에 따라 자체적인 오픈소스 저장소를 운영하기도 함
 - 구글은 깃허브와 함께 Git-on-borg라는 자체 깃 기반 오픈소스 저장소를 내부와 외부 호스팅에 활용, 안드로이드와 Chromium을 이 서비스를 통해 운영함⁹²⁾
 - 텐센트는 코딩닷넷이라는 소스코드 관리 도구를 개발하여(2014년), 2백만 명 이상의 개발자와 5만 개 이상의 기업 고객이 이용하고 있음
 - 그 외에 AWS CodeCommit, Phabricator, Gogs, Gitea, Apache Allura 등의 다양한 오픈소스 저장소들이 있음

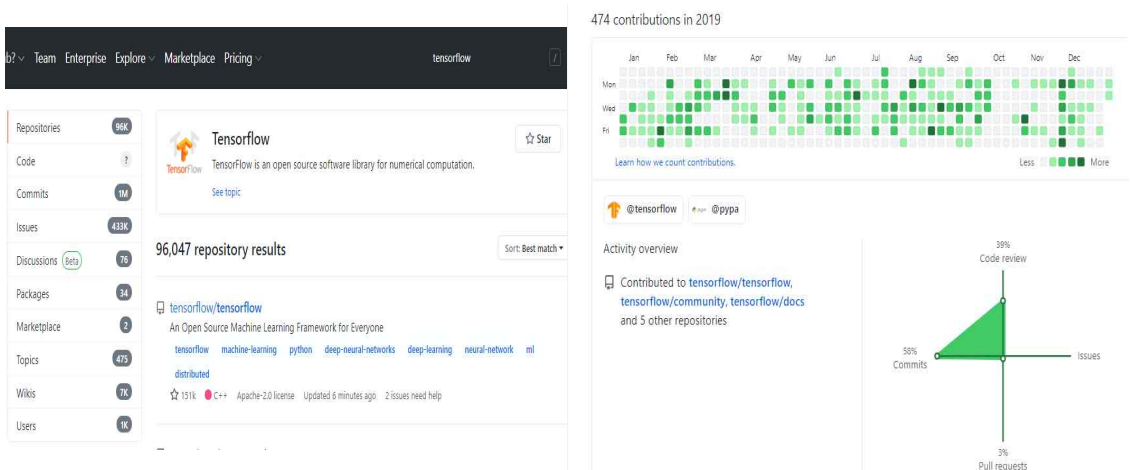
□ 글로벌 협업개발의 중심지 - '깃허브'의 성장

- (깃허브 특징) 2008년 시작된 분산버전 관리 툴인 깃(Git)을 사용하여 오픈소스 협업 개발을 지원하는 웹호스팅 서비스로, 다양하고 편리한 기능들을 제공함
 - 깃이 텍스트 명령어 입력 방식인 데 반해, 깃허브는 화려한 그래픽 유저 인터페이스를 제공
 - 다른 저장소에 있는 코드를 자기 계정의 저장소로 복사하는 포크(Fork) 기능과 수정 사항을 원래 프로젝트에 반영할 수 있도록 하는 pull request 기능 제공

92) Google Open Source blog, <https://opensource.googleblog.com>

- 깃허브 게시판 등을 통해 의견을 공유하고 조율하여 언제 어디서나 협업이 용이해짐
- 깃허브는 아래 그림과 같이 개발자별 코드 기여분에 대해 철저하게 버전을 관리함으로써 투명성과 공정성을 확보하여, 실력 있는 개발자가 코드 기여를 많이 할수록 인정받을 수 있는 체계를 제공함
- 개발자들은 깃허브에 공개된 기업들의 프로젝트 참여를 통해 자신의 개발 능력을 홍보할 수 있고, 기업들은 유능한 개발자를 발굴하고 채용하는 루트로 활용

그림 3-10 깃허브에 기록되는 개인별 컨트리뷰션 그래프



※ 출처: www.github.com

- (MS의 깃허브 인수) 마이크로소프트가 깃허브를 75억 달러에 인수(2018년)하고 애저 등 자사의 기술개발에 깃허브를 적극적으로 활용하고 있음
 - (깃허브의 이유) 2천700만 명에 달하는 개발자들이 활동하는 최대 커뮤니티이지만 지속되는 적자로 경영난을 겪고 자생 방안을 찾지 못함
 - (MS의 이유) 애저와 깃허브를 결합하여 개발자들이 클라우드 환경에서 수월하게 깃허브를 사용하고, 깃허브 개발자들의 애저 등 MS 프로젝트 접근 용이
 - (개발자 반발) 마이크로소프트가 개발자의 자유로운 활동을 침해할 것이라는 우려로 인해 일부 개발자들과 프로젝트들이 깃랩으로 이동하기도 하였음
 - (인수 이후) 2017년 2,400만명이던 개발자수는 2019년 4천만명으로 증가하며 급성장하고 있으며, 마이크로소프트는 코드리뷰 협업툴 Pull Panda를 인수하여 무료로 기능을 제공하고(2019), 개발자 후원시스템을 추가하는 등 깃허브 기능을 강화



□ 성장과 협력의 심화 : 협력의 광역화, 자기조직화, 내부화

- 깃허브가 글로벌 오픈소스 생태계의 중심으로 성장하면서 깃허브의 협력 방식은 오픈소스 생태계의 협력방식으로 전 세계 개발자들에게 자리매김해가고 있음
- 깃허브 등 오픈소스 저장소는 ① 개방성 ② 투명성 ③ 용이성 ④ 분산과 통제의 균형 ⑤ 네트워크 외부성 등의 특성으로 오픈소스 생태계 확장에 기여함
 - ① 개방성 : 온라인기반으로 누구에게나 무료로 개방되고 특별한 절차 없이 깃허브에 올려진 프로젝트의 현황과 개발자 정보 등을 보고 쉽게 참여할 수 있음
 - ② 투명성 : 프로젝트 진행현황과 개발자별 기여실적이 저장소에서 투명하게 집계되고 공개되어 실력 있는 개발자들이 프로젝트 참여 여부를 결정하고 기여도를 인정받기 쉬워면서 개발자들의 참여가 증가하게 됨
 - ③ 용이성 : 프로젝트와 개발자에 대한 정보를 간결하고 유려한 그래픽으로 전달하며, 개발자 협업 환경이 쉽고 편리하여 접근 가능성이 높아짐
 - ④ 분산과 통제의 균형 : 원(原) 프로젝트에 로컬 개발자의 변경 본만 반영시켜주는 기능 등으로 개발자가 분산되어 각자 개발하더라도 이를 프로젝트 리더가 통제하고 관리할 수 있으므로 개발자가 많아지고 프로젝트 규모가 커질 수 있음
 - ⑤ 네트워크 외부성 : 기업들이 프로젝트를 저장소에 공개하고, 이에 참여하기 위해 개발자들이 모이고, 모인 개발자들을 활용하려는 기업들이 더 많은 프로젝트를 공개하는 등 네트워크 외부성이 생기면서 성장이 가속화됨
- 위의 5가지 오픈소스 저장소의 특징은 협력 개발을 다음과 같이 진화시킴
 - (협업의 광역화) 참여 및 협업 범위의 제약이 없어지고 기업간, 글로벌 협업 증가
 - (자기조직화) 프로젝트의 관리가 쉬워지고 커뮤니케이션이 쉬워지면서 기업과 기업, 기업과 개발자, 개발자와 개발자 등 다양한 조합의 협력이 쉽게 조직화됨
 - (외부자원의 내부화) 저장소를 통해 기업 외부의 개발자 및 타 기업 프로젝트의 정보 및 기술력을 파악하고, 협업 개발하다가 사업이 본격화되면 필요한 개발자를 직접 채용하거나 필요한 기업을 인수합병하게 되는 사례 증가

3 오픈소스 생태계 경쟁 구도의 변화

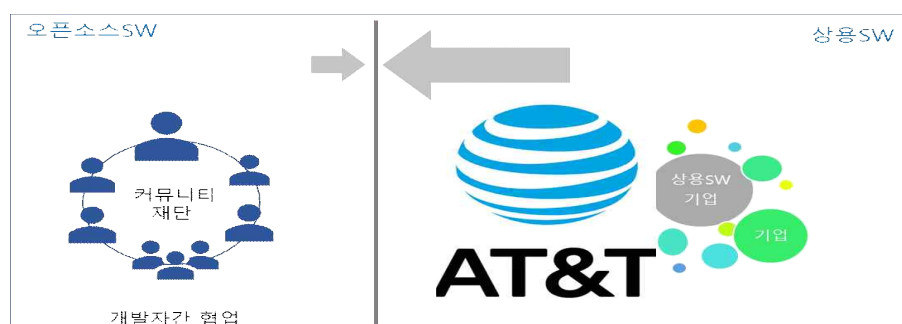
□ 분석의 개요

- 본 파트에서는 II장에서 설명한 오픈소스 1.0~4.0까지의 대표적인 오픈소스 프로젝트 및 기술들 중 특징적인 주요 사례에 대해 협력 및 경쟁구도를 유형화함
 - 주요 오픈소스 프로젝트에 대한 문헌, 홈페이지, 위키피디아 등 인터넷에 소개된 프로젝트들의 연혁 및 시장 경쟁구도에 대한 자료를 기반으로 분석함

□ 주요 사례 분석

- (운영체제 : 오픈소스 개발자 vs. 상용SW 기업) AT&T 등 소프트웨어 저작권을 주장하며 상업화를 추진하는 상용SW 기업들과 이에 맞서는 개발자 커뮤니티간의 대립 구도⁹³⁾
 - 1980년 미연방법규에서 컴퓨터 프로그램의 저작권을 인정하는 법개정이 이루어지자 소프트웨어 상용화 시작, AT&T는 사실상 무료로 배포하던 유닉스를 판매
 - ※ AT&T의 유닉스를 무료로 배포해오다 상용 유닉스 System V는 10만불(1998년 기준)에 판매
 - ※ GNU는 저작권의 개념으로 소스코드를 공유하지 않겠다는 선택을 막는 GPL 라이선스를 발표
 - 리처드 스톨만(1983년)이 유료의 유닉스를 대체하기 위한 GNU 프로젝트를 개발하여 공개하고, 리누스 토발즈(1991년)가 개발한 리눅스를 공개하는 등 개발자 리더를 중심으로 해커 문화의 영향을 받은 개발자들이 협력 개발을 추진
 - ※ 개발자들(기업소속 포함)이 소스코드 테이프/디스크(150불 수준)를 구입, 오픈소스를 재정적으로 후원

그림 3-4 오픈소스 1.0 시대의 협력 및 경쟁 구도 - 운영체제의 경우



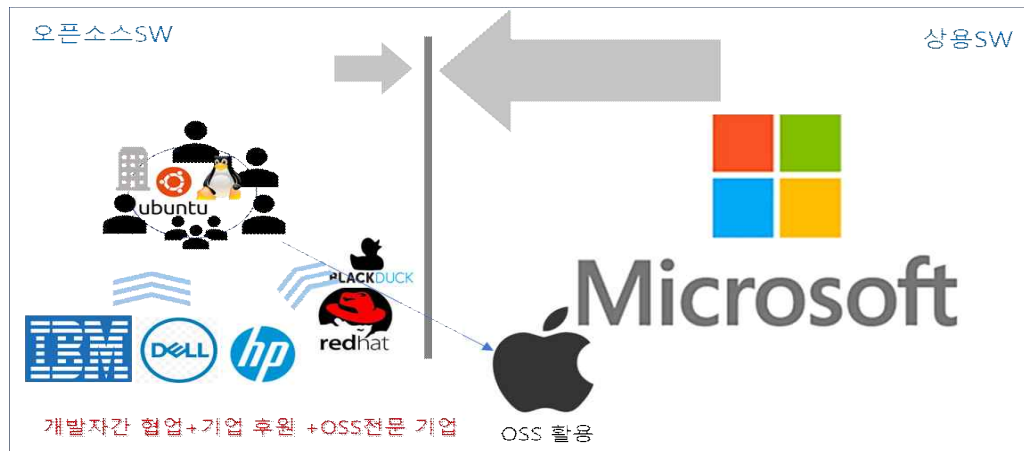
※ 출처: 저자 작성

93) 크리스토퍼 토찌(2017), '프리-오픈소스 소프트웨어 혁명의 역사'의 내용을 기반으로 재구성함



- (PC 운영체제 : 오픈소스 개발자&지원기업 vs. 상용SW기업) 하드웨어 기업들이 자사 하드웨어에 리눅스나 우분투를 무료/저가에 탑재하고 판매하여 오픈소스 진영을 지원하고, 상용 SW를 판매하는 마이크로소프트에 대항하는 구도 형성⁹⁴⁾
 - 개인용 컴퓨터(PC) 시장이 성장하면서 일반 이용자 편의성이 높은 마이크로소프트의 윈도가 시장에서 급성장하는 한편,
 - 윈도에 대항하는 오픈소스SW 리눅스와 우분투가 공개되어 개발자들 중심으로 개발되고 IBM, Dell, HP 등이 PC에 이를 탑재하여 배포함
 - ※ 2020년 6월 기준 전세계 운영체제 시장점유율은 1위 윈도우즈(77.74%), 2위 MacOS(17.07%), 3위 리눅스(1.85%), 4위 크롬OS(0.67%), 5위 기타(2.76)로 상용SW가 지배적임⁹⁵⁾
 - ※ 마이크로소프트 등 상용SW 기업들은 PC를 이용하지 않던 새로운 계층을 사용자로 끌어들이기 위해 PC에 익숙하지 않은 고객의 의견을 반영하여 윈도우를 개발하는 반면, 리눅스 개발자들은 가장 똑똑한 고객의 의견을 반영하여 전문가 지향적으로 평가됨⁹⁶⁾
 - 오픈소스SW는 전문 개발자들만 사용할 수 있었으나, 전문적으로 교육/관리하는 기업 레드햇 등이 저렴한 비용으로 오픈소스SW를 배포/지원하고, 블랙덕이 라이선스로 인한 법적 분쟁을 해결해주면서 오픈소스SW 사용계층이 확대됨
 - 애플은 오픈소스SW 유닉스 소스코드를 기반으로 매킨토시 컴퓨터를 위한 macOS를 개발하고 이를 공개하지 않고 상업적으로 활용함(일부 오픈소스 포함)⁹⁷⁾

그림 3-5 오픈소스 2.0 시대의 협력 및 경쟁 구도 : PC 운영체제의 경우

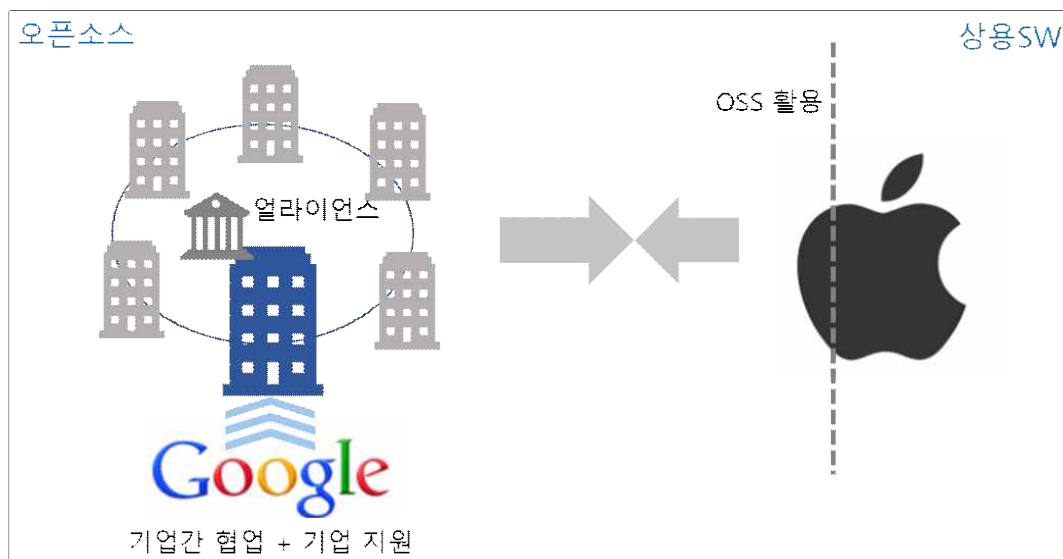


※ 출처: 저자 작성

94) 크리스토퍼 토찌(2017), 리눅스, 우분투, 레드햇 홈페이지 등의 자료를 기반으로 재구성
 95) Statista (2020), <https://www.statista.com/statistics/218089/global-market-share-of-windows-7/>
 96) 조쉬 러너, 마크 생커멘 (2010), '오픈소스와 소프트웨어 산업, 상생의 경제학', 99쪽 기반 정리
 97) 위키백과, 'macOS'.

- (모바일OS : 구글진영 vs. 애플) 모바일 OS는 오픈소스SW 중심의 개방형 생태계를 구축한 안드로이드 진영과 애플 진영과의 경쟁구도로 안드로이드가 80% 이상 점유함
 - 2005년 구글은 작은 안드로이드社를 인수하고, 2007년 오픈 핸드셋 얼라이언스(OHA)에서 모바일 기기 공개 표준으로 안드로이드를 발표, 2008년 안드로이드를 오픈소스로 선언하고 아파치 라이선스로 공개함⁹⁸⁾
 - ※ 안드로이드는 구글에 의해 개발되고 있으며, 안드로이드 오픈소스 프로젝트(Android Open Source Project, AOSP)에서 공개하여 누구나 이용 가능
 - ※ OHA는 2007년 11월 구글을 중심으로 34개 모바일 핸드셋 제조업체와 애플리케이션 개발자, 통신사, 반도체 제조업체 등이 마이크로소프트와 애플에 대항하기 위해 함께 설립, 현재 96개 기업 및 단체가 참여
 - 안드로이드는 상업용으로 개발한 SW의 소스코드 공개 의무가 없는 아파치 라이선스를 채택하여 상용화가 용이하지만, 안드로이드라는 명칭을 사용하고 구글 플레이스토어 등을 이용하기 위해서는 구글 인증을 받아야 함
 - 애플은 macOS(OS X)를 기반으로 모바일 운영 체제 iOS를 개발하고, 아이폰에 탑재하여 판매하며 SW 대부분을 클로즈드소스로 운영하되, 애플의 소프트웨어 개발 키트인 SDK를 공개하고⁹⁹⁾ 저작도구 Swift는 오픈소스로 운영하고 있음¹⁰⁰⁾

그림 3-6 오픈소스 3.0 시대의 협력 및 경쟁 구도 : 모바일 플랫폼의 경우



※ 출처: 저자 작성

98) 위키백과, '안드로이드(운영 체제)', [https://ko.wikipedia.org/wiki/안드로이드_\(운영_체제\)](https://ko.wikipedia.org/wiki/안드로이드_(운영_체제))

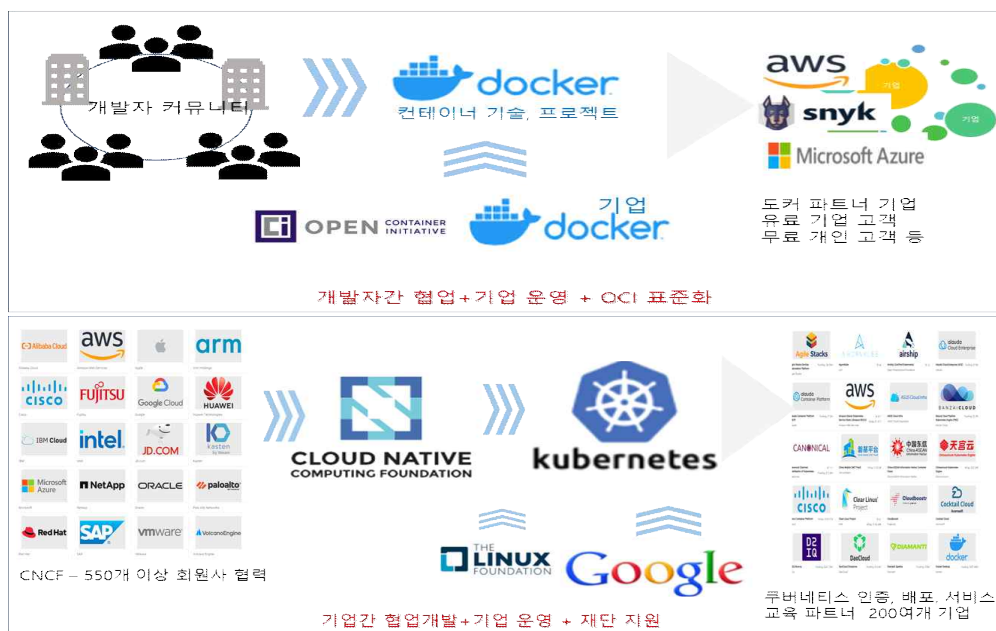
99) 위키백과, 'iOS'와 'iOS SDK' 기반 정리

100) <https://developer.apple.com/kr/swift/>



- (클라우드 기술 : 오픈소스 시장 표준) 온프레미스(On-premise) 기업환경을 클라우드 기반으로 전환해가면서 오픈소스SW 도커(Docker)와 쿠버네티스(Kubernetes)가 시장 지배적 기술로 자리잡고 다양한 기업이 공동개발하고 공동활용하는 구도 형성
 - 기업별로 서버를 구축/운영하던 온프레미스 환경이었으나 IT혁신과 함께 비즈니스 민첩성에 대한 니즈가 높아지면서 클라우드 환경으로 급속히 전환하고 있음
 - ※ 도커는 기업이 운영해야 하는 서버가 여러대로 증가할 때 이들을 효과적으로 운영하기 위한 기술로, 각 서버의 소프트웨어를 컨테이너라는 표준화된 유닛으로 패키징하여 애플리케이션을 신속하게 구축, 테스트, 배포할 수 있게 하는 플랫폼¹⁰¹⁾
 - ※ 쿠버네티스는 컨테이너화된 애플리케이션을 자동으로 배포, 스케일링 및 관리해주는 오케스트레이션 플랫폼 및 관리 시스템
 - 도커 프로젝트는 도커기업이 적극적으로 개발자 커뮤니티를 지원하고 도커 서비스를 유료/무료로 배포하여 컨테이너 기술의 시장표준으로 자리매김함
 - 쿠버네티스는 구글이 개발하여 공개고 CNCF 재단을 중심으로 수백개의 기업이 협력 개발하고, 200여 개의 기업이 파트너가 되어 확산시켜 시장표준이 됨
 - ※ RedMonc(2017.9)에 따르면 Fortune100대 기업의 71%가 컨테이너를 사용하고 50% 이상은 쿠버네티스를 컨테이너 오케스트레이션 플랫폼으로 사용하고 있다고 함

그림 3-7 오픈소스 3.0~4.0 시대의 협력 구도 : 클라우드 기술의 경우

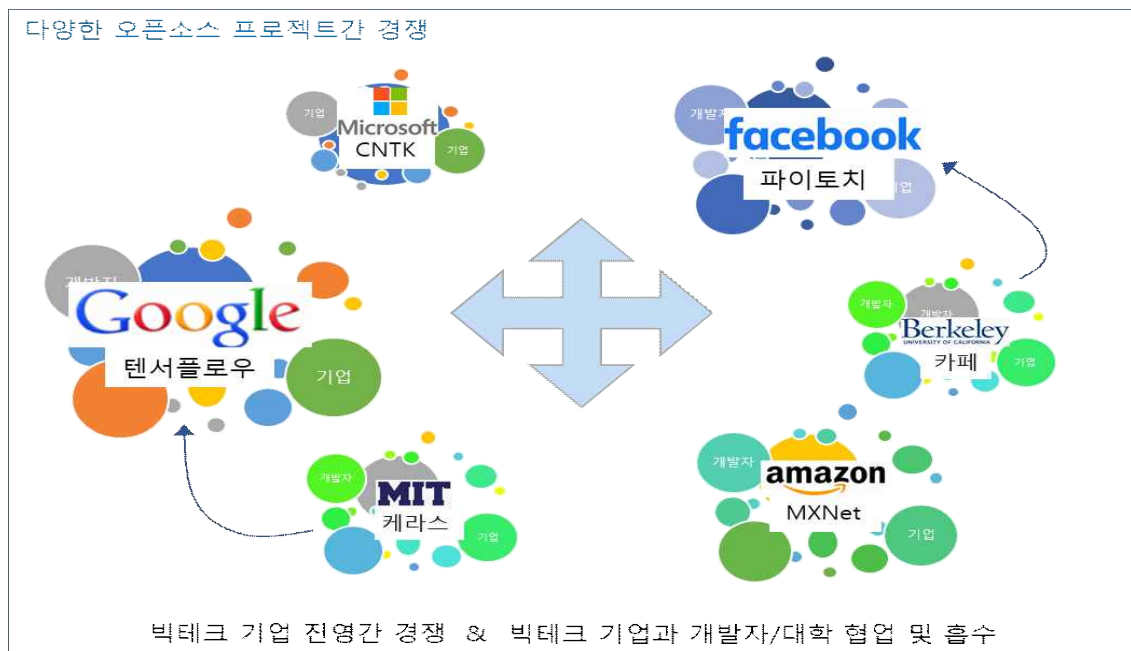


※ 출처: 도커, 쿠버네티스, CNCF, OCI 홈페이지 자료를 기반으로 저자 작성

101) <https://aws.amazon.com/ko/docker/>, 위키백과 ‘도커’ 등을 참조로 정리

- (딥러닝 프레임워크 : 오픈소스내 빅테크 진영간 경쟁) 인공지능 기술 중 가장 빠르게 발전하고 있는 딥러닝 프레임워크 기술들은 빅테크 기업별로 기술을 개발하고 오픈소스 프로젝트를 지원하면서 빅테크 기업 중심의 진영간 경쟁 구도 형성¹⁰²⁾
 - 구글의 텐서플로우, 페이스북의 파이토치, 마이크로소프트의 CNTK(Computational Network Toolkit), 아마존의 MXNet 등은 빅테크 기업들이 직접 개발을 주도하거나 오픈소스 프로젝트를 지원하며 경쟁하고 있음
 - MIT 대학(케라스), 버클리 대학(카페), 스카이마인드(DL4J) 등 대학 및 스타트업들도 다양한 딥러닝 기술을 개발하고 있으며, 딥러닝 기술은 아직 기술적 한계가 많아 당분간 이러한 한계 극복을 위해 다양한 주체들의 협력적 기술개발이 주를 이룰 것이나,
 - 버클리 대학의 카페²⁾는 파이토치(페이스북)로 흡수되고, 케라스는 텐서플로우 및 MXNet과 호환성을 확보하면서, 주도적 사업자의 영향력이 커지고 있으며,
 - 다양한 딥러닝 기술들이 기능상으로 차별적 특징을 보유하고 있음에도 불구하고 개발자 및 사용자들이 대규모 커뮤니티가 형성된 기술에서 상호작용을 통해 얻는 이점이 크기 때문에 네트워크 외부성이 커지는 빅테크 기업으로의 쏠림 현상이 커질 전망

그림 3-8 오픈소스 4.0 시대의 협력 및 경쟁 구도 : 딥러닝 기술의 경우



※ 출처: 저자 작성

102) 최성호 (2020), '4차 산업혁명의 숨은 원동력, 오픈소스 현황과 시사점', KDB 산업은행 산은조사월보, 2020.1 제770호, 中 딥러닝 분야 주요 오픈소스 프로젝트 내용을 기반으로 저자 정리



□ 경쟁구도 변화의 특징 : 저항운동에서 오픈소스 진영간 경쟁으로

- 시대별 대표적인 오픈소스 주요 기술들의 협력 및 경쟁 구도에 대한 사례를 분석해 본 결과, 다음과 같은 변화의 특징을 볼 수 있었음
 - (협력 주체의 변화) 오픈소스 1.0 시대에는 개발자들이 협력의 주체이자 중심이었으나, 오픈소스 4.0 시대에는 기업들이 협력의 중심에 자리매김하고, 오픈소스에 대한 직접지원 (재단설립, 커뮤니티 운영 등)을 확대해감
 - (경쟁구도) 초기에는 오픈소스 진영과 거대 상용SW기업 간의 대결 구도였으나, 오픈소스 진영에 대기업들이 적극 참여하면서 오픈소스 프로젝트간 경쟁 구도로 변화

표 3-8 협력 주체 및 경쟁구도 요약

구분		OSS 협력 주체 (OSS 진영 구성원)	대결 및 경쟁 구도
사례	시대		
운영체제	OSS 1.0	개발자들	오픈소스진영 vs. AT&T(상용SW기업)
PC 운영체제	OSS 2.0	개발자들+지원기업들	오픈소스진영 vs. MS(상용SW기업)
모바일OS	OSS 3.0	구글+기업들	오픈소스진영 vs. 애플(상용SW기업)
클라우드(도커)	OSS 3.0	도커社+개발자	시장표준 (오픈소스 진영내 경쟁)
클라우드(쿠버네티스)	OSS 4.0	구글+기업들	시장표준 (오픈소스 진영내 경쟁)
딥러닝 프레임워크	OSS 4.0	빅테크+개인/소기업	오픈소스 內 빅테크 진영간 경쟁

※ 출처: 저자 작성, OSS: 오픈소스SW

- 협력 및 경쟁 구도는 사례별로 모두 다른 특징을 가지는데, 이는 기술의 특성/성숙도와 참여 기업/구성원들의 전략 등 다양한 변수에 따라 결정되기 때문임
 - 따라서 오픈소스로 기술을 공개하고 성장하려는 기업/기관은 해당 기술 생태계 현황, 특히 영향력이 큰 리딩 플레이어의 전략을 분석하고 이에 대응하는 협력대상 및 경쟁의 대상을 설정하는 등 생태계 전략에 기반한 오픈소스 전략이 필요
 - 특히 핵심기술 개발 역량이 있는 주체(기업/기관)와 이 기술의 지속적인 성장 발전을 지원할 수 있는 주체(기업/기관)간의 협력 관계 형성을 위한 전략이 필요함
 - 오픈소스 생태계를 활성화하려는 정부 및 공공기관은 개발자 커뮤니티 지원 등의 일반화된 지원책을 넘어 기술별 생태계 현황을 고려한 전략적이고 다각적인 지원책 마련이 필요함

IV 결론 및 시사점

□ 오픈소스 진화 단계별 특징 : 저항 - 조직화 - 자생적 성장 - 혁신주도

- 본 연구는 오픈소스SW 생태계의 주요사건과 특징을 바탕으로 4개의 시기로 구분하고, 각 시기별 변화 동인과 전개과정, 주요 변화내용을 정리함

표 4-1 오픈소스 진화 단계 특징

구분		오픈소스 1.0 (태동기)	오픈소스 2.0 (확립기)	오픈소스 3.0 (촉진기)	오픈소스 4.0 (성장기)
시기	구분	GNU선언 이후 (1983~1997)	OSI 창설 이후 (1998~2005)	깃 공개 이후 (2006~2014)	텐서플로우 공개 이후 (2015~ 현재)
	특징	<ul style="list-style-type: none"> SW사유화에 저항한 자발적 참여운동 	<ul style="list-style-type: none"> 라이선스 분화 커뮤니티 조직화 	<ul style="list-style-type: none"> 생태계 자생력 확보 일부 기술시장 우위 	<ul style="list-style-type: none"> 빅테크 기업 주도 혁신 도구화
변화 동인	기술환경	PC 대중화	인터넷 성장	모바일, 클라우드 확산	인공지능, 데이터 경제 확산
	시장환경	상용 소프트웨어 시장 성장	웹 서버 시장 성장	SW산업 패러다임 전환, 신규 개발 수요 폭증	디지털전환(x+AI) 수요 성장
전개과정		<ul style="list-style-type: none"> PC대중화-SW산업화, 해커문화유희-개발자 저항운동 리눅스공개→자유 SW운동 동력확보 	<ul style="list-style-type: none"> 라이선스정립→기업 상업적 활용 촉진 커뮤니티조직화-지속 가능성 강화 상용화 라이선스 등 오픈소스 외형 확립 	<ul style="list-style-type: none"> 신규개발수요, 빅테크 기업 오픈소스 참가 증대→오픈소스진영 신규개발자면입증가 오픈소스 개발자 생태계 자생력 확보 오픈소스 기술/시장 우위 증명 	<ul style="list-style-type: none"> 빅테크기업의 시드 기술제공, 커뮤니티 운영 등 적극 참여 오픈소스의 ICT산업 내 영향력증대 다양한 분야로 오픈소스 방식 확대
주요 변화 내용	핵심 참여자	개발자 커뮤니티	재단	재단 및 빅테크 기업	빅테크 기업
	대표 프로젝트	리눅스	LAMP*	안드로이드, 도커, 오픈스택, 하둡 등	텐서플로우, 파이토치, 쿠버네틱스, 아폴로 등
	프로젝트 성격 및 수준	유닉스계열 모방, 대체로 열위	일부대등, 틈새시장	일부 선도 & 시장표준	기술산업 혁신선도, 보편적 개발방식
	비즈니스 모델	<ul style="list-style-type: none"> 의도적 수익모델 없음 배포사업등 일부개시 	<ul style="list-style-type: none"> 지원서비스, SW판매 등 다양한 모델출현 	<ul style="list-style-type: none"> 기업 재단후원 대항화 플랫폼기반 수익모델 등장(모바일클라우드) 	<ul style="list-style-type: none"> 빅테크기업의 플랫폼 수익모델 가속화

* 출처: 저자 작성, LAMP: Linux, Apache, MySQL, PHP



- (기술·시장환경) 오픈소스SW는 빠르게 변화하는 기술·시장환경에 기민하게 적응하며 이를 성장의 발판으로 삼음
 - PC에서 인터넷, 모바일, 클라우드, 인공지능에 이르기까지 주요한 기술·시장 변곡점에서 오픈소스는 생태계 자생력을 확보하고 시장을 확장해나감
- (제품수준 및 성격) 오픈소스 초기에는 상용제품 기능을 재현하는 수준으로, 성능 측면에서는 상용대비 열위로 평가되었으나, 시기를 거듭할수록 성능과 시장 선호도에서 기존 상용제품 대비 우위를 점하며, 오늘날 ICT 혁신을 선도하는 주류로 자리매김
 - 이는 오픈소스 방식이 갖는 개방성, 투명성, 공동의 이익 추구를 위한 자발적 참여가 환경변화에 대한 적응력과 신속성을 담보했기 때문으로 보여짐
 - ※ 예를 들어, 리눅스는 1.0시기에 PC 대중화와 함께 IBM PC에서 작동하는 유닉스 호환 OS로 개발되었으나, 2.0시기 인터넷 확산과 함께 대표적 서버 OS로 변모하였고, 이를 기반으로 3.0시기 이후로는 클라우드, 모바일 OS의 기초로서 시장표준을 이끔
- (비즈니스모델) 오픈소스 초기, 활동의 목적이 이윤 추구가 아니었기 때문에 수익모델이 크게 발달하지 않았으나, 2.0시기 이후, 기업의 오픈소스SW 활용이 늘면서, 지원서비스 등 본격적인 수익화 모델이 등장하였고, 오늘날, 빅테크 기업들은 오픈소스를 통해 고객을 자사의 플랫폼으로 편입시키는 일종의 관문(gate)으로 활용 중임
- (주도 세력) 오픈소스 초기와 후기의 생태계 내 주도세력이 달라짐. 초기에 개발자와 자생적 커뮤니티가 주도했다면, 후기로 갈수록 조직화된 재단과 전략적 활용을 염두한 기업의 영향력이 증대되었음
 - 기업들은 자사 직원들의 오픈소스 프로젝트 참여를 독려하거나, 재단 지원, 오픈소스 프로젝트 개설 및 운영 등 다양한 방식으로 참여와 영향력을 확대
- (종합) 오픈소스는 소수 개발자의 소프트웨어 사유화에 대한 저항운동으로 시작되었으나, 라이선스와 재단 등을 통해 ‘조직화’되고 기업의 참여확대로 ‘수익성’이 높아지면서 일부 시장에서는 ‘지배적 기술’로 자리매김하였으며, 오늘날, 제4차 산업혁명 시대의 주요 기술의 ‘혁신의 장’으로 급성장하고 있음

□ 오픈소스, 협력과 경쟁의 진화

- (협력의 대상 - 개발자, 기업, 재단의 선택적 협력) 오픈소스 프로젝트의 성장경로를 분석해본 결과, 오픈소스 프로젝트의 목적성과 공개 주체의 목표에 따라 기업, 재단 등 다양한 주체와의 협력 관계를 통해 성장해온 것을 볼 수 있었음

- 개발자 및 학계 프로젝트는 프로젝트의 목적에 따라 재단 또는 기업 등의 큰 주체의 지원을 받아 성장한 것으로 분석됨
- 기업의 프로젝트는 기업의 전략적인 방향성에 따라 직접 관리, Alliance 구성, 재단 설립 등의 형태로 성장하는 모습을 보임
- (협력 기술/방식 - 저장소의 진화와 협업의 심화) 오픈소스 저장소는 온라인 기반 분산 버전관리시스템으로 진화하고 깃허브가 글로벌 오픈소스 생태계의 중심으로 성장하면서 오픈소스 협업 개발 방식을 발전시키고 이를 개발자 생태계의 보편적 방식으로 자리매김시키고 있음
 - 깃허브 등 오픈소스 저장소는 ① 개방성 ② 투명성 ③ 용이성 ④ 분산과 통제의 균형 ⑤ 네트워크 외부성 등의 특징으로 오픈소스 생태계 성장을 견인함
 - 오픈소스 저장소의 이러한 특성은 협업방식을 국경과 물리적 한계를 넘어 ‘광역화’시키고, 기술과 인력의 거래를 활성화하며 “협업을 심화”시키고 있음
- (경쟁 구도- 저항세력에서 시장의 주역으로) 오픈소스 주요 기술들의 협력 및 경쟁 구도에 대한 사례를 분석해본 결과, 시대별&기술별로 협력 대상과 경쟁구도가 변화함
 - (협력 주체의 변화) 오픈소스 1.0 시대에는 개발자들이 협력의 주체이자 중심이었으나, 오픈소스 4.0 시대에는 기업들이 협력의 중심에 자리매김하고, 오픈소스에 대한 직접지원(재단설립, 커뮤니티 운영 등)을 확대해감
 - (경쟁구도) 초기에는 오픈소스 진영과 거대 상용SW기업 간의 대결 구도였으나, 오픈소스 진영에 대기업들이 적극 참여하면서 오픈소스 기술이 시장표준으로 자리매김하거나 오픈소스 프로젝트 진영간 경쟁이 형성되는 경우가 많음

□ 오픈소스 생태계의 성장 요인들

- 오픈소스 진화단계, 프로젝트 성장경로, 협력 방식, 협력 및 경쟁 구도 등을 살펴본 결과, 다음의 요인들에 의해 오픈소스SW가 SW 생태계의 주류로 자리매김하였음을 알 수 있었음
 - ① (오픈소스 라이선스 정립) 보다 완화된 오픈소스 라이선스가 정립되면서 기업들의 상업적 활용이 촉진되고, 다양한 수익 모델이 등장함
 - ② (커뮤니티 성장 및 재단 설립) 커뮤니티가 성장하고 조직화되면서 재단이 설립되었으며, 재단이 주축이 되어 오픈소스 생태계를 법적·재정적으로 지원하고 프로젝트



의 성장을 도우면서 새로운 프로젝트 및 개발자의 유입이 증가

- ③ (지원기업의 성장) 레드햇, 블랙덕 등 전문기업들이 오픈소스SW를 배포, 설치, 교육, AS 등을 지원하면서 오픈소스SW 사용자층이 확대됨
- ④ (협업기반 개발 플랫폼 발전) SW 협력 개발을 위한 버전관리 시스템이 중앙집중식에서 온라인기반 분산관리 시스템(Git)으로 발전하면서 투명하고 효율적인 관리가 가능해지고, 개발자들의 참여가 용이해지면서 프로젝트가 확대됨
- ⑤ (오픈소스 기술 발전) 오픈소스SW의 품질이 올라가고 주류기술로 채택이 되면서 자생력을 확보하였으며 더 나아가 혁신의 도구로 자리 잡게 되면서 오픈소스 생태계가 더욱 확장됨
- ⑥ (기업 참여 확대) 기업들은 자사 직원들의 오픈소스 프로젝트 참여를 독려하거나¹⁰³⁾, 재단 지원, 오픈소스 프로젝트 개설 및 운영 등 다양한 방식으로 참여를 확대하면서 신규 개발자들의 편입이 증가함
- ⑦ (빅테크 기업의 참여) 구글, 마이크로소프트, 아마존 등 빅테크 기업들이 기술을 공개하고, 재단을 후원하는 등 오픈소스 생태계를 적극적으로 지원하면서 새로운 성장 원동력을 제공

□ 시사점

- 오늘날, 오픈소스는 일부 개발자들의 소프트웨어 개발방식이 아니라, 인공지능, 클라우드, 자율주행차 등 4차산업 혁명시대의 ICT 핵심 및 융합 기술 분야의 주된 기술 개발 방식이자 생태계 경쟁 방식임
- 제4차 산업혁명 시대, 인공지능 등 국가 지능화를 위한 기술력 확보를 위해서는 오픈소스 생태계를 적극 활용하고, 그 안에서 주도권을 장악할 수 있어야 함
- 우리 정부에서도 오랜 기간 동안 개발자 커뮤니티 지원 및 공개SW 과제 지원 등 오픈소스 생태계 활성화를 위한 지속적 노력을 해왔으나, 3.0시대와 4.0시대의 흐름과 특성을 고려한 좀 더 다각적이고 입체적인 전략 마련이 필요함
 - 우수한 오픈소스 개발 인력 양성과 함께 이들의 역량이 기업 및 기관 등 조직의 경쟁력 향상으로 이어지게 하기 위한 시스템 마련 필요

103) 오픈소스를 지원하는 기업들은 직원들이 근무시간 중에 공식적, 또는 비공식적으로 오픈소스 프로젝트에 기여 하도록 독려하여 ①프로그래머의 능력을 높이고, ②타 오픈소스 프로젝트와 해당 기업의 프로젝트를 통합할 수 있고, ③경쟁기업의 정보를 확보하고, ④고객들에게 홍보하는 등의 방식으로 오픈소스 프로젝트와 교류를 확대해 감 (러너·생커맨, 2017)

- 이에 본 연구의 분석에 따라 오픈소스 생태계 활성화 정책 마련을 위한 아래의 네 가지 전략 방향을 제안하고자 함
- ① 협력 전략 : 오픈소스 프로젝트를 성장시키기 위해서는 누구와 어떻게 협력하는가가 매우 중요하며 이를 위해서는 프로젝트, 또는 기업의 전략적 방향성에 대한 고려가 필요함
 - (개발자 및 학계) 오픈소스 프로젝트를 성장시키기 위해 재단 및 기업과 같은 큰 주체의 도움을 받는 것이 필요하며, 성장 방식을 선택할 때, 프로젝트의 성격 및 방향성을 고려할 필요가 있음
 - (기업) 어느 정도의 협력이 필요하고 어느 정도 기술 주도권의 확보가 필요한지를 고려하여 전략적으로 오픈소스 성장 방식을 선택한다면 기술의 성장 및 수익 확보에 도움이 될 것으로 보임
- ② 투자 주체 유인 : 오픈소스 생태계의 지속적인 성장을 위해서는 기술력 있는 개발자 및 기술기업의 지속적인 성장을 이루어줄 수 있는 투자 여력과 능력이 있는 주체들의 오픈소스 프로젝트에 대한 지원 및 상생을 위한 노력이 중요함
 - (기업) 시드 기술 공개, 오픈소스 개발 참여, 재단 및 프로젝트 후원 등 국내 ICT분야 대기업들의 전략적이고 적극적인 오픈소스 활동을 유인할 필요
 - (재단) 국내에서의 오픈소스 재단의 역할을 강화하여 개발자·중소기업·학계 등에서 직접 관리하기 힘든 오픈소스 프로젝트들을 관리하고 법적·재정적으로 지원하여 오픈소스 생태계의 기반을 다지는 것이 필요
 - (지원기업) 지원 서비스를 제공하는 오픈소스 전문기업들의 활성화는 오픈소스 개발 및 활용에 대한 진입장벽을 낮춰 오픈소스 생태계 활성화에 도움을 줄 수 있을 것임
- ③ 공정하고 안전한 협업을 위한 지원 정책 마련 : 국내 중소기업이나 스타트업, 개발자들이 성장하기 위해서는 국내 대기업 및 해외 빅테크 기업과 협업을 필요한데 공정한 협업을 위한 법제도 장치 마련을 통해 중소기업의 참여 활성화가 필요함
 - ※ 빅테크 기업은 씨드기술을 투입하고 문호를 개방하여 우호적 개발자, 사용자 생태계를 우선 조성하고, 해당분야의 주도권을 강화하는 한편, 수익모델 측면에서는 소프트웨어와 코드는 공개하되, 인프라와 플랫폼 제공으로 수익을 창출하는 플랫폼전략을 구사
 - 라이선스 정책에 대한 법률지식 부족, 언어장벽, 오픈소스 생태계 및 글로벌 비즈니스 관행에 대한 이해 부족 등으로 인한 피해 방지를 위한 교육 및 법률 지원 필요



- ④ 제4차 산업혁명 시대 기술력 확보를 위한 국가 R&D 전략으로서의 오픈소스 전략을 인식해야 함
 - 오늘날 오픈소스는 인공지능, 클라우드, 빅데이터, 자율차, 스마트공장, 스마트시티 등 다양한 혁신기술 개발의 협력과 경쟁의 장(場)으로 실력있는 개발자와 기업들이 모여 들고 있음
 - 오픈소스 생태계의 경쟁 및 협력 방식을 올바르게 이해하고, 우리가 가진 기술, 자산, 인력 중 어느 것을 공유하고 어느 포인트로 차별화하여 경쟁 우위를 차지할 것인지 등에 대한 치밀한 전략 필요
 - 특히, 정부 R&D는 (1) 글로벌 오픈소스 생태계에서 해외 빅테크 기업들과 협력 개발할 분야와 (2) 인프라 기술로 공개하면 유리한 분야, (3) 국내기업들을 위한 차별화된 기술을 개발하여 공급해야 할 분야 등을 전략적으로 선택하고, 그 기술의 성장경로에 대한 전략 마련 필요
 - 아울러 정부 오픈소스 R&D 혁신을 위한 기획, 수행, 평가 방안의 전향적인 개선과 함께 출연연의 오픈소스 협업 활성화를 위한 제도적·문화적 혁신이 필요함

□ 맺음말

- 오픈소스SW 개발은 단순한 소스코드의 공유가 아닌 지구 반대편의 개발자들이 함께 모여 하나의 기술을 개발하고 이를 공유하는 새로운 생산방식으로 자리매김 하고 있음
- 저항, 공유, 나눔, 기여 등 개인적 동기로 참여하는 개발자도 있지만, 오픈소스는 기술과 핵심인재를 확보하고, 생태계를 장악하려는 기업들의 전쟁터이기도 함
- 오픈소스 활성화가 다소 더디게 일어나고 있는 우리나라에서 이러한 오픈소스 생태계에 적응하고 주도권을 확보하기 위해서는 오픈소스 생태계 및 각 기술을 둘러싼 기술 및 산업 생태계의 특성에 대한 보다 다각적이고 지속적인 연구와 노력이 필요할 것임



참고문헌

◆ 국내자료

- Synopsys (2020), 2020 오픈소스 시큐리티&리스크 분석보고서(한글본).
- 금융보안원(2017), 머신러닝 활용을 위한 오픈플랫폼 현황.
- 김종배·김광용(2017), 오픈소스 소프트웨어 비즈니스 모델, 한티미디어.
- 남충현(2016), 오픈소스 AI : 인공지능 생태계와 오픈 이노베이션, 정보통신정책연구원.
- 이제응(2020), 「Cloud Native Computing Foundation」, 오픈 테크넷 서밋 2020 버추얼 컨퍼런스 발표자료, 2020.09.16.
- 이진휘(2019), 오픈소스 중요성과 시사점, 정보통신산업진흥원.
- 조만석(2020), 인공지능 오픈소스, ETRI 발표자료, 2020.09.16.
- 조쉬 러너, 마크 쉥커멘(2010), 오픈소스와 소프트웨어 산업, 상생의 경제학, 에이콘출판, p.99.
- 최성호 (2020), 4차 산업혁명의 숨은 원동력, 오픈소스 현황과 시사점, KDB 산업은행 산은 조사월보, 2020.1 제770호.
- 크리스토퍼 토찌(2017), 프리-오픈소스 소프트웨어 혁명의 역사, 지식함지.
- 클라우드스토어 씨앗(2018), 쿠버네티스가 클라우드 네이티브 컴퓨팅 재단을 졸업해 본격적인 오픈소스 프로젝트가 되다, 클라우드 이슈리포트, 2018.04.19.

◆ 국외자료

- Marketsandmarkets (2018), Open Source Services Market by Service Type (Support, Maintenance and Management, Implementation, Consulting), Industry (Manufacturing, Healthcare and Life Sciences, Retail and Distribution), and Region - Global Forecast to 2022,



[https://www.marketsandmarkets.com/Market-Reports/open-source-services-mar-
ket-27852275.html](https://www.marketsandmarkets.com/Market-Reports/open-source-services-market-27852275.html)

Statista (2020), Global market share held by operating systems for desktop PCs,
<https://www.statista.com/statistics/218089/global-market-share-of-windows-7/>

◆ 신문기사

[국내기사]

CIO코리아(2015.09.17.), 핵심 오픈소스 SW 재단 8곳, 그리고 그들이 중요한 이유,
<https://www.ciokorea.com/slideshow/26664>

디지털데일리(2012.04.13.), 레드햇의 마법...공짜 SW 팔아서 1조원 매출,
http://ddaily.co.kr/m/m_article/?no=89646

디지털타임스(2013.05.24.), 김택완 블랙텍SW코리아 대표, "오픈소스 활용은 세계적 추세",
http://www.dt.co.kr/contents.html?article_no=2013052402019960718003

머니S(2020.07.10.), [박기자의 스마트폰] 스마트폰 운영체제 10년 전쟁 '돌만 남았다',
<https://moneys.mt.co.kr/news/mwView.php?no=2020070111268036630>

블로터(2019.12.20.), 대형 클라우드는 오픈소스SW 회사들을 위협하는가,
<http://www.bloter.net/archives/365067>

블로터(2013.02.03.), MySQL 개발자 "오라클, 잘못 가고 있다",
<http://www.bloter.net/archives/142487>

지디넷(2020.08.26.), 리눅스 29돌...사건으로 보는 그 역사,
<https://zdnet.co.kr/view/?no=20200826135027>

[해외기사]

Ars Technica(2015.11.19.), Visual Studio now supports debugging Linux apps; Code editor now open source,

<https://arstechnica.com/information-technology/2015/11/visual-studio-now-supports-debugging-linux-apps-code-editor-now-open-source/>

Businesswire(2012.09.19.), OpenStack Launches as Independent Foundation, Begins Work Protecting, Empowering and Promoting OpenStack,
<https://www.businesswire.com/news/home/20120919005997/en/OpenStack-Launches-Independent-Foundation-Begins-Work-Protecting>

Forbes(2014.12.09.), Cloud Foundry Foundation Matures—Becomes A Linux Foundation Collaborative Project,
<https://www.forbes.com/sites/benkepes/2014/12/09/cloud-foundry-foundation-matures-becomes-a-linux-foundation-collaborative-project/?sh=700400fb452b>

Infoworld(2018.02.15.), Google Cloud Platform services guide: The right tools for the job,
<https://www.infoworld.com/article/3254749/google-cloud-platform-services-guide-the-right-tools-for-the-job.html>

Infoworld(2016.12.14.), MXNet review: Amazon's scalable deep learning,
<https://www.infoworld.com/article/3149598/mxnet-review-amazons-scalable-deep-learning.html>

InfoQ(2013.03.27.), “Docker: Automated and Consistent Software Deployments”,
<https://www.infoq.com/news/2013/03/Docker/>

Linux Journal(2006.05.31.), Linux Timeline, <https://www.linuxjournal.com/article/9065>

O'Reilly(2017.12.07.), “When two trends fuse: PyTorch and recommender systems”,
<https://www.oreilly.com/content/when-two-trends-fuse-pytorch-and-recommender-systems>

Synced(2018.04.03.), Caffe2 Merges With PyTorch,
<https://medium.com/@Synced/caffe2-merges-with-pytorch-a89c70ad9eb7>

Wired(2015.11.09.), Google Just Open Sourced TensorFlow, Its Artificial Intelligence Engine,
<https://www.wired.com/2015/11/google-open-sources-its-artificial-intelligence-engine/>

ZDNet(2014.08.22.), Most popular open-source cloud projects of 2014,
<https://www.zdnet.com/article/most-popular-open-source-cloud-projects-of-2014>



◆ 웹사이트

Amazon AWS 홈페이지, https://aws.amazon.com/ko/?nc2=h_lg

Amazon AWS, “AWS 소개”, <https://aws.amazon.com/ko/about-aws/>

Amazon AWS, “Docker란 무엇입니까”, <https://aws.amazon.com/ko/docker/>,

Amazon AWS Blog, “Excited about MXNet joining Apache!”,

<https://aws.amazon.com/ko/blogs/aws/excited-about-mxnet-joining-apache/>

Amazon Open Source,

<https://aws.amazon.com/ko/opensource/?blog-posts-content-open-source.sort-by=item.additionalFields.createdDate&blog-posts-content-open-source.sort-order=desc>

Apache Software Foundation, “The Apache Incubator”, <http://incubator.apache.org/>

Apollo 홈페이지, <https://apollo.auto/>

Apple Swift, <https://developer.apple.com/kr/swift/>

Bitbucket 홈페이지, <https://bitbucket.org/>

Cloud Native Computing Foundation 홈페이지, <https://www.cncf.io/>

Crunchbase, “Open Source Startups”, <https://www.crunchbase.com/hub/open-source-startups>

Debian, “간단한 데비안 역사”, <https://www.debian.org/doc/manuals/project-history/>

Docker 홈페이지, <https://www.docker.com/>

ETRI 오픈소스 포털, “라이선스 소개”, https://oss.etri.re.kr/kor/sub02_01_01.do

GitHub 홈페이지, <https://github.com/>

GitHub Blog, <https://github.blog>

GitHub Blog, “10 Million Repositories”, 2013.12.23.,

<https://github.blog/2013-12-23-10-million-repositories/>,

GitHub Blog, “Thank you for 100 million repositories”, 2018.11.08.,

<https://github.blog/2018-11-08-100m-repos/>.

GitHub The State of Octoverse 2019, <https://octoverse.github.com/2019>
 GitHub The State of Octoverse 2017, <https://octoverse.github.com/2017>
 GitLab 홈페이지, <https://about.gitlab.com/>
 Google Cloud 홈페이지, <https://cloud.google.com/>
 Google Open Source, <https://opensource.google/>
 Google Open Source Blog, “Open source by the numbers at Google”,
<https://opensource.googleblog.com/2020/08/open-source-by-numbers-at-google.html>
 HAMONI KR, “오픈소스란 무엇인가요?”, <https://hamonikr.org/oss/51202>
 Hoffa, F., “Who contributed the most to open source in 2017 and 2018? Let’s analyze
 GitHub’s data and find out.”, 2017.10.24.,
<https://www.freecodecamp.org/news/the-top-contributors-to-github-2017-be98ab854e87/>,
 IBM, “IBM Invests \$1B to Deliver Unique Platform-as-a-Service Capabilities to
 Connect Enterprise Data and Applications to the Cloud”,
<https://www-03.ibm.com/press/us/en/pressrelease/43257.wss>
 Intellipaat Blog, “Hadoop Creator goes to Cloudera”, 2015.11.26.,
<https://intellipaat.com/blog/hadoop-creator-goes-to-cloudera/>,
 Javatpoint 홈페이지, “Git Tutorial”, <https://www.javatpoint.com/git>
 Kubernetes 홈페이지, <https://kubernetes.io/ko/>
 Linux Foundation 홈페이지, <https://www.linuxfoundation.org/>
 Linux Foundation, “Linux Foudation Landscape”,
<https://lf-landscape.netlify.app/category=lf-members&format=members&grouping=category>
 Microsoft Azure 홈페이지, <https://azure.microsoft.com/ko-kr/>
 Microsoft Blog, “Windows Azure General Availability”, 2010.02.01.,
<https://blogs.microsoft.com/blog/2010/02/01/windows-azure-general-availability/>,



Microsoft GitHub, <https://github.com/microsoft/>

Microsoft Open Source, <https://opensource.microsoft.com/>

NIPA 오픈소스 소프트웨어 통합지원센터 홈페이지, <https://www.oss.kr>

Open Container Initiative 홈페이지, <https://opencontainers.org/>

Open Handset Alliance, “Google and the Open Handset Alliance Announce Android Open Source Availability”, 2008.10.21., http://www.openhandsetalliance.com/press_102108.html,

Open Handset Alliance, “Industry Leaders Announce Open Platform for Mobile Devices”, 2007.11.05., http://www.openhandsetalliance.com/press_110507.html,

Open Source Initiative, “The Open Source Definition”, <https://opensource.org/osd>

Papers with code, <https://www.paperswithcode.com/>

Php 공식 홈페이지, “History of PHP”, <https://www.php.net/manual/en/history.php.php>

Propublica Nonprofit Explorer, “The Linux Foundation”, <https://projects.propublica.org/nonprofits/organizations/460503801>

PyTorch GitHub, <https://github.com/pytorch/pytorch/releases/tag/v0.1.1>

Red Hat 홈페이지, <https://www.redhat.com/ko>

Red Hat, “Red Hat: Leading the enterprise Linux server market”, 2019.12.05., <https://www.redhat.com/ko/blog/red-hat-leading-enterprise-linux-server-market>,

Red Hat, “레드햇에 대해 알아야 할 25가지 사실”, <https://www.redhat.com/en/events/레드햇에-대해-알아야-할-25가지-사실>

Samsung SDS, “깃허브(GitHub)의 변화와 플랫폼 전략”, 2020.08.07., <https://www.samsungsds.com/kr/insights/github.html>

SourceForge 홈페이지, <https://sourceforge.net/>

TTA 신규용어 소개, “깃허브”, http://www.tta.or.kr/data/weeklyNoticeView.jsp?pk_num=5775

Ubuntu 홈페이지, <https://ubuntu.com/>

WhiteSource, “Open Source License Comparison”,

<https://resources.whitesourcesoftware.com/blog-whitesource/open-source-licenses-comparison-guide>

Wikipedia, “Apache Spark”, https://en.wikipedia.org/wiki/Apache_Spark

Wikipedia, “macOS”, <https://en.wikipedia.org/wiki/MacOS>

Wikipedia, “iOS”, <https://ko.wikipedia.org/wiki/IOS>

Wikipedia, “iOS SDK”, https://ko.wikipedia.org/wiki/IOS_SDK

Wikipedia, “깃랩”, <https://ko.wikipedia.org/wiki/깃랩>

Wikipedia, “도커”, [https://ko.wikipedia.org/wiki/도커_\(소프트웨어\)](https://ko.wikipedia.org/wiki/도커_(소프트웨어))

Wikipedia, “리눅스”, <https://ko.wikipedia.org/wiki/리눅스>

Wikipedia, “리눅스 재단”, https://ko.wikipedia.org/wiki/리눅스_재단

Wikipedia, “빗버킷”, <https://ko.wikipedia.org/wiki/빗버킷>

Wikipedia, “소스포지”, <https://ko.wikipedia.org/wiki/소스포지>

Wikipedia, “안드로이드(운영 체제)”, [https://ko.wikipedia.org/wiki/안드로이드_\(운영_체제\)](https://ko.wikipedia.org/wiki/안드로이드_(운영_체제))

w3resource, “MySQL Tutorial”, <https://www.w3resource.com/mysql/mysql-tutorials.php>

네이버캐스트 용어로 보는 IT, “이클립스”,

<https://terms.naver.com/entry.nhn?docId=3580755&cid=59088&categoryId=59096>



저자소개

- 김성민** ETRI 지능화융합연구소 기술정책연구본부 지능화정책연구실 책임연구원
e-mail: songmin516@etri.re.kr Tel. 042-860-6172
- 홍아름** ETRI 지능화융합연구소 기술정책연구본부 지능화정책연구실 연구원
e-mail: areumh@etri.re.kr Tel. 042-860-0742
- 최새솔** ETRI 지능화융합연구소 기술정책연구본부 지능화정책연구실 선임연구원
e-mail: saesol.choi@etri.re.kr Tel. 042-860-1803
- 연승준** ETRI 지능화융합연구소 기술정책연구본부 지능화정책연구실장
e-mail: sjyeon@etri.re.kr Tel. 042-860-6437

오픈소스 4.0 - 협력과 경쟁을 위한 혁신의 도구 -

- 발행인** 이 지 형
- 발행처** 한국전자통신연구원 지능화융합연구소 기술정책연구본부
- 발행일** 2020년 12월 31일



www.etri.re.kr

본 저작물은 공공누리 제4유형:

출처표시+상업적이용금지+변경금지 조건에 따라 이용할 수 있습니다.



ETRI Electronics and Telecommunications
Research Institute

34129 대전광역시 유성구 가정로 218
TEL.(042) 860-6114 FAX.(042) 860-6504

비매품/무료



9 788955 192964
ISBN 978-89-5519-296-4