

A-05-HellTheGit

2013045957 이진솔

2018044811 우경완

2018044375 김상준

2018044566 박서연

2018044439 김영웅

목차

1 오픈소스 정의 및 의의

- 가) 오픈소스 정의
- 나) 오픈소스 장단점
- 다) 오픈소스의 의의
- 라) 오픈소스 유사개념

2 오픈소스 역사

- 가) 과거
- 나) 현재
- 다) 미래

3 오픈소스의 기본 이념 및 조건

- 가) 오픈소스의 기본 이념
- 나) 오픈소스의 조건

4 오픈소스 관련 단체 및 규정 사례

- 가) 오픈소스 관련 단체 및 기업
- 나) 리눅스의 개발자 행동강령

5 오픈소스의 법적 측면 및 해외정책 동향

- 가) 오픈소스의 법적 측면
- 나) 오픈소스 소프트웨어 라이선스의 독점 규제 및 공정거래에 대한 법률상 쟁점
- 다) 오픈소스 소프트웨어 수용정책의 확산과 해외정책 동향

1. 오픈소스의 정의 및 의의

1. 오픈소스 정의

오픈소스 소프트웨어(Open Source Software)는 소프트웨어의 소스코드가 공개되어, 누구나 소스코드를 자유롭게 사용하고 수정하여 재배포할 수 있는 소프트웨어이다. OSS, FOSS, 오픈소스, 공개 소프트웨어라는 용어가 사용되나 통상 '오픈소스' 라고 말한다.

2. 오픈소스 장단점

가) 오픈소스 장점

- 융통성

라이선스 비용이나 예산에 제한을 받지 않고, 다양한 오픈 소스들을 테스트한 뒤 최선의 것을 선택할 수 있다.

- 기술지원

신속한 문제 해결, 빨라진 성능 개선 프로세스, 기술의 공동 습득이 가능하다.

- 기술 혁신

유료일 경우 사용하지 않았을 기술들의 실험 적용이 가능하다.

- 재활용

소스 코드 접근이 가능하기 때문에 재활용이 증가한다.

- 품질

이미 검증된 소스를 사용함으로써 개발이 빨라지고 유연해진다. 또한 한정된 자원으로 인해 품질 비용에 대해 제한의 장벽을

깨트릴 수 있다.

- 표준

표준에 충실하고 상호 유연성(원하는 대로 변경이 가능함)이 뛰어나다.

- 비용 절감

이용에 비용이 들지 않거나 적다. 기업들은 필요한 인력과 시간을 절약하여 시장 적시성과 비용절감을 위해 검증된 오픈소스 소프트웨어를 활용할 수 있다.

- 다양성

대부분의 기업들은 시장의 규모와 수요를 파악한 후 시장성과 수익성을 고려해 소프트웨어 개발을 진행하게 된다. 하지만, 오픈소스 소프트웨어의 경우는 개인의 필요나 지적유희의 목적으로 개발되는 경우가 상당부분이기에 전 세계에 흩어져 있는 여러 개발자들에 의해 개발되는 오픈소스 소프트웨어의 다양성은 상용 소프트웨어를 능가한다.

나) 오픈소스 단점

- 기능의 불확실성

무엇이 OSS를 구성하고 있는지, 상대적인 장단점이 무엇인지 불명확하다.

- 기술지원의 불확실성

제품의 지원이 분산되거나 지원을 못 받는 것에 대한 불안감이 존재한다.

- 탐색의 어려움

특정한 비즈니스 문제들에 대한 적절한 OSS 응용제품을 확인하는

것에 대한 어려움이 존재한다.

- 문서화 미비

문서화가 특정 개인의 취향에 따라 편협하게 이루어졌거나 아예 없는 경우도 존재한다.

- 하드웨어 지원

독점 소프트웨어보다 새로운 하드웨어를 지원하는 면에서 낮다.
(호환성 문제)

3. 오픈소스 의의

2008년, Standish Group에 의한 리포트에 따르면 오픈 소스 소프트웨어 모델의 채택은 소비자에게 연간 600억 달러의 절약을 가능하게 한다.

4. 오픈소스 유사개념

가) 프리웨어 (Freeware)

무료로 라이선스 비용 없이 배포되는 소프트웨어를 말하는데, 사용자가 사용하는 데에는 따로 제한이 없지만, 영리를 목적으로 배포할 수 없다는 특징을 가지고 있다.

나) 셰어웨어 (Shareware)

프리웨어와 비용을 지불하지 않는다는 점에서 유사하다. 다만, 상용 소프트웨어를 구매하기 전 제한된 기간이나 제한된 기능으로 사용성 평가를 위해 사용할 수 있는 소프트웨어를 뜻한다.

다) 애드웨어 (Adware)

해당 프로그램을 무료로 사용 기간의 제한 없이 사용 가능하지만 해당 소프트웨어를 사용하는 동안에는 지속적으로 광고에 노출되는 소프트웨어를 말한다.

라) 퍼블릭 도메인 소프트웨어 (Public Domain Software)

퍼블릭 도메인 소프트웨어라는 개념은 미국의 대학에서 유래하였다. 컴퓨터 프로그램이 공적자금의 지원을 받아 개발된 경우, 공유를 할 것이라는 조건을 달아 일반 공중이 장래 개발되는 컴퓨터프로그램을 자유롭게 사용할 수 있게 하였는데, 이 개념은 이로부터 유래하였다. 현재는 개인 프로그래머들도 컴퓨터 프로그램을 자신의 이윤추구목적이 아닌 이유로 배포하는 경우 이 개념을 사용하고 있다.

마) 셰어드 소스 소프트웨어 (Shared Source Software)

셰어드 소스 소프트웨어란 Microsoft가 제한된 조건으로 소스코드를 공개하면서 생겨났다. 이러한 제한된 조건이란, 이용자는 소스코드에 접근 및 수정하고 이를 테스트 목적으로 제공할 수 있는 반면에 이를 상업적으로 재배포 할 수는 없다는 것이다. PC 운영체제의 80%를 차지하는 Microsoft는 리눅스 등 오픈소스 소프트웨어의 확산에 대응하기 위하여 보안 강화, 그리고 응용프로그램 개발을 위해 학계 연구소, 정부 기관, 시스템 통합 개발자 등에게 소스코드의 일부를 공개하고 있다.

2. 오픈소스의 역사

1 과거

가) 1950 - 60년대

이 시기에 만들어진 소프트웨어는 연구나 학술적인 목적으로 만들었기 때문에 거의 모두 퍼블릭 도메인(Public Domain, 저작권 보호대상이 아닌 프로그램)이었다. 당시에는 하드웨어 간의 호환성이 없었기 때문에 사용자 간에 소프트웨어를 자유롭게 배포하여 마음대로 수정하는 관습은 흔히 있는 일이었다.

나) 1970 - 80 년대

소프트웨어의 개념이 상용화 측면을 옮겨 갔다. 그러나, 학자들은 여전히 소프트웨어를 공동으로 개발했다. 예를 들어, Donal Knuth는 1979년 TeX typesetting system 또는 Richard Stallman은 1983년 GNU 운영체제를 사용했다. 벨 연구소에 의해 Unix 개발, AT&T를 비롯한 여러 회사, 버클리 등 비영리 단체에서 다양한 버전을 개발하였고, 1980년대 HP/UX, AIX, Solaris 등의 상용버전 Unix가 개발되었고 상용화됨. UNIX 운영체제 소스 코드의 공유와 이에 기반을 둔 다양한 수정본의 개발은 오픈소스 운동에 지대한 영향을 미쳤다.

다) 1976년

빌 게이츠가 컴퓨터 애호가들에게 공개 편지 "Open Letter to Hobbyists"를 보냄. 이를 계기로 소프트웨어 개발자들이 소프트웨어 저작권에 대해 인지를 하였고, 상업용 소프트웨어 시대가 시작되었다. 따라서 모든 코드를 비공개 형태로 만들어 버리는 계기가 된 상용 소프트웨어들이 등장했다.

라) 1980년대

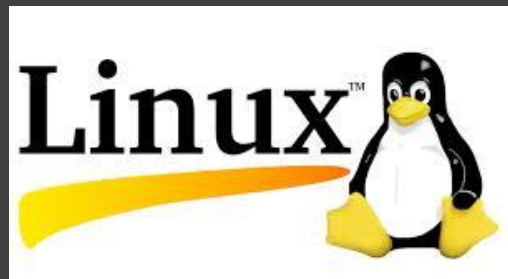
AT&T가 유닉스와 관련된 지적재산권을 독점하면서 본격적인 소스 코드 독점에 대한 문제가 부각되기 시작했다. 이러한 시기에 맞춰 리처드 스톨만은 자유 소프트웨어 운동을 주도했



다. 상용 소프트웨어에 대항한 자유로운 대안을 만들기 위한 의지를 표출하며, GNU is Not a Unix(GNU) 프로젝트를 시작했다. 이 운동 후 자유 소프트웨어 재단을 설립했다. (자유라는 의미는 비용을 지불하지 않는다는 의미보다는 소스코드에 대한 변경 등의 접근권한이 최초 제작자와 동일하게 누구에게나 주어진다는 의미가 더 강함) 이 재단을 만든 이유는 지적재산권이 기업의 독점적 지위를 높이고 시장을 독차지하며 기술혁신을 가로막을 것이라는 우려로 설립되었다.

마) 1989 – 1991년

오픈소스 라이선스의 대표격인 General Public License가 발표되었다. 이것은 오늘날 지적재산권을 상징하는 의미의 Copy Right에 반대적인 입장인 Copy Left 운동의 핵심이다. 91



년 Linux가 탄생하였다. 리눅스 토발즈가 취미로 개발한 커널, MINIX를 응용하여 리눅스 커널 개발 수 GNU GPL로 배포하였다.

바) 1997년

Eric Raymond는 자유 소프트웨어 철학을 대변하는 The Cathedral and the Bazaar을 발행했다. 이 소논문은 개방형 개발모델의 강점을 주장했다. 성당모델은 폐쇄형 소프트웨어 개발 모델, 시장모델은 개방형 소프트웨어

개발 모델에 비유했다. 다수의 사람에 의해 버그를 잡아낼 가능성이 높기 때문에 시장 모델이 우수한 개발 모델이라고 주장했다. 후에 1998년 초, 엄청난 주목을 받았고 Netscape Communication이 인기있는 Netscape Communicator 인터넷 제품군을 Free Software로 출시하도록 동기를 부여하는 한가지 요소가 되었다.

사) 1998년



Eric Raymond는 자유라는 표현이 갖는 무비용의 인식을 바꾸기 위해 오픈소스라는 표현을 사용했다. 넷스케이프 브라우저 코드에 대한 공개형태를 결정하는 전략회의에서 만들었다. 2월 3일 미국 캘리포니아 마운틴 뷰의 VA 리눅스 시스템즈 사물실에서 열린 전략회의에서 오픈소스라는 용어가 처음 사용되었다.

넷스케이프의 소스코드 공개에 자극을 받아, 새로운 용어의 사용을 장려하고 오픈소스 원칙을 전도하기 위해 OSI가 설립되었다. Eric Raymond가 초대사장을 맡았다. OSI로고는 콜린 비브룩이 만들었는데, O는 개방성을, 열쇠구멍은 소스코드의 잠금 해제를 암시한다.(뒤에 이미지가 첨부되어있다.) OSI는 폐쇄형 소프트웨어개발의 이력과 인터넷 개발자 커뮤니티가 이미 제공한 오픈 개발 사례에서 최소 20년의 증거를 확보하여 넷스케이프 같은 상업적 사업에 오픈소스 사례를 제시하였다.

아) 1999년 – 2009년

Eric Raymond가 전체 소프트웨어의 95%이상이 상업적 목적이 아닌 실용적 목적으로 개발되고 있으므로 오픈소스가 되어야 한다고 주장했다. 또한 브루스 페렌스가 데비안 프리 소프트웨어 지침을 소프트웨어 라이선스가 어떻게 오픈소스로 인식될 수 있는지에 대해 객관적인 정의를 시작했는데, 그것이 바로 OSD(Open Source Definition)

2000년대 중반에는 오픈소스에 대한 인기가 높아지면서 라이선스의 무분별한 확산이 문제가 되었다. 2004-2006년 OSI는 이 문제를 처리하기 위해 공공의견 수렴과정을 진행하였다.

2009년 EU 집행위원회 Flossmetrics의 연구 결과에 따르면 전체코드의 35%가 오픈소스였다.

2 현재



현재 오픈소스 시장의 대부분은 2005년에 리눅스 토발즈가 만든 Git이 지배하고 있다. Git에는 8500만개 가량의 프로젝트와 2800만 명 이상의 개발자를 가진 거대한 오픈소스 코드 시장이다. Github가 오픈소스 시장을 지배하기 전 오픈소스 시장을 장악했던 소스포지가 있었다. 그러나 소스포지는 서비스 형태를 재구축하면서 개발들에게 금전적 부담을 지웠고, 이로 인해 개발자들이 Github로 대거 이주하게 되면서 소스포지는 하락세를, Github는 상승세를 타게 되었다. 이뿐만 아니라 MS에서 운영하던 코드 플렉스 역시 Github에 밀려 폐쇄하게 되었다.



2017년 Github는 자사의 오픈소스 커뮤니티인 코드 플렉스를 폐쇄한 MS에게 700억 달러에 인수되었다. 그러나 개발자들은 Github가 거대 IT기업인 MS에 종속되었다는 그 자체를 불안해하며 Gitlab등으로 이전하기 시작했다.

현재까지 오픈소스 시장은 Github가 장악하고 있지만, 과거 오픈소스 시장을 독점하다 몰락한 소스포지처럼 언제든지 Github도 무너질 가능성이 있다. MS가 깃허브를 인수한 700억 달러를 어떤 방식으로 회수하느냐에 따라 오픈소스 시장의 흐름이 바뀌게 될 것이다.

3 미래

비록 지난 20년 간 오픈소스 코드를 '사용'하는 기업들은 늘었지만, 결국 오픈소스의 성공 여부를 판가름하는 것은 과연 그것이 소프트웨어에서의 혁신 방식을 바꿔 놓을 수 있는지, 어떻게 바꿔 놓을 것인지에 달려 있다. 물론 모든 소프트웨어에 오픈소스가 적합한 것은 아니다. 아파치 소프트웨어 재단 디렉터이자 어도비 수석 과학자 베르트랑 델라크레테즈의 말처럼, "오픈소스는 인프라 소프트웨어에 최적"이다. 반대로 애플리케이션 소프트웨어에는 오픈소스가 크게 확산되기 어려운데, 왜냐하면 '합의가 어려워지기 때문'이다. 또한, 특정 소프트웨어에 관심과 적합한 역량을 지닌 개발자 수 역시 스택 레이어가 올라 갈수록 줄어든 것이다.

그러나 기반형 소프트웨어(foundational software)의 경우 오픈소스가 혁신을 주도해 나가는 경향이 있다. 기업이 오픈소스에 맞춰 "소프트웨어의 패러다임을 바꿀 정도"라고 워커는 말한다. 그러나 앞으로의 20년 동안은 점차 이러한 경향이 사그라들고 오픈소스 커뮤니티 활동 및 참여가 더욱 활발해질 것으로 예상된다.

3. 오픈소스 기본 이념 및 조건

가) 오픈소스의 기본 이념

소프트웨어의 소스 코드를 자유롭게 읽고, 재배포 및 개조를 가능하게 함으로써 소프트웨어가 향상되고, 한사람이 느린속도로 소프트웨어를 개발하는 것보다 여러 사람들이 고치고 쓰고 버그를 개선하는 것이 보다 빠를 수 있다.

- ① 소프트웨어의 소스 코드를 자유롭게 읽고, 재배포 및 개조를 가능하게 함으로써 소프트웨어가 향상되고, 한 사람이 느린 속도로 소프트웨어를 개발하는 것보다 여러 사람들이 고치고 쓰고 버그를 개선하는 것이 보다 빠를 수 있다.
- ② 이미 있는 오픈소스를 fork하여 새 프로젝트가 만들어지는 경우, 라이선스 때문에 공개해야 한다.
- ③ 소프트웨어를 더 좋게 만들기 위해, 외부 개발자들이 참여하도록 하면 더 다양한 환경에서 소프트웨어를 시험해보고 품질을 높일 수 있다
- ④ 프로그램의 신뢰성을 보장하는 방법이 될 수 있다.

나) 오픈소스의 조건

오픈소스 소프트웨어와 다른 소프트웨어의 차이는 라이선스의 차이에서 비롯된다. 오픈소스 소프트웨어가 되기 위해서는 소프트웨어의 배포 조건이 OSI에서 정한 다음의 8가지 기준을 만족하여야 한다.

① 자유로운 배포와 재배포(Free Redistribution)

특정한 소프트웨어의 라이선스에는 해당 소프트웨어의 일부나 전부가 다수의 프로그램으로 구성되는 배포판의 일부로 포함되어 재배포되지 못하도록 배포나 판매상의 제한을 설정할 수 없다. 또한 이러한

종류의 배포판에 대한 판매나 양도에 있어서 별도의 라이선스 비용을 징수할 수 없다.

② 소스코드의 공개

프로그램 저작물에는 반드시 소스코드가 포함되어야 하며, 컴파일된 형태 뿐만 아니라 소스코드의 배포 또한 허용되어야 한다. 만약, 소스코드를 제외한 상태로 배포하고자 한다면 일반적으로 통용되는 매체를 이용해서 제작 실비에 준하는 비용으로 소스코드를 제공해야만 한다. 소스코드는 프로그래머들이 개작하기에 용이한 형태로 제공되어야 하며, 고의로 복잡하고 혼란스럽게 만들어진 형태와 실행처리거나 번역기에 의해서 생성된 중간 형태의 코드는 허용되지 않는다.

③ 2차적 저작물

라이선스에는 프로그램 원저작물의 개작이나 이를 이용한 2차적 프로그램의 창작이 허용되어야 하며, 이러한 파생적 프로그램들은 최초의 프로그램이 갖고 있던 라이선스의 규정과 동일한 조건하에서 재배포될 수 있어야한다.

④ 소수 코드 수정의 제한(Integrity of Author's Source Code)

빌드 과정을 통해서 프로그램을 개작할 목적으로 소스코드와 패치 파일을 함께 배포할 경우에는, 정상적인 빌드를 보장하기 위해서 라이선스 안에 소스 코드의 수정을 제한하는 항목을 추가할 수 있다. 그러나 이러한 경우에도 수정된 소스 코드를 이용해서 만들어진 소프트웨어에 대한 자유로운 배포를 허용해야 하며, 수정된 소스코드를 통해서 만들어진 2차적 프로그램을 원래의 프로그램과 구별하기 위해서 별도의 이름과 버전을 사용할 것을 요구하는 항목을 추가할 수 있다.

⑤ 사람이나 그룹에 대한 차별 금지

라이선스라 모든 개인이나 단체에 대하여 동일하게 적용되어야 한다.

⑥ 라이선스의 배포(Distribution of License)

프로그램에 대한 권리는 반복되는 배포에 따른 별도의 라이선스 승인이나 양도 과정 없이도 프로그램을 배포받은 모든 사람에게 동일하게 적용된다.

⑦ 라이선스 적용 상의 동일성 유지

프로그램에 대한 권리는 반복되는 배포 과정에서 특정한 배포판에 포함되어 있는 상태로만 유효하지 않고, 모든 배포 단계에서 동일한 효력을 갖는다. 만약, 특정한 배포판에 포함되어 있던 프로그램을 독립적으로 사용하거나 재배포한다면 해당 프로그램을 배포받은 사람은 프로그램이 포함되어 있던 최초의 배포판 상태에서 발생한 권리와 동일한 권리를 갖는다.

⑧ 다른 라이선스의 포괄적 수용

라이선스에 오픈소스 소프트웨어와 함께 배포되는 소프트웨어에 대한 제한을 설정해서는 안된다.

4. 오픈소스 관련 단체 및 규정 사례

1. 오픈소스 관련 단체 및 기업

가) 아파치 소프트웨어 재단(Apache Software Foundation, ASF)



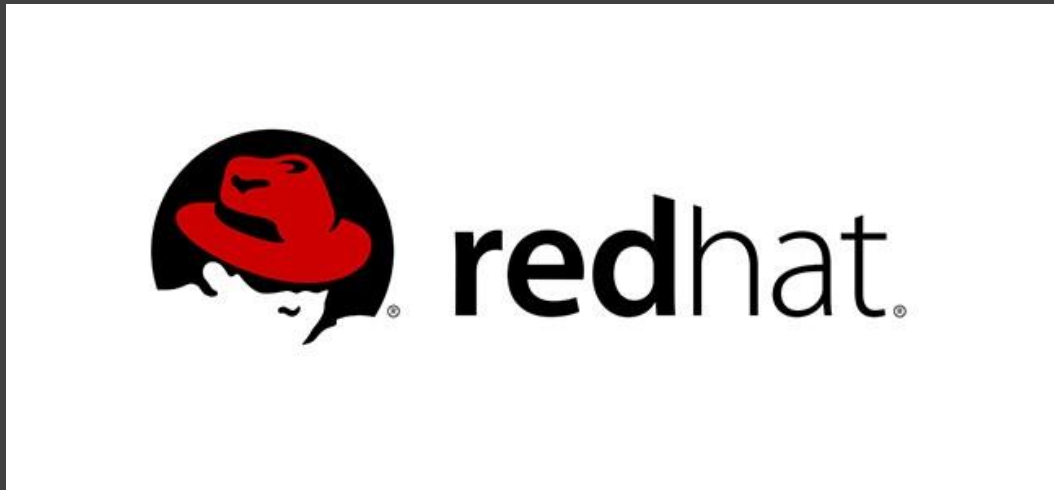
아파치 HTTP 서버를 포함한 아파치 소프트웨어 프로젝트를 지원하는 비영리 재단이다. 1999년 6월, 아파치 소프트웨어 재단은 아파치 그룹으로 구성되어 미국 델라웨어에서 설립되었다. 그들이 개발하고 있는 소프트웨어는 아파치 라이선스 조항 아래 배포되고 자유 소프트웨어/오픈소스 소프트웨어이어야 한다. 아파치 프로젝트는 공동 제작과 합의에 기반한 개발 프로세스와 오픈되고 실용적인 소프트웨어 라이선스라는 특징으로 규정된다. 각각의 프로젝트는 프로젝트에 공헌자로 활동하는 기술 전문가들로 구성된, 자발적으로 참여하는 팀에 의해 관리된다.

나) 모질라(Mozilla)



1998년 넷스케이프 출신 개발자들이 창립한 자유 소프트웨어 커뮤니티이다. 파이어폭스 웹 브라우저, 썬더버드 이메일, 버그질라 오류 추적기등을 개발했다. 2003년 모질라 커뮤니티가 중심이 되어 비영리 재단인 모질라 재단을 설립했다.

다) 레드햇(RedHat)



1993년 창립된 리눅스 배포판 제작회사로, 레드햇 리눅스를 제작했다. 레드햇은 오픈소스를 비즈니스로 전개하면서 소프트웨어를 무료로 공급하고 지원을 유료로하는 비즈니스 모델을 기본으로 하였지만, 최근에는 고도화된 소프트웨어를 유료로 판매하는 비즈니스도 시작하고 있다.

라) OSI(Open Source Initiative)



오픈소스 소프트웨어 사용을 장려하기 위하여 만들어진 단체, 1998년 2월에 브루스 페렌스와 에릭 레이먼드가 넷스케이프 커뮤니케이터에 대한 소스코드를 공개한 것에 대해 고무되어 설립하였다. 그뒤로 1998년 8월에 이 단체는 일련의 관리자들을 추가하였다. 레이먼드는 2005년 2월까지 설립부터 단체 장을 맡아왔으나 현재의 장은 마이클 티에먼이 맡고있다.

2. 리눅스의 개발자 행동강령

```
author      Greg Kroah-Hartman <gregkh@linuxfoundation.org> 2018-09-15 20:26:44 +0200
committer   Linus Torvalds <torvalds@linux-foundation.org> 2018-09-16 11:42:28 -0700
commit      8a104f8b5867c682d994ffa7a74093c54469c11f (patch)
tree        a8c6159ab818cc601c662d0e5cadfb67ea240fca
parent      27c5a778dfe23d1de8d9ebfbc6a54595a79ac709 (diff)
download    linux-8a104f8b5867c682d994ffa7a74093c54469c11f.tar.gz
```

Code of Conduct: Let's revamp it.

The Code of Conflict is not achieving its implicit goal of fostering civility and the spirit of 'be excellent to each other'. Explicit guidelines have demonstrated success in other projects and other areas of the kernel.

Here is a Code of Conduct statement for the wider kernel. It is based on the Contributor Covenant as described at www.contributor-covenant.org

From this point forward, we should abide by these rules in order to help make the kernel community a welcoming environment to participate in.

리눅스 커널 개발자들은 행동강령을 가지고 있다. Linux Code of Conduct 중에는 '개방하고 환영하는 환경을 조성하기 위해, 우리는 기여자와 관리자로서 나이, 신체크기, 장애, 민족성, 성적 표현, 경제적 지위, 국적, 개인적 외모, 인종, 종교 또는 성적 정체성과 지향성에 관계없이 우리 프로젝트와 우리 커뮤니티에 참여할 것을 약속합니다.' 라는 문구가 있다.

기여자 행동 강령 규약에 '우리의 규격' 부분에서는 긍정적인 환경을 조성하는데 기여하는 행동의 예를 나타내고 있다. 예를 들어 우호적이고 포괄적인 언어 사용, 서로 다른 견해와 경험에 정중하기, 건설적인 비판을 예의바르게 받아들이기, 명백한 허락없이 다른 사람의 개인정보 게시 등이 있고 '우리의 책임' 부분에서는 유지 관리자와 관리자의 역할들이 쓰여 있다.

5. 오픈소스 법적 측면 및 해외 정책 동향

1 오픈소스의 법적 측면

일반적으로, 저작권은 타인이 인계, 훼손 또는 소송의 위험이 없이 작업을 사용, 복사, 배포 또는 수정할 수 없음을 의미한다. 그러나 오픈소스는 다른 사람들이 작업을 사용, 수정 및 공유하기를 기대하기 때문에 예외적이다. 그러나 명시적으로 사용권한을 명시한 사용권이 필요하다.

오픈소스 라이선스를 신청하지 않으면, 프로젝트에 기여한 모든 사람이 자신의 저작물의 독점적인 저작권자가 된다. 즉, 아무도 자신의 기여를 사용, 복사, 배포 또는 수정할 수 없다.

프로젝트는 사용자가 알지 못하는 라이선스 요구 사항과의 종속성을 가질 수 있다. 프로젝트의 커뮤니티 또는 고용주의 정책에 따라 프로젝트에서 특정 오픈소스 라이선스를 사용해야 할 수도 있다.

예를 들어, Github에서 프로젝트를 공개하는 것은 프로젝트 라이선싱과 동일하지 않다. 공개 프로젝트는 Github의 서비스 약관에 명시되어 있으며, 다른 사람들이 프로젝트를 fork할 수는 있지만, 그렇지 않은 경우에는 권한이 없다.

다른 사람들이 프로젝트를 사용, 복사, 수정 또는 다시 사용할 수 있게 하려면, 오픈소스 라이선스를 포함해야 한다. 예를 들어, 권한을 부여하지 않는다는 조건에서는 공개적으로 Github 프로젝트의 일부를 코드에 명시적으로 사용할 수 없다.

2 오픈소스 소프트웨어 라이선스의 독점 규제 및 공정거래에 대한 법률상 쟁점

지식재산권의 행사를 통해 획득한 시장 지배력에 대한 소유는 그 자체로는 독점규제 및 공정거래에 관한 법률에 위반되지 않는다. 다른 소유권과 동일하게 지식재산권의 소유자에게 경쟁 수준을 초과하는 이익을 얻을 수 있게하는

‘우수한 제품이나 경영의 탁월성’의 결과로 발생한 시장지배력은 독점 규제 및 공정거래에 관한 법률에 위반하는 것이 아니라는 것이다. 그러나 다음과 같은 경우에 있어서는 공정거래법 위반여부에 대한 면밀한 판단이 필요하다.

가) 가격제한

오픈소스 소프트웨어는 원시 프로그램인 소스코드와 목적 프로그램이 동시에 공개 이용된다는 특징을 가진다. 그리고 이러한 특징은 누군가가 오픈소스 소프트웨어를 유료로 판매하는 것을 어렵게 만든다. 오픈소스 소프트웨어를 유료로 판매한다는 것은 경쟁 프로그램과의 관계에 있어 부당한 가격제한의 문제를 일으킬 수 있다. 이러한 특성으로 참여하고자 하는 프로그래머들과 기업들도 참여를 주저하는 경향이 있다. 따라서 최근에는 동일한 소프트웨어이지만 특정 시장에서는 오픈소스 소프트웨어로 배포하고 또 다른 시장에서는 카피라이트(copyright)로 배포하는 듀얼 라이선스(dual license)가 관심의 대상으로 떠오르고 있다. 이에 따라 라이선스 계약을 체결함에 있어서 오픈소스 소프트웨어 라이선스가 지식재산권을 이용한 상품의 판매가격 및 재판매 가격을 제한하는 가격제한에 해당하는지에 대한 의문이 제기 되고있다.

나) 거래상대방 제한

오픈소스 소프트웨어를 이용하여 개작하는 경우, 라이선스는 개작된 프로그램 전체를 오픈소스 소프트웨어 라이선스에 의해 다시 공개해야 할 의무를 가진다. 그리고 이러한 사항을 위반하였을 경우 애초부터 오픈소스 소프트웨어의 사용허락이 없었던 것으로 보고 오픈소스 소프트웨어 저작권 침해의 책임을 진다. 이러한 규정으로 인하여, 독점적 소프트웨어 저작권자는 자신의 소프트웨어를 창작하는데 오픈소스 소프트웨어를 이용할 수 없게 되었는데, 이러한 의무가 이용분야와 거래 상대방을 제한할 가능성이 존재한다는 점에서 공정거래법 위반여부에 해당하는지를

살펴보아야 한다. 지식재산권은 하나 이상의 용도를 가지는 경우가 많다. 따라서 이러한 제한들은 지식재산권자가 다양한 종류의 라이선스 계약을 체결함으로써 지식재산권자 자산이 관심을 가지고 있는 분야에서의 경쟁을 피할 수 있게 하고, 지식재산권의 효용을 최대한 활용할 수 있게 함으로 결과적으로 발명이나 창작 등의 사용허락을 촉진할 수 있게 한다. 그러나 권리소진의 원칙에 따라 상품이 판매된 후에 그 상품에 대하여 가해지는 제한은 특허권의 보호범위를 벗어난 것이기 때문에 경쟁제한성이 있다면 공정거래 위반에 해당될 수 있다. 그 밖에 라이선스 계약이 이용 분야 제한이나 거래상대방제한이 특허받은 공정을 이용하거나 특허받은 구성요소를 포함한 특허를 받지않은 상품에 대해 가해질 경우에 해당되거나, 노골적인 기업연합을 은폐하기 위한 위장에 지나지 않는 경우에 해당될 때에도 공정거래법 위반 문제가 발생할 수 있다.

다) 경쟁상품 취급제한

GNU GPL은 독점적 소프트웨어에 오픈소스 소프트웨어를 이용하지 못하도록 하고 있으며, 당해 오픈소스 소프트웨어에 저작권을 행사하는 저작물과 특허권을 행사하려는 기술 발명을 포함시키지 않도록 하고있다. 이러한 조건은 경쟁상품 취급제한의 문제를 야기할 수 있다. 경쟁상품 취급제한이란, 라이선스 계약을 체결할 때 지식재산권자의 특허와 경쟁관계에 있는 특허를 사용하지 못하도록 하거나 경쟁특허 사용시 지식재산권자의 승인을 받도록 하는 경우를 말한다. 이러한 행위는 구속조건부 거래행위에 해당할 수 있는데 지식재산권자가 시장지배력을 가지고 있지 않다면 경쟁상품 취급제한 규정은 경쟁을 제한하지 않을 것이기 때문이다. 따라서 시장 지배적 지위를 요건으로 하는 것이 타당하다.

공정거래위원회의 '지식재산권의 부당한 행사에 대한 심사지침' 제 3조 라.(4)에서는 '산업재산권자가 실시권자로 하여금 계약기간

중 또는 계약이 종료한 후 계약기술 제품 업종 등과 유사하거나 대체가 가능한 경쟁기술 제품 업종등을 사용 취급하지 못하도록 하거나 산업재산권자의 사전 동의 또는 승인을 받고 계약기간 중 경쟁제품을 취급하도록하는 경우 '불공정거래행위에 해당할 수 있다고 규정하고 있다. 반면 '산업재산권자가 실시권자로 하여금 계약기간 중 경쟁 제품을 취급할 경우 산업재산권자와 사전에 협의하도록 하는경우'는 위법으로 보기 어렵다고 본다. 오픈소스 소프트웨어는 시장 지배적 지위에 있다고 보기 어려운 것이 대부분이며 목적또한 공공의 이익을 증대시키기 위함이므로 경쟁상품 취급제한으로 보기는 어려울 것이라 생각된다. 하지만 오픈소스 소프트웨어가 관련시장에서 지배적 지위에 있다면 공공의 이익 증대와의 형평을 고려하여 판단하여야 할 것이다.

라) 그랜트백

그랜트백(Grant Back) 조항이란, 라이선서가 라이선시에게 계약을 통해 허락한 라이선스 기술에 대하여 라이선스가 그 이후에 개량한 기술에 대하여 사용할 권리를 라이선서에게 부여하는 것에 동의하여야 한다는 것을 내용으로 하는 계약을 말한다. 오픈소스 소프트웨어는 개작프로그램의 소스코드를 공개하고, 동일한 오픈소스 소프트웨어 라이선스의 조건으로 사용허락 할 것을 요구하고 있다. 그리고 이러한 조항은 그랜트백에 해당될 가능성이 있다. 그랜트백 조항은 개량기술에 대한 이용권을 실시권설정자에게만 귀속시키는 경우나 실시권설정자가 실시권자가 다른 기업에 대해서도 개량기술을 실시허락 할 수 있는 경우처럼 배타적 비 배타적인 경우를 포함하고있다. 그리고 그랜트백 조항은 특허를 받을 수 있는 개량기술인지 아닌지 여부에 관계없이, 심지어 특허와 일부 관련된 노하우까지 실시권설정자에게 부여하도록 포괄적으로 규정하는 경우도 있고, 지리적 시간적 범위 및 용도까지 제한적으로 규정하는 경우도 있다. 그랜트백 조항은 독자적으로 부여

되는 경우도 있고 다른 제한들의 일부로서 그랜트백 의무를 부과하는 경우도 있다. 심지어 리서치 툴(research tool)에 대한 라이선스 계약을 체결하면서 그 리서치 툴을 이용하여 개발된 약품 특허에 대한 그랜트백을 규정하는 경우와 같이 개량기술 이전 의무가 실시 허락된 최초의 지식재산권과 무관한 것인 경우도 있다. 라이선스 계약에 그랜트백 조항을 포함시키는 이유는 대부분 사업상의 정당한 필요에 의해서이다. 예를 들어 그랜트백 조항이 없다면 특허권자는 실시권자가 개량발명 할 기초를 제공하였음에도 불구하고 개량발명에 대한 이익은 전혀 공유하지 못한 채 자신의 특허가 구식이 되어 버릴 위험에 처하게 된다. 이 경우 라이선스 실시가 위축될 수 있을 것이다.

그랜트백 조항을 규정하는 정당한 이유는 다음과 같다. 첫째, 실시권자와 실시권설정자가 개량기술에 대한 위험을 공유할 수 있다. 둘째, 라이선스가 된 기술에 근거하거나 라이선스 된 기술에 의해 알려진 추가적인 혁신에 대해 실시권자에게 보상을 해줌으로써 최초의 혁신을 촉진한다. 셋째, 초기에 라이선스의 가치를 정확히 측정하기가 어려울 경우 이후에 실시권설정자가 일정한 가치를 취득할 수 있게 함으로써 라이선스 부여를 촉진할 수 있다. 넷째, 비 배타적 그랜트백은 장래의 개량 가능성 및 그 가치가 불분명할 때 높은 기술료 부과에 대한 대안이 될 수 있다. 다섯째, 장래의 상호저촉특허에 대한 우려를 줄일 수 있다. 그러나 이러한 그랜트백 조항은 경쟁을 저해하고 혁신을 방해하는 경우 공정거래법에 이반될 가능성이 있다. 그랜트백 조항이 공정거래법에 위반되는지에 대하여는 그랜트백 조항이 혁신에 미치는 효과, 그랜트백 조항의 범위, 그리고 실시권설정자의 시장 지배력 전이가능성 등을 종합적으로 고려하여 평가해야한다.

마) 라이선스 거절

사실상 오픈소스 소프트웨어 라이선스는 오픈소스 소프트웨어

를 이용하여 제작하는 순간 소스코드를 공개해야 할 의무가 발생하기 때문에, 오픈소스 소프트웨어 공동체의 이념에 동조하지 않는 이에게 라이선스 거절하는 효과가 있다. 그러나 이러한 규정에도 불구하고 점점 오픈소스 소프트웨어 라이선스를 이용하는 기업들이 증가하고 있다. 이는 오픈소스 소프트웨어 라이선스가 갖는 장점을 잘 설명해주는 것인데, 과거에는 오픈소스 소프트웨어가 소수 천재들이 자신의 이름을 널리 알리기위해 개발한 창작물로서 인식되었으나, 최근에는 독점적 소프트웨어의 개방방식에 비해 오픈소스 소프트웨어의 개발방식에 비해 오픈소스 소프트웨어의 개방방식이 저비용 및 개발성과 효율성을 극대화할 수 있다는 점이 부각됨에 따라 기업들은 오픈소스 소프트웨어를 이용한 프로젝트의 진행에 대해 진지하게 검토하기 시작했다.

지식재산권은 지식재산권자에게 독점권을 부여하고 기술혁신에 대하여 인센티브를 부여함과 동시에 기존에 없던 새로운 지식으로부터 발생하는 이익을 정당한 대가를 지불하지 않는다면 제 3자가 이용할 수 없게 배타적 권리를 부여한 것이다. 따라서 라이선스 거절에 관한 기본원칙은 지식재산권자는 자신의 지식재산권을 사용허락하지 않을 수 있다는 것을 유념하여야 한다. 다시말해 일반적으로 시장지배적 지위유무를 불문하고 지식재산권자는 지식재산권을 사용허락 할 의무를 지지않기 때문에 지식재산권자가 사용허락 하지 않았다는 사실만 가지고는 공정거래법상 거래거절이 성립될 수 없다.

그러나 예외적인 경우에는 공정거래법상 거래거절이 성립할 여지가 있다. 해당 지식재산권이 필수 설비에 해당하는 경우인데, Magill사건에서 유럽법원은 첫째, 방송사의 프로그램 정보는 대체제가 없는 필수 불가결한 원재료에 해당하고 둘째, 거절의 정당한 사유가 없으며, 셋째, 새롭게 형성되는 프로그램 가이드시장의 경쟁을 제한하기 때문에 라이선스의 거절이 남용행위에 해당한다

고 판단하였다.

바) 부당한 공동행위

오픈소스 소프트웨어 공동체에 참여하고 있는 기업들은 공동체 내의 기업들 간에 자신들이 소유한 지식재산권에 대하여 특허풀(Patent Pool) 또는 크로스 라이선스(Cross License)와 같은 공동행위를 발생시킨다는 점에서 공정거래법 위반여부가 문제될 수 있다.

먼저 크로스 라이선스란 라이선시와 라이선서 간에 체결한 라이선스에 의한 개량기술 혹은 실시과정 중에 라이선시가 취득한 특허권을 라이선서가 사용할 수 있게 상호 교환하는 라이선스, 그리고 상대방이 이미 보유하고 있는 특허나 우수한 기술을 서로 교환하여 사용하는 라이선스를 말한다. 특히 크로스 라이선스는 각 당사자가 상대방의 기술실시를 못하게 하는 특허 기술을 보유하고 있을 때 특히 유용하다. 다음으로 특허풀이란 서로 이용저촉 관계에 있는 특허권자들이 그들의 특허를 공동으로 라이선스하기 위해 단일의 주체에 특허권을 이전시키고 실시권을 허락받는 계약 관계를 말한다. 크로스 라이선스와 특허풀 이용 시 장점은 다음과 같다. 첫째, 보완적인 기술을 통합시킨다. 둘째, 거래비용을 감소시킨다. 셋째, 특허 간 상호저촉 상황을 해소시킨다. 넷째, 특허침해소송을 피할 수 있다. 다섯째, 기술의 보급을 촉진시킬 수 있다. 따라서 특허풀 혹은 크로스 라이선스 자체가 공정거래법상 당연히 위법한 것에 해당되는 것은 아니다. 그러나 상호 대체적 혹은 경쟁적 관계에 있는 지식재산권자들이 각자가 보유한 지식재산권과 관련하여 공동 실시 허락계약 혹은 크로스 라이선스를 체결함으로써 경쟁을 제한하는 효과를 발생시킬 경우 독점규제 및 공정거래에 관한 법률 제 4조의 규정을 적용할 수 있다.

3 오픈소스 소프트웨어 수용정책의 확산과 해외정책 동향

1990년대 후반부터 오픈소스 소프트웨어 개발방식에 대한 논의가 활성화되고, 확장성과 가용성 및 재사용율이 높다는 평가 아래 해외 주요국의 정부와 민간은 오픈소스 소프트웨어의 도입과 활용에 적극 나서고 있다.

그래서 우리나라와 해외 오픈소스 소프트웨어 정책 동향을 알아보하고자 한다.

가) 독일의 오픈소스 소프트웨어 정책

독일 연방의회는 연방정부 행정기관이 오픈소스 소프트웨어의 사용과 기반을 조성하도록 2001년에 권고하였다. 또한 독일 연방경제기술부와 독일 연방정보 기술 보안청에서는 BerliOS프로젝트, SPHINX프로젝트, GNUPG프로젝트 등 오픈소스 소프트웨어 사용 활성화를 위한 노력을 해왔다.

특히 독일연방내무부 산하에 전자정부관련 업무의 조정 및 자문역할을 수행하는 기관인 KBSt를 설치하여 공공부문에서 오픈소스 소프트웨어 활용을 지원해주도록 하고 있다. 2002년에는 독일 연방내무부는 IBM과 협약을 맺고 모든 공공기관이 IBM을 통해 오픈소스 소프트웨어를 저렴한 가격으로 조달할 수 있도록 했다.

뮌헨시의 경우 2003년 약 1만 4천대의 컴퓨터 소프트웨어를 리눅스로 전환하였으며, 시의회의 경우 2009년까지 클라이언트 컴퓨터에 오픈소스 소프트웨어를 사용하기로 하는 등 오픈소스 소프트웨어 정책을 적극적으로 추진하였다.

나) 프랑스의 오픈소스 소프트웨어 정책

프랑스는 오픈소스 소프트웨어 도입을 적극 추진했던 나라이

다. 2002년 상원은 모든 공공 행정기관에 오픈소스 소프트웨어 사용을 의무화하고, 자유소프트웨어기구를 설립하는 방안을 발의 하였으나 입법화시키지는 못했다. 그 대신 오픈소스 소프트웨어 활용 가이드 등을 통해 활성화를 유도해 오고 있다.

중앙 정부 수준에서는 내무부, 관세청, 재정경제산업부, 프랑스 국립경찰청 등에서 오픈소스 소프트웨어인 OpenOffice를 도입하였으며, 2004년 수립된 전자정부 계획의 일환으로 정부기관의 데스크톱에 오픈소스 소프트웨어를 설치하기로 하였다. 또한 ATICA, NIRIA등 다양한 기구를 통해 오픈소스 소프트웨어 정책을 적극적으로 추진해 오고 있다.

지방 정부 수준에서의 오픈소스 소프트웨어 추진상황을 보면 Arles의 경우, 2005년 시정부의 모든 지적재산권 응용서버들을 FOSS로 전환하는 작업을 진행하였고, 파리는 같은 해 395개 서버에 리눅스 운영체제를 도입하였다.

다) 미국의 오픈소스 소프트웨어 정책

미국의 경우 연방정부 차원보다는 주 정부 차원에서의 오픈소스 소프트웨어 활용정책이 활발히 추진되고 있다. 그러나 2003년 미 국방부의 오픈소스 사용에 대한 규정 제정과 OMB의 소프트웨어 구입에 있어 오픈소스 소프트웨어를 상용소프트웨어와 차별하지 않는다는 조달 중립적 정책 등으로 오픈소스 소프트웨어 도입을 활성화하고 있으며, 세계 오픈소스 소프트웨어 시장을 주도해 가고 있다.

주 정부 차원에는 오픈소스 소프트웨어 구매와 관련하여 법제화를 추진했던 사례가 다수 있는데, 2003년 오리건 주에서는 새로운 모든 소프트웨어를 획득할 경우 오픈소스 소프트웨어의 사용을 고려할 것을 요구하는 법안이 제출되었고, 캘리포니아 주 정부도 공공기관의 오픈소스 소프트웨어 사용의무를 지우는

‘Digital Software Security Act’ 를 상정하였다.

라) 남아프리카 공화국의 오픈소스 소프트웨어 정책

2000년대 초 남아프리카 공화국의 GITOC는 정부가 추진하는 전자정부 정책에서 공개 소프트웨어의 역할이 명확하게 인정되어야 한다는 결론을 내리고 이를 정부에 건의하였다. 이는 전자정부 등 정부의 IT 기술도입에 대한 비용이 증가하고 있는 상황에서 이미 제공되고 있는 공개소프트웨어가 제공할 수 있는 편익을 최대한 이용할 수 있도록 하기 위함이었다.

남아프리카 공화국 정부의 OSS정책은 크게 3가지이다. 첫째, 가치면에서 공개소프트웨어의 구현은 궁극적으로 가치를 창출하여야 한다. 경제적인 면에서는 외화 절감, 조달비용 절감이 있고 사회적인 면에서는 정보에 대한 접근권, IT교육에 대한 수단 제공이 있다. 둘째, 공개 소프트웨어를 구현하고 유지하기 위한 적절한 능력이 필요하다. 공개 소프트웨어를 구현하고 지원하기 위한 능력을 확대하기 위한 action steps와 정보의 제공, 전문가 조언, 개발지원 등을 정책으로 내세웠다. 셋째, 모든 주요 참여자들이 정책에 대해 충분한 지원을 할 것이다. 세부 정책으로는 컨퍼런스, 워크샵 등을 통해 공개 소프트웨어 관련 정보의 확산, 공개소프트웨어와 관련된 토론을 촉진하고 정보를 확산하기 위한 웹 사이트 구축 등이 있다.

마) 우리나라 오픈소스 소프트웨어 정책 동향

우리나라는 2003년부터 정부주도로 오픈소스 소프트웨어 활성화가 본격적으로 시작되었다. 그 이전 1990년대 후반에는 세계적으로 웹서버 등의 용도로 빠르게 확산되는 리눅스를 국내 기술로 전환하는데 집중하는 리눅스 육성정책이 주를 이루었다. 그러나 2002년 소프트웨어 산업 육성 중장기 계획에서 리눅스

육성정책이 제외되고 2003년 시행계획에서 '공개소프트웨어 활성화 추진' 정책이 포함되어 오늘에 이르고 있다.

한편, 정부의 '공개 소프트웨어 활성화' 정책과 관련되는 기관으로 2004년 설립된 '공개 소프트웨어 기술지원센터', '공개 소프트웨어 기술지원단'이 있다. 2007년에는 직접적인 기술지원업무는 민간시장 영역이라는 판단 하에 센터는 역할을 종료한다. 그러나 공공영역에서 지원해야 할 '공개 소프트웨어 기업 및 솔루션 현황, 정보제공 역할'등은 필요했으므로 2009년에 '공개 소프트웨어 역량프라자'를 개소하여 현재에 이르고 있다.

정부의 오픈소스 소프트웨어 활성화 정책은 조달부문의 제도개선으로 이어져 '공개 소프트웨어 관련기업의 공공부문 사업 참여 기회 확대, 입찰과정에서 기회 봉쇄 장벽제거'등이 이루어졌다. 특히 업무 시스템 고도화, 사이버 인프라 구축, 보건 의료 정보화 사업등 오픈소스 소프트웨어 도입효과가 큰 사업을 통해 상용소프트웨어보다 약 50%의 비용절감 효과를 얻을 수 있다는 결과를 얻었다.

또 오픈소스 소프트웨어에 대한 저작권 분쟁 등에 대비하기 오픈소스 소프트웨어 라이선스에 대한 분석지원 사업도 추진하였는데, 오픈소스 소프트웨어 라이선스에 대한 올바른 사용이 이루어지도록 하는 기반을 조성해오고 있다. 실제 2009년 미국의 소프트웨어 자유 법률센터(SFLC)로부터 국내기업이 라이선스 위반으로 제소된 경우를 생각해 볼 때 정부와 기업의 참여와 협력이 필요한 분야라 하였다. 2010년 오픈소스 소프트웨어가 적용된 국내 프로젝트 현황을 조사한 자료에 의하면 (IDC Korea 2010), 제조업 분야의 경우 삼성전자가 반도체 생산라인을 증설할 때 리눅스 기반의 서버를 배치하였으며, 휴대폰 네트워크 기지국 신호 모니터링 서버에 MySQL을 사용한 것으로 나타났다. 또한 LG그룹이 모바일 오피스 도입을 위한 모바일 변화 서비스

를 실시할 때 오픈소스 소프트웨어 기반의 서버를 적용하였다. 통신 분야에서는 SKT 멜론 서비스 등 과금 서비스를 포함한 웹 기반 서비스에 리눅스 OS를 도입하였고, SKT 대리점 관리 DB에 MySQL를 사용하였다.

<표 1> 우리나라 정부의 오픈소스 소프트웨어 활성화 정책 및 제도개선 내용³⁵⁾

(단위: 억 원)

연도	정책 추진 및 제도개선 내용	투자비
2003	▪ 공개SW인식제고, 법제도 연구 및 해외 선진사례 연구	10
2004	▪ 공개SW 불공정 경쟁 해소를 위한 법제도 개선 - 전자정부사업 공개SW도입 권고안, '공개SW 가이드' 발간 등 ▪ 공공기관 대상의 공개SW 시범 적용(8개 기관) ▪ 공개소프트웨어기술지원센터 개소 및 글로벌파트너십센터 운영	52
2005	▪ '공개SW기반 정보시스템구축 사용자 가이드' 제작 배포 ▪ 행당용PC 규격에 구체적인 리눅스 커널 버전을 명시 ▪ 한중일 웹표준 현황공유, 브라질·독일·EU 등과 정보교류 ▪ 공공기관 대상의 공개SW 시범 적용(11개 기관)	50
2006	▪ 공개SW 가이드(정통부) 및 공개SW 도입운영 가이드(국방부) 마련 ▪ 세출예산집행지침(기획예산처)에 공개SW 우선 검토 반영 ▪ '07년 예산안 작성 세부지침에 공개SW 우선 검토 반영 ▪ 정부업무평가관리기본법 세부 평가 지침에 공개SW 도입을 반영	63
2007	▪ '08년 예산편성 지침(기획예산처)에 홈페이지 신규구축 및 개선사업 추진 시 웹 표준 준수 의무화 반영 ▪ 공개SW 초기시장 창출 및 이용활성화를 위해 선도 시범사업 추진 ▪ 공개SW기반 국산 솔루션 테스트 지원 등	70
2008	▪ '공개SW기반 디지털교과서 서비스 환경구축 사업' 추진 ▪ 공공기관 대상 공개SW 도입 컨설팅(10개 기관) ▪ 나라장터 등을 통한 공정경쟁 모니터링, 공개SW라이선스 분석지원	80
2009	▪ 공개SW 역량프라자 구축 및 운영 ▪ 공개SW 기반 SW 교육혁신 프로젝트 지원 ▪ 공개SW 기반 디지털교과서 서비스 환경 고도화	54
2010	▪ 공개SW 역량프라자 운영 활성화 - 공개SW 기능/성능 Test, 공개SW Governance 고도화 등 ▪ 국내 공개SW 프로젝트 국제 기여도(2건)	80
2011	▪ 국제협력 강화(한중일, 한-브라질, 한-몽골 등) ▪ 국내 공개SW 프로젝트 국제 기여도(14건) ▪ 공개SW 시장 수요창출 가능 신규과제 발굴(9건) 및 컨설팅(40건)	75
2012	▪ 기획재정부 예산편성 지침 내 상용SW 도입 시 공개SW 병행검토 내용 반영 및 공개SW 유지 관리 서비스 가이드라인 공표 ▪ 공개SW 적용사업 발굴 및 지원(12건) 및 컨설팅 수행	69
2013	▪ 정보통신·방송 연구개발 관리규정 개정(공개SW관련 용어, 사업 성과활용 및 공개SW 특례조항 등 신설) ▪ 공개SW거버넌스 프레임워크 및 적용가이드 개발	64
2014	▪ 공개SW 개발과제 민간부담금 및 기술료 면제 ▪ 공개SW 전문기업 성장 촉진(공개SW 마케팅 지원 14건) 등	69
2015	▪ 개방형 OS 컴퓨팅 환경 시범개발 및 도입지원 착수 ▪ 글로벌 SW개발 프로젝트 참여 확대, 우수 공개SW 개발자 양성 등	128

출처

1. 오픈소스 소프트웨어 라이선스의 법적문제에 관한 고찰, 왕빛나
2. 오픈소스 소프트웨어 개발방식의 흐름 및 라이선스의 법적 쟁점, 한재용
3. 오픈소스 소프트웨어에 관한 경제적 연구
 - 정책 지향의 탐색적 접근 -, 김동규
4. 소셜 코딩 서비스 '깃허브'를 통한 오픈소스 소프트웨어
 공동체의 협력과 규제, 원인호
5. Apache logo: Wikipedia
6. git logo: sapjil.net
7. Mozilla logo: Wikipedia
8. Open Source logo: opensource.org
9. redhat logo: naver.com