

# 오픈 소스 소프트웨어가 공동체 사회에 끼치는 영향

정준영

경희대학교 전자·전파공학과, 컴퓨터공학과

[jjy920517@khu.ac.kr](mailto:jjy920517@khu.ac.kr)

## 목 차

1. 서론
2. 오픈 소스 소프트웨어의 의의
  - 2-1. 독점 소프트웨어 개발자들의 주장
  - 2-2. 오픈 소스 소프트웨어의 자유
  - 2-3. 오픈 소스 소프트웨어의 특성
3. 오픈 소스 소프트웨어의 문화와 공생
4. 결론

## 1. 서론

오픈 소스 소프트웨어(OSS, Open Source Software)는 소프트웨어의 본질인 ‘소스 코드(Source code)’가 공개되어 누구나 사용할 수 있는 소프트웨어를 의미한다[1]. 1970년대까지만 해도 소프트웨어의 소스 코드를 확보하는 것이 불가능한 것은 아니었다. 그러나 소프트웨어가 독립적으로 상품성이 있음에 눈을 뜬 기업들은 소프트웨어를 상업화했다. 이렇게 상업화 되어 출시된 소프트웨어를 독점 소프트웨어라 하며, 독점 소프트웨어의 소유자들은 소스 코드를 공유하는 것을 저작권 침해라 주장하였다. 그리고 저작권 침해를 해적 행위와 동일시 하며 소스 코드의 공유와 교환을 금지하였다. 오픈 소스 소프트웨어의 개념은 독점 소프트웨어에 대한 반발로 일어난 ‘GNU 프로젝트’를 통해 등장한다. GNU는 “GNU는 “GNU’s Not Unix”의 약자로, 널리 쓰이던 독점 소프트웨어 운영체제인 유닉스와 구별하자는 의미이다[2].

대표적인 오픈 소스 소프트웨어는 모바일 운영체제인 안드로이드(Android)가 있다. 안드로이드의 모든 소스 코드는 오픈 소스 라이선스인 ‘아파치 v2 라이선스’로 배포되어 있어, 기업이나 사용자는 각자 안드로이드 프로그램을 독자적으로 개발하여 탑재할 수 있다[3]. 따라서 수많은 스마트폰 제조 기업들은 안드로이드 운영체제를 자사 스마트폰에 적용하여 판매하였다. 이 결과 안드로이드는 전세계적으로 가장 많이 사용하는 모바일 운영체제가 되었다[4].

바둑기사 이세돌 9단과 대국을 한 인공지능 알파고(AlphaGo)도 오픈 소스 소프트웨어 라이브러리인 ‘텐서플로우 (TensorFlow)’를 사용하여 만들어졌다[5].

텐서플로우는 일반인들도 사용하기 쉽도록 공개되어 있으며, 이를 통해 많은 사람이 텐서플로우를 사용할 수 있어 알파고라는 인공지능이 탄생할 수 있었다.

본 논문에서는 2장에서 오픈 소스 소프트웨어가 가지는 의의를 알아보기 위해 독점 소프트웨어와 비교하여 그 특성을 살펴본다. 3장에서는 오픈 소스 소프트웨어를 통해 공동체 사회의 의미를 다시 생각해본다. 그리고 4장에서는 결론을 맺는다.

## 2. 오픈 소스 소프트웨어의 의의

### 2.1 독점 소프트웨어 개발자들의 주장

독점 소프트웨어 개발자들이 사용하는 “저작권 침해 행위를 중단하라”, “우리의 권리를 행사한다”와 같은 표현은 법률적이고 과격한 표현을 사용함으로써 대중들로 하여금 그들의 주장을 비판 없이 수용하도록 유도하는데 있다. 이런 표현 안에는 소프트웨어 제조 회사들이 소프트웨어를 유/무형으로 완전히 소유할 수 있는 천부적인 자연권을 갖고 있어서, 소프트웨어에 대한 사용자들의 권리를 통제할 수 있다는 전제가 포함되어 있다. 만약, 그들의 권리가 진정한 자연권이라면 대중이 그들로부터 어떠한 피해를 입는다 하더라도 이에 반대하거나 저항할 수 없을 것이다.

또한 독점 소프트웨어 개발자들은 소프트웨어를 만들도록 유도하기 위해 소유권자가 있어야 한다고 주장한다. 사용자를 만족시킨다는 정당한 목표에 기반을 두고, 개발자들이 적절한 대가를 받는다면 더 많은 소프트웨어를 만들 것이라는 의견이다. 이 의견은

정당해 보이지만 오픈 소스 소프트웨어를 잘못 이해한 주장이다. 오픈 소스 소프트웨어는 무료나 공짜라는 말이 내포하고 있는 금전적인 측면과의 자유가 아니라, “구속되지 않는다”는 관점에서의 자유를 의미한다. 이는 언론의 자유와 일맥상통하는 개념이다.

## 2.2 오픈 소스 소프트웨어의 자유

오픈 소스 소프트웨어가 주장하는 자유란 다음 네 가지 관점에서의 자유를 의미한다[6].

- 목적에 상관없이 프로그램을 실행시킬 수 있는 자유
- 필요에 따라서 프로그램을 수정할 수 있는 자유
- 무료 또는 유료로 프로그램을 재배포할 수 있는 자유
- 수정된 프로그램의 이익을 공동체 전체가 얻을 수 있도록 배포할 수 있는 자유

오픈 소스 소프트웨어는 사용자들에게 자유를 주는 것을 목적으로 한다. 따라서 오픈 소스 소프트웨어가 독점 소프트웨어로 변질 되는 것을 막을 수 있는 조항이 필요했다. 그 결과 지적재산권을 상징하는 ‘카피라이트(Copy Right)’에 반대적인 입장인 ‘카피레프트(Copy Left)’라는 방식을 사용했다. 카피 레프트는 저작권법을 그 근간으로 하지만 저작권법이 갖고 있는 주된 목적을 반대로 이용하여, 소프트웨어를 개인의 소유로 사유화시키는 대신 자유로운 상태로 유지할 수 있게 하였다. 즉 프로그램에 대한 실행과 복제, 수정, 배포의 모든 자유를 허용하는 것이다.

## 2.3. 오픈 소스 소프트웨어의 특성

오픈 소스 소프트웨어는 수정과 재배포에 제한이 없기 때문에 독점 소프트웨어에 비해 비용 효율적이라 할 수 있다[7]. 이는 오픈 소스 소프트웨어가 독점 소프트웨어에 비해 개발 비용이 저렴하기 때문이다[8].

오픈 소스 소프트웨어는 교차 점검을 통해서 품질을 높일 수 있으며, 보안에 있어서도 독점 소프트웨어에 비해 유리하다.

이외에 오픈 소스 소프트웨어와 독점 소프트웨어의 특성을 비교하여 <표 1>에 정리하였다[9].

## 3. 오픈 소스 소프트웨어의 문화와 공생

과거에는 독점 소프트웨어만 개발했던 기업들도 현재는 오픈 소스 소프트웨어 개발에 참여하고 있다. 그 이유는 기업 내부에서 소프트웨어에 대한 권한을 독점하며 개발하기엔 한계가 있기 때문이다. 개방, 공유,

협업이 점점 더 중요해지고 있는 최근에 오픈 소스 소프트웨어는 ICT 산업 경향에 부합하다. 또한 다양하고 복잡해지는 IT 기기와 환경의 연결성과 상호 운용성을 충족시킬 수 있는 대안으로 급 부상 중이다.

국가적인 차원에서도 오픈 소스 소프트웨어 생태계 형성에 지원을 아끼지 않고 있다. 국내 오픈 소스 소프트웨어 환경은 아직 취약한 상황이나, 외국은 벌써 오픈 소스 소프트웨어 정책을 추진하고 있다. 유럽 국가들은 보조금을 지급하기도 하고, 공공기관으로 하여금 기존 독점 소프트웨어를 오픈 소스 소프트웨어로 대체하게 한다.

이와 같이 오픈 소스 소프트웨어는 개인과 기업, 국가를 연결하는 하나의 공동체 문화라고 볼 수 있다. 이는 자본주의와 개인주의를 기반으로 발전해 온 현대사회의 특이점이다.

최근 대두되고 있는 ‘훈술’, ‘훈밥’ 문화는 타인과의 관계를 피곤해하는 현대인의 모습을 잘 보여주는 사례이다. 하지만 인간은 사회적 동물이며 집단을 이루어 생존한다. 지나친 개인주의로 자신의 삶에만 관심을 기울이다 사회라는 광범위한 시야를 잃어서는 안 된다. 따라서 오픈 소스 소프트웨어의 의의를 생각하며 타인과의 관계를 형성하는 공동체 의식의 회복이 필요하다.

## 4. 결론

현대 사회는 서로가 서로를 불편해하는 개인주의가 팽배하다. 자신의 기분이 안 좋아 다른 사람을 해치는 ‘묻지마 범죄’와 같은 사례가 개인주의의 폐해이다. 이러한 개인주의의 어두운 면은 자기 자신에만 초점이 강조된다는 점이다.

그러나 우리는 오픈 소스 소프트웨어를 통해 공동체의 의미를 다시 깨닫고 있다. 소프트웨어의 독점을 반대하고 모든 사용자에게 자유를 허용하며 함께 발전하는 오픈 소스 소프트웨어는 공동체 의식의 좋은 영향을 미치고 있다. 타인과의 관계에서 일상의 즐거움을 나누며 생산적인 일을 한다면, 서로를 인정하고 배려하는 건강한 사회가 될 것이다.

## 참고 문헌

- [1] 공개SW포털, <http://www.oss.kr>
- [2] GNU organization, <https://www.gnu.org>
- [3] Apache organization, <https://www.apache.org/licenses/LICENSE-2.0>
- [4] Gartner, <https://www.gartner.com>
- [5] TensorFlow organization, <https://www.tensorflow.org/>
- [6] GNU organization, <https://www.gnu.org>

구분	오픈 소스 소프트웨어	독점 소프트웨어
비용	<ul style="list-style-type: none"> <li>- 적용비용이 낮음</li> <li>- 유지비용이 낮고 기능 확장에 대한 추가비용이 들지 않음</li> <li>- 다수의 공개된 사용자에게 의한 관리로 관리자에 대한 관리 비용이 낮음</li> <li>- 서비스에 대한 비용 발생</li> </ul>	<ul style="list-style-type: none"> <li>- 적용비용이 적음</li> <li>- 유지비용 및 시스템 개선비용이 있음</li> <li>- 소수의 관리자에 대한 관리비용이 높음</li> <li>- 라이선스 수수료에 대한 비용 발생</li> </ul>
성능	<ul style="list-style-type: none"> <li>- 규모가 작은 시스템 환경에 성능이 높음</li> <li>- 높은 안정성 및 비용효율이 높음</li> </ul>	<ul style="list-style-type: none"> <li>- 규모가 큰 시스템 환경에서 비교적 성능이 높음</li> <li>- 고가의 장비로 인한 고성능</li> <li>- 전체적으로 오픈 소스 소프트웨어와 비슷한 성능</li> </ul>
보안	<ul style="list-style-type: none"> <li>- 개발 시부터 공개되어 있는 많은 취약점이 해결된 안전화 상태</li> <li>- 공개 키 기반의 인증 메커니즘 구현을 위한 통합패키지 존재</li> <li>- 다양한 암호화 알고리즘 및 키 관리에 대한 기능 제공</li> </ul>	<ul style="list-style-type: none"> <li>- 폐쇄적인 운영으로 인한 공개되지 않은 시스템 취약점 보유</li> <li>- 최근에 다수의 취약점 발견으로 많은 보안 위협에 노출</li> <li>- 프로토콜 호환이 어려워 인증체계가 취약함</li> </ul>
경제성	<ul style="list-style-type: none"> <li>- 총 소유비용 낮음</li> <li>- 라이선스 비용 없음</li> </ul>	<ul style="list-style-type: none"> <li>- 총 소유비용 높음</li> <li>- 구입비, 유지비 높음</li> </ul>
기술성	<ul style="list-style-type: none"> <li>- 재 사용성 높음</li> <li>- 유지보수, 업그레이드 용이</li> <li>- 독점 피해 방지</li> </ul>	<ul style="list-style-type: none"> <li>- 재 사용성 없음</li> </ul>
저작권	<ul style="list-style-type: none"> <li>- GPL, BSD (Copy Left)</li> </ul>	<ul style="list-style-type: none"> <li>- 일반 라이선스 (Copy Right)</li> </ul>
확장성	<ul style="list-style-type: none"> <li>- 효율적인 클러스터링 구현 가능</li> <li>- 소프트웨어 간의 호환성이 조금 떨어지나 적용비용이 거의 들지 않고 낮은 수주에선의 기능 추가 가능</li> </ul>	<ul style="list-style-type: none"> <li>- 서버의 가용성 측면에서 클러스터링 비효율성</li> <li>- 소프트웨어 간의 호환성이 보장되나 높은 적용 비용과 제한된 시스템 운영환경</li> </ul>
경쟁력	<ul style="list-style-type: none"> <li>- 공동 개발방식에 따른 교육 효과 우수</li> <li>- 우리나라와 같은 후발국에 적합</li> </ul>	<ul style="list-style-type: none"> <li>- 소프트웨어 선진국에 유리</li> </ul>

<표 1> 오픈 소스 소프트웨어와 독점 소프트웨어 비교

[7] 문상식, 김기홍, “IT환경 변화에 따른 한국의 오픈소스 소프트웨어의 정책 방향 연구”,

인터넷전자상거래연구 제 14권 제 1호 19 pages, 2014.

[8] Lerner, J. and Tirole, J., “The Economics of Technology Sharing: Open source and Beyond,” NBER Working Paper, No. 10956, 2004.

[9] 정보통신산업진흥원, 2012