

Version 1.0.0

개발자, 관리자, OS 전문가 및 기업을 위한 실무 지침

# 오픈소스 라이선스 해설

오픈소스 라이선스 준수 방법

Karsten Reincke Greg Sharpe

Version 1.0.0

개발자, 관리자, OS 전문가 및 기업을 위한 실무 지침

# 오픈소스 라이선스 해설

## 오픈소스 라이선스 준수 방법\*

Karsten Reincke<sup>†</sup>      Greg Sharpe<sup>‡</sup>

2015. 3. 2

### 저작권과 이용허락 안내

\*) 본 문서 내용은 Creative Commons Attribution-ShareAlike 3.0 라이선스

(<http://creativecommons.org/licenses/by-sa/3.0/de/>)에 의해 사용 허가된 내용입니다. “본 문서와 동일하거나 유사한 라이선스에 의한 저작물을 배포”하고 “저자가 지정한 방식의 저작물로 판단”한다면 제약 없이 “공유(복제, 배포 및 전송하기 위한)” 또는 “수정”을 할 수 있습니다. :

인터넷에서 재사용 시 적절한 방식으로 최초 저자들을 언급하고, 이들의 스폰서 도이치텔레콤(주)(Deutsche Telekom AG) 명칭을 제시하며, <http://www.telekom.com>에 링크를 적용하기 바랍니다. 문서 유형으로 재사용 시에는 서문에서 <http://www.telekom.com>, 최초 저자 그리고 스폰서인 도이치텔레콤(주)에 대한 내용을 간략하게 제시하기 바랍니다. 일반적인 인용문인 경우에는 과학 표준 방식을 사용하면 됩니다.

[*myCsrf* (= 'mind your Scholar Research Framework') c K. Reincke CC BY 3.0 <http://mycsrf.fodina.de/>]를 기초한 내용]

†) Deutsche Telekom AG, Products & Innovation, T-Online-Allee 1, 64295 Darmstadt

‡) Deutsche Telekom AG, Telekom Deutschland GmbH, Landgrabenweg, Bonn

본 문서 내용은 정보통신산업진흥원이 위 저작권 이용허락에 따라 번역하여 배포합니다. (2017년 6월 30일)

본 번역 내용에 대한 오탈자 및 잘못된 내용에 대한 수정 정보는 공개SW포털 홈페이지나 아래 이메일로 보내주시면 감사하겠습니다.

\*공개SW포털 홈페이지 [www.oss.kr](http://www.oss.kr) / 이메일 oss@nipa.kr

오픈소스 커뮤니티에는 다수의 사람들이 활동하고 있습니다. (임의로 선정된 전문가들보다 더 영향력 있음).

이 커뮤니티로부터 지원을 받는데 대해 영광으로 생각합니다.

이분들에게 감사를 드립니다. (알파벳 순서로 열거):

Eitan Adler,

Stefan Altmeyer (Deutsche Telekom AG),

Ronald Dauster,

John Dobson,

Steffen Härtlein,

Ta'ld Holmes (Deutsche Telekom AG),

Michael Kern (Deutsche Telekom AG),

Michael Machado (Deutsche Telekom AG),

Thorsten Müller (Deutsche Telekom AG),

Tanja Neske (Deutsche Telekom AG),

Oliver Podebradt (Deutsche Telekom AG),

Thomas Quiehl (Deutsche Telekom AG),

Peter Schichl (Deutsche Telekom AG),

Michael Schierl,

Helene Tamer (T-Systems International AG),

Bernhard Tsai (Deutsche Telekom AG),

Thomas Weischuh (Amadeus Germany GmbH),

와

# 목차

1 서론 .....	12
2 오픈소스: 동일한 관념, 다양한 라이선스 .....	17
2.1 GNU Affero General Public License(GPL)의 보호 기능 .....	28
2.2 아파치 라이선스(Apache-2.0)의 보호 기능 .....	28
2.3 BSD 라이선스의 보호 기능 .....	28
2.4 CDDL 라이선스의 보호 기능 [추후 보완] .....	29
2.5 Eclipse Public License(EPL)의 보호 기능 .....	29
2.6 European Union Public License(EUPL)의 보호 기능 .....	31
2.7 GNU General Public License(GPL)의 보호 기능 .....	32
2.7.1 GPL-2.0 .....	33
2.7.2 GPL-3.0 .....	33
2.8 Lesser GNU General Public License(LGPL)의 보호 기능 .....	34
2.8.1 LGPL-2.1 .....	35
2.8.2 LGPL-3.0 .....	36
2.9 MIT 라이선스의 보호 기능 .....	37
2.10 Mozilla Public License(MPL)의 보호 기능 .....	37
2.11 Microsoft Public License(MS-PL)의 보호 기능 .....	39
2.12 Postgres License(PostgreSQL)의 보호 기능 .....	39
2.13 PHP 라이선스의 보호 기능 .....	40
2.14 요약 .....	40
3 오픈소스: 일부 부작용에 대해 .....	43
3.1 불명확하게 특허를 공개하는 문제 .....	43
3.1.1 AGPL 특허 관련 조항 .....	46
3.1.2 Apache-2.0 특허 관련 조항 .....	47
3.1.3 CDDL 특허 관련 조항 .....	48
3.1.4 EPL 특허 관련 조항 .....	49
3.1.5 EUPL 특허 관련 조항 .....	50
3.1.6 GPL 특허 관련 조항 .....	50
3.1.6.1 GPL-2.0 .....	50
3.1.6.2 GPL-3.0 .....	51
3.1.7 LGPL 특허 관련 조항 .....	51
3.1.7.1 LGPL-2.1 .....	51
3.1.7.2 LGPL-3.0 .....	52
3.1.8 MPL 특허 관련 조항 .....	52
3.1.9 MS-PL 특허 관련 조항 .....	53

3.2 부가사항: 링킹이 부차적 기준이 되는 이유 .....	53
3.3 부가사항: 오픈소스의 기본 개념 - ‘파생저작물’이란? .....	57
3.4 부가사항: 리버스 엔지니어링과 오픈소스 .....	59
3.4.1 LGPL-2의 리버스 엔지니어링 .....	63
3.4.1.1 언어적 해석 .....	63
3.4.1.2 논리적 해석 .....	66
3.4.1.3 경험적 해석 .....	69
3.4.1.4 최종 결론 .....	71
3.4.1.4.1 라이브러리의 일부를 수동으로 복제한 저작물 배포 시 Copyleft 가 적용된다. ....	71
3.4.1.4.2 스크립트 배포 시 리버스 엔지니어링을 필요로 하지 않는다. ....	72
3.4.1.4.3 정적 결합된 바이트 코드 배포 시 리버스 엔지니어링 허가가 필요하다 .....	73
3.4.1.4.4 정적 결합된 바이너리 배포 시 리버스 엔지니어링 허가가 필요하다 .....	73
3.4.1.4.5 동적 결합이 가능한 바이트 코드 및 링킹 가능한 오브젝트 코드는 리버스 엔지 니어링 허가를 필요로 하지 않는다. ....	75
3.4.1.4.6 리버스 엔지니어링 허가가 있을 때와 없을 때의 LGPL-2.0 컴플라이언스 .....	76
3.4.1.5 최종 확보 .....	77
3.4.2 LGPL-3.0의 리버스 엔지니어링 .....	79
3.4.3 기타 오픈소스 라이선스의 리버스 엔지니어링 .....	84
3.4.4 오픈소스 라이선스의 리버스 엔지니어링: 개요 .....	86
3.5 부가사항: 라이선스 준수성 문제 [추후 보완] .....	87
3.6 부가사항: 오픈소스 소프트웨어와 수익 [추후 보완] .....	87
 4 오픈소스 사용 사례: 개념과 분류체계 .....	88
 5 오픈소스 사용 사례: 과업 목록(To-Do List)을 충족시키는 라이선스 찾기 .....	92
5.1 관련 정보 수집을 위한 표준 포맷 .....	92
5.2 분류적 오픈소스 사용 사례 파인더 .....	94
5.3 오픈소스 사용 사례 및 그 과업 목록 참고자료 .....	96
 6 오픈소스 라이선스 준수: 과업 목록 .....	107
6.1 누군가에게 파일을 ‘제공’함에 있어서의 일반적 설명 .....	107
6.2 GPL 라이선스 소프트웨어 .....	108
6.2.1 GPL-3.0-C1: 단독으로 소프트웨어 사용 .....	108
6.2.2 GPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포 .....	109
6.2.3 GPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포 .....	109
6.2.4 GPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포 .....	110
6.2.5 GPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포 .....	111
6.2.6 GPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포 .....	112
6.2.7 GPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포 .....	113

6.2.8 AGPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	114
6.2.9 AGPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	115
6.2.10 AGPL-3.0-CA 수정한 라이브러리를 임베디드 소스코드 형태로 배포	116
6.2.11 AGPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	117
6.2.12 AGPL-3.0-CC: 수정한 프로그램을 네트워크와 연계하여 실행	118
6.2.13 AGPL-3.0-CD: 수정한 임베디드 라이브러리를 네트워크와 연계하여 실행	120
6.2.14 논의 및 설명	121
<b>6.3 아파치 라이선스 소프트웨어</b>	<b>125</b>
6.3.1 Apache-2.0-C1: 단독으로 소프트웨어 사용	125
6.3.2 Apache-2.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포	126
6.3.3 Apache-2.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포	126
6.3.4 Apache-2.0-C4: 수정한 프로그램을 소스코드 형태로 배포	127
6.3.5 Apache-2.0-C5: 수정한 프로그램을 바이너리 형태로 배포	128
6.3.6 Apache-2.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	129
6.3.7 Apache-2.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	129
6.3.8 Apache-2.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	130
6.3.9 Apache-2.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	131
6.3.10 논의 및 설명	132
<b>6.4 BSD 라이선스 소프트웨어</b>	<b>133</b>
6.4.1 BSD-3-Clause-C1: 단독으로 소프트웨어 사용	134
6.4.2 BSD-3-Clause-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포	135
6.4.3 BSD-3-Clause-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포	135
6.4.4 BSD-3-Clause-C4: 수정한 프로그램을 소스코드 형태로 배포	136
6.4.5 BSD-3-Clause-C5: 수정한 프로그램을 바이너리 형태로 배포	136
6.4.6 BSD-3-Clause-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	137
6.4.7 BSD-3-Clause-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	137
6.4.8 BSD-3-Clause-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	138
6.4.9 BSD-3-Clause-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	139
6.4.10 BSD-2-Clause-C1: 단독으로 소프트웨어 사용	139
6.4.11 BSD-2-Clause-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포	140
6.4.12 BSD-2-Clause-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포	140
6.4.13 BSD-2-Clause-C4: 수정한 프로그램을 소스코드 형태로 배포	141
6.4.14 BSD-2-Clause-C5: 수정한 프로그램을 바이너리 형태로 배포	141
6.4.15 BSD-2-Clause-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	142
6.4.16 BSD-2-Clause-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	142
6.4.17 BSD-2-Clause-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	143
6.4.18 BSD-2-Clause-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	144
6.4.19 논의 및 설명	144
<b>6.5 CDDL 라이선스 소프트웨어 [추후 보완]</b>	<b>146</b>

6.5.1 CDDL-1: 단독으로 소프트웨어 사용 .....	147
6.5.2 CDDL-2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포 .....	147
6.5.3 CDDL-3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포 .....	147
6.5.4 CDDL-4: 수정한 프로그램을 소스코드 형태로 배포 .....	148
6.5.5 CDDL-5: 수정한 프로그램을 바이너리 형태로 배포 .....	148
6.5.6 CDDL-6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포 .....	148
6.5.7 CDDL-7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포 .....	149
6.5.8 CDDL-8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포 .....	149
6.5.9 CDDL-9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포 .....	149
6.5.10 논의 및 설명 .....	150
<b>6.6 EPL-1.0 라이선스 소프트웨어 .....</b>	<b>150</b>
6.6.1 EPL-1.0-C1: 단독으로 소프트웨어 사용 .....	151
6.6.2 EPL-1.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포 .....	152
6.6.3 EPL-1.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포 .....	152
6.6.4 EPL-1.0-C4: 수정한 프로그램을 소스코드 형태로 배포 .....	153
6.6.5 EPL-1.0-C5: 수정한 프로그램을 바이너리 형태로 배포 .....	154
6.6.6 EPL-1.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포 .....	155
6.6.7 EPL-1.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포 .....	156
6.6.8 EPL-1.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포 .....	157
6.6.9 EPL-1.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포 .....	158
6.6.10 논의 및 설명 .....	159
<b>6.7 EUPL-1.1 라이선스 소프트웨어 .....</b>	<b>161</b>
6.7.1 EUPL-1.1-C1: 단독으로 소프트웨어 사용 .....	162
6.7.2 EUPL-1.1-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포 .....	162
6.7.3 EUPL-1.1-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포 .....	163
6.7.4 EUPL-1.1-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포 .....	164
6.7.5 EUPL-1.1-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포 .....	164
6.7.6 EUPL-1.1-C6: 수정한 프로그램을 소스코드 형태로 배포 .....	165
6.7.7 EUPL-1.1-C7: 수정한 프로그램을 바이너리 형태로 배포 .....	166
6.7.8 EUPL-1.1-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포 .....	167
6.7.9 EUPL-1.1-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포 .....	168
6.7.10 EUPL-1.1-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포 .....	169
6.7.11 EUPL-1.1-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포 .....	169
6.7.12 논의 및 설명 .....	170
<b>6.8 GPL 라이선스 소프트웨어 .....</b>	<b>172</b>
6.8.1 GPL-2.0-C1: 단독으로 소프트웨어 사용 .....	173
6.8.2 GPL-2.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포 .....	173
6.8.3 GPL-2.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포 .....	174
6.8.4 GPL-2.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포 .....	175

6.8.5 GPL-2.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포	176
6.8.6 GPL-2.0-C6: 수정한 프로그램을 소스코드 형태로 배포	177
6.8.7 GPL-2.0-C7: 수정한 프로그램을 바이너리 형태로 배포	177
6.8.8 GPL-2.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	179
6.8.9 GPL-2.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	180
6.8.10 GPL-2.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	181
6.8.11 GPL-2.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	182
6.8.12 GPL-3.0-C1: 단독 <u>소프트웨어</u> 사용	183
6.8.13 GPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포	183
6.8.14 GPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포	184
6.8.15 GPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포	185
6.8.16 GPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포	185
6.8.17 GPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포	186
6.8.18 GPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포	187
6.8.19 GPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	188
6.8.20 GPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	189
6.8.21 GPL-3.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	190
6.8.22 GPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	191
6.8.23 논의 및 설명	192
<b>6.9 LGPL 라이선스 <u>소프트웨어</u></b>	<b>196</b>
6.9.1 LGPL-2.1-C1: 단독으로 소프트웨어 사용	197
6.9.2 LGPL-2.1-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포	198
6.9.3 LGPL-2.1-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포	198
6.9.4 LGPL-2.1-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포	199
6.9.5 LGPL-2.1-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포	200
6.9.6 LGPL-2.1-C6: 수정한 프로그램을 소스코드 형태로 배포	201
6.9.7 LGPL-2.1-C7: 수정한 프로그램을 바이너리 형태로 배포	201
6.9.8 LGPL-2.1-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	202
6.9.9 LGPL-2.1-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	202
6.9.10 LGPL-2.1-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	203
6.9.11 LGPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	204
6.9.12 LGPL-3.0-C1: 단독으로 소프트웨어 사용	206
6.9.13 LGPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포	206
6.9.14 LGPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포	207
6.9.15 LGPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포	208
6.9.16 LGPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포	208
6.9.17 LGPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포	209
6.9.18 LGPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포	210
6.9.19 LGPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	211

6.9.20 LGPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	212
6.9.21 LGPL-3.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	213
6.9.22 LGPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	214
6.9.23 논의 및 설명	215
<b>6.10 MIT 라이선스 소프트웨어</b>	<b>219</b>
6.10.1 MIT-C1: 단독으로 소프트웨어 사용	220
6.10.2 MIT-C2: 수정하지 않은 소프트웨어 배포	220
6.10.3 MIT-C3: 수정한 프로그램 배포	221
6.10.4 MIT-C4: 수정한 라이브러리를 독립적으로 배포	221
6.10.5 MIT-C5: 수정한 라이브러리를 임베디드 컴포넌트로 배포	222
6.10.6 논의 및 설명	223
<b>6.11 MPL-2.0 라이선스 소프트웨어</b>	<b>223</b>
6.11.1 MPL-2.0-C1: 단독으로 소프트웨어 사용	224
6.11.2 MPL-2.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포	225
6.11.3 MPL-2.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포	225
6.11.4 MPL-2.0-C4: 수정한 프로그램을 소스코드 형태로 배포	226
6.11.5 MPL-2.0-C5: 수정한 프로그램을 바이너리 형태로 배포	227
6.11.6 MPL-2.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포	228
6.11.7 MPL-2.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포	229
6.11.8 MPL-2.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	230
6.11.9 MPL-2.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	231
6.11.10 논의 및 설명	233
<b>6.12 Microsoft Public 라이선스</b>	<b>234</b>
6.12.1 MS-PL-C1: 단독으로 소프트웨어 사용	235
6.12.2 MS-PL-C2: 수정하지 않은 소프트웨어 배포	235
6.12.3 MS-PL-C3: 수정한 프로그램을 소스코드 형태로 배포	237
6.12.4 MS-PL-C4: 수정한 프로그램을 바이너리 형태로 배포	237
6.12.5 MS-PL-C5: 수정한 라이브러리를 돋립적인 소스코드 형태로 배포	237
6.12.6 MS-PL-C6: 수정한 라이브러리를 돋립적인 바이너리 형태로 배포	238
6.12.7 MS-PL-C7: 수정한 라이브러리를 임베디드 소스코드 형태로 배포	238
6.12.8 MS-PL-C8: 수정한 라이브러리를 임베디드 바이너리 형태로 배포	239
6.12.9 논의 및 설명	239
<b>6.13 PostgreSQL 라이선스 소프트웨어</b>	<b>240</b>
6.13.1 PostgreSQL-C1: 단독으로 소프트웨어 사용	241
6.13.2 PostgreSQL-C2: 수정하지 않은 소프트웨어 배포	241
6.13.3 PostgreSQL-C3: 수정한 프로그램 배포	241
6.13.4 PostgreSQL-C4: 수정한 라이브러리를 돋립적으로 배포	242
6.13.5 PostgreSQL-C5: 수정한 라이브러리를 임베디드 컴포넌트로 배포	242
6.13.6 논의 및 설명	243

6.14 PHP-3.0 라이선스 소프트웨어 .....	243
6.14.1 PHP-3.0-C1: 단독으로 소프트웨어 사용 .....	244
6.14.2 PHP-3.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포 .....	244
6.14.3 PHP-3.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포 .....	245
6.14.4 PHP-3.0-C4: 수정한 프로그램을 소스코드 형태로 배포 .....	246
6.14.5 PHP-3.0-C5: 수정한 프로그램을 바이너리 형태로 배포 .....	246
6.14.6 PHP-3.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포 .....	247
6.14.7 PHP-3.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포 .....	247
6.14.8 PHP-3.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포 .....	248
6.14.9 PHP-3.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포 .....	249
6.14.10 논의 및 설명 .....	250
7 결론 .....	251
8 부록 .....	253
8.1 OSLiC 인용 스타일에 관한 몇 가지 추가 설명 .....	253
8.2 일부 오픈소스에 관한 오해 .....	254
8.2.1 이유(Why) .....	256
8.2.2 무엇(What) .....	263
간행물, 쇼트컷, 약어 .....	265
참고문헌 .....	268

## 주요 부분

- 작업 목록에 AL, AFL, CDDL, MPL-1.[0|1], MS-RL, OSL 삽입
- 소프트웨어 개발 측면에서 파생저작물(derivative work) 개념 설명을 개선 / 정리
- proapose를 snimoli 및 v.v로 변환하는 수정사항 다루기
- 라이선스 호환성 논의
- 오픈소스와 수익간의 관계 설명
- 간행물 리스트 업데이트

표 0.1: 오픈소스 라이선스 해설(Open Source License Compendium) 진행 경과

2015-03-01	1.0.0	<ul style="list-style-type: none"> <li>■ 발표           <ul style="list-style-type: none"> <li>• AGPL관련 6번째 문항 추가</li> <li>• OSUC-03에 AGPL 서브타입 L(octal) &amp; I(nternet) 내용 추가</li> <li>• AGPL 파인더 및 라이선스 준수 과업 목록 추가</li> </ul> </li> </ul>
2015-01-21	0.99.9	<ul style="list-style-type: none"> <li>• 리버스 엔지니어링 도전에 대한 솔루션 추가</li> </ul>
2014-03-09	0.99.1	<ul style="list-style-type: none"> <li>• LATEXsource로부터 OSCAd사용 용도로 데이터파일 생성</li> <li>• LGPL C9 사례의 버그 수정</li> <li>• 6장의 일반적인 수정</li> </ul>
2014-01-08	0.98.2	<ul style="list-style-type: none"> <li>• CDDL 내 특허관련 문구에 대한 섹션 추가</li> <li>• PDF 파일에 하이퍼링크 추가 (hyperref, pdftex 사용)</li> <li>• 1장부터 5장까지 일반적인 수정</li> </ul>
2013-11-27	0.98.1	<ul style="list-style-type: none"> <li>■ 한국 FLOSS 컨퍼런스 발표</li> </ul>
2013-08-19	0.97.2	<ul style="list-style-type: none"> <li>• M.Schierl의 제안한 오타 수정</li> <li>• 파생저작물 관련 내용 추가</li> <li>• OSLiC은 프로토타입 사례에 대해 다룬다는 내용 추가</li> </ul>
2013-07-28	0.97.1	<ul style="list-style-type: none"> <li>• 간접적으로 사용된 2차 문헌 추가</li> <li>• LGPL 특성 파인더 개선</li> <li>• OSCAd와 일치, 인터페이스 개선</li> </ul>
2013-05-20	0.96.1	<ul style="list-style-type: none"> <li>■ Linux Days 발표           <ul style="list-style-type: none"> <li>• 오픈소스 사용 사례 라이선스 특정 사용사례 명명 변경</li> <li>• OSCAd 컨텐츠와 버전 일치</li> </ul> </li> </ul>
2013-04-15	0.95.2	<ul style="list-style-type: none"> <li>■ 유럽자유소프트웨어재단(FSFE) LLW 후 발표           <ul style="list-style-type: none"> <li>• 알려진 대부분의 OSI 라이선스의 과업 목록(to-do list)</li> <li>• 항상된 GPL과 EUPL 파인더</li> <li>• 간소화된 형태와 개선된 구조로 된 OSLiC 파인더</li> <li>• 통합된 부문과 간행된 신규 원본</li> </ul> </li> </ul>
2013-04-05	0.95.1	<ul style="list-style-type: none"> <li>■ FSFE LLW 전 발표           <ul style="list-style-type: none"> <li>• 허용되는 모든 약한 효력의 카피레프트 라이선스의 과업 목록</li> <li>• 통합된 부문과 간행된 신규 원본</li> </ul> </li> </ul>
2013-03-15	0.94.1	<ul style="list-style-type: none"> <li>■ Chemnitzer Linux Day 공개           <ul style="list-style-type: none"> <li>• 허용되는 모든 weak copyleft 라이선스의 과업 목록</li> <li>• 부문 통합과 신규 원본 간행</li> </ul> </li> </ul>
2013-03-08	0.90.1	<ul style="list-style-type: none"> <li>■ CeBIT 공개           <ul style="list-style-type: none"> <li>• 추가된 일부 주요 라이선스의 과업 목록</li> <li>• 부문 통합과 신규 원본 간행</li> </ul> </li> </ul>
2013-02-16	0.8.90	<ul style="list-style-type: none"> <li>■ CeBIT 전 공개           <ul style="list-style-type: none"> <li>• 주요 라이선스 이행사항에 중점을 두어 새로 논의되는 구조</li> <li>• 새로이 가름하는 라이선스 검토</li> <li>• 하향식의 새 서론 내용</li> </ul> </li> </ul>
2012-12-28	0.8.0	<ul style="list-style-type: none"> <li>■ 내부 연말(EOY) 공개           <ul style="list-style-type: none"> <li>• branch kreibnck에 통합된 편재되어 있는 대다수의 개선 사항</li> </ul> </li> </ul>
2012-08-25	0.5.2	<ul style="list-style-type: none"> <li>■ 공개를 통한 변경내용 확장           <ul style="list-style-type: none"> <li>• MIT 라이선스를 충족시키는 과업 목록</li> <li>• Eclipse 스펠 체킹 방법 사용</li> </ul> </li> </ul>
2012-07-06	0.4.0	<ul style="list-style-type: none"> <li>■ 공개를 통한 변경내용           <ul style="list-style-type: none"> <li>• 오픈소스 사용 사례 정의 및 분류체계</li> <li>• 오픈소스 사용 사례에 입각한 파인더</li> <li>• 해당 BSD 특정 미니 파인더</li> <li>• BSD 라이선스를 충족하는 과업 목록</li> </ul> </li> </ul>

2012-03-22	0.2.1	<ul style="list-style-type: none"> <li>커뮤니티 초판으로 간행된 글자</li> </ul>
2012-01-31	0.1.8	<ul style="list-style-type: none"> <li>머리말로 기존의 서론에 관한 재차 설명</li> <li>더 간략하게 하향식으로 작성된 서론 삽입</li> <li>OSLiC 경고문 삽입</li> <li>oscCopiedButNotRead.bib 확대</li> <li>정기 간행물 및 쇼트 컷 리스트 추가</li> <li>많은 문헌 데이터 추가</li> </ul>
2011-09-29	0.1.4	<ul style="list-style-type: none"> <li>문서 이력 통합</li> <li>오자 삭제</li> <li>영어 교사의 리뷰 포함</li> <li>John의 개선의견 포함</li> </ul>
2011-09-12	0.1.0	<ul style="list-style-type: none"> <li>서론 정리: 목적 및 방법</li> </ul>

## 경고문

본 저작물은 오픈소스 커뮤니티와 공동 작업을 통해 앞으로도 충분히 더 보완되어야 한다. 그리고 궁극적으로 '한 마리의 물고기보다 무리의 물고기가 더 많이 알고 있다'는 통찰력을 인정하는 인식 위에서 신뢰할만한 정보가 전달되어야 한다.

그럼에도 불구하고 본 문서(OSLiC, Open Source License Compendium)는 저자와 기고자들이 제시하는 의견 이상을 제공할 수는 없다. 본 해설서의 목적은 오픈소스 라이선스에 대한 논의를 단지 한목소리로 전달하는 것이다. 저자와 기고자를 보상청구 및 피소로부터 보호하기 위해 약간의 수정을 거친 GPL3 경고문을 채택하였다 :

본 문서(OSLiC)는 법률 적용에 의해 허용되는 범위까지 어떤 보증도 제공되지 않는다. 저작권 소유자 및 기타 당사자들은 특정 용도에 대한 적합성과 상품성을 은연중에 보증하는 것을 포함하여 명시적이든 묵시적이든 보증하지 않고 "있는 그대로" 내용을 제공하는 것이다. 본 문서(OSLiC)의 품질과 이행과 관련되는 전체 리스크는 사용자가 부담하여야 한다. 본 문서(OSLiC)에 결함이 있는 것으로 판단되면 사용자는 필요한 모든 정비, 정정 또는 수정에 대한 비용을 부담하여 수정할 수 있다.

적용 가능한 법률에 의해 요구되거나 서면 합의가 이루어지지 않는 한 어떠한 경우에도 저작권자와 본 문서(OSLiC)를 수정 및 제공하는 기타 당사자는 손해 배상 가능성에 대해 통지를 받았더라도 본 문서(OSLiC)를 사용하거나 (자료의 손실, 부정확하게 제공되는 자료, 사용자나 제3자가 입는 손실, 다른 의미의 기술과 본 문서상의 잘못인용 등을 포함하는) 사용 불능으로 인해 발생하는 일반적, 특수적, 부수적, 결과적 손해를 포함하여 손해 배상에 대한 책임을 지지 않는다.

특히 본 문서(OSLiC)는 사용자에게 있어서 변호사에 의한 법률 조언 또는 법적 검토를 대신할 수 없고 대체될 수 없다. 그러나 법적 상담인이<sup>1)</sup> 최종적으로 검토해야 하는 해결 방법을 찾는데 개발자, 관리자, 오픈소스 전문가 및 기업에게 기여할 수는 있다.

---

1) 독일 독자의 경우: OSLiC는 물론 독일의 '비사법적 법률 서비스에 관한 법률(RDG, Rechtsdienstleistungsgesetz)'을 중요시한다. 일반 대중에게 언급하는 법적 의견을 담고 있을 뿐이다. OSLiC는 일반 대중에게 법적 문제와 관련된 정보를 전달하기 위한 목적으로만 사용하는 것으로서 이해할 수 있다.

## 1 서론

이번 장에서는 OSLiC의 기본 개념, 사용 방법 및 실제로 이해할 수 있는 방법에 대해 간략하게 설명한다.

이 문서는 '우리가 사용하고 있는 오픈소스 소프트웨어의 라이선스에 따라 행동하기 위해 필요한 것은 무엇인가?'라는 오직 한 가지 주제에만 초점을 맞춘다. 오픈소스 라이선스 해설(Open Source License compendium)은 이 질문에 간단하고 쉽게 방법을 이해하는 방향으로 신뢰성 있는 답변을 하는 것을 목적으로 한다. 그러나 본 해설서는 다른 일반적인 오픈소스에 관한 문서들과는 다르다.<sup>2)</sup> 오히려 라이선스 준수를 위한 활동을 단순화하기 위한 도구가 되는 것이 그 의도이다.

본 해설서는 도이치텔레콤(주)의 일부 소프트웨어 개발자 및 프로젝트 관리자들이 직면한 과제에 대응하기 위해 필요에 의해 작성되었다. 물론 회사 자체는 직원들이 라이선스에 부합되게 행동하길 원하지만 유감스럽게도 회사 내에서 사용하는 일부 오픈소스 라이선스에 부합하기 위해 정확히 해야 할 일을 간단히 열거한 참조문을 찾을 수 없었다.

텔레콤 프로젝트의 동료들 및 심지어 OSLiC의 초기 저자인 우리들의 경우에도 오픈소스 소프트웨어를 사용하고 각각의 라이선스를 준수하기 위해서 오픈소스 라이선스 전문가가 되기를 원하지 않았다. 또한 우리는 변호사도 되고 싶지 않았다. 단지 그 당시에 대부분의 시간과 자원이 요구되었던 일을 보다 효과적으로 수행하기를 원했다. 우리는 오픈소스 라이선스의 정글 속에서 올바른 방향을 결정하는 것 대신에 프로젝트별로 그리고 반복적으로 사용할 수 있는 명확한 지침을 찾는 중이었다. 우리는 성능을 향상시키는 고품질의 오픈소스를 사용하는 것을 매우 선호했다. 그리고 우리는 이를 합법적으로 사용하기를 원했다. 하지만 우리는 많은 다른 오픈소스 라이선스의 법적 제약에 대해 힘들게 논의하는 것은 원하지 않았다.

우리가 필요로 하는 것은 하나의 사용하기 쉬우면서 직접적으로 실행 가능한 목록을 나열하는 문건이었다. 우리만의 쓰임새와 라이선스에 맞는 과업목록(to-do list)을 얻고 싶었다. 우리는 오픈소스 라이선스를 준수했음을 확신하기 위한 신뢰할 만한 작업 목록이 필요했다. 우리가 처음 시작했을 때는 그런 개론은 존재하지 않았다.

이 문제를 해결하기 위해 우리 회사는 세 가지를 결정을 내렸다:

---

2) 반면에 오픈소스를 다루는 글들이 무수히 많다. 이러한 책들과 기사들을 통해 지식을 넓히려고 시도하는 것은 그 글 속에 파묻혀 의미를 잃어버릴 가능성이 있다 : 우리의 2차 문헌 목록은 이러한 '압도당할 위험'을 개략적으로 알려주고 있다. 하지만 그럼에도 불구하고 OSLiC의 마지막에 있는 도서목록은 전부가 될 수 없다. 심지어 그것을 완성하려는 의도도 없다. 그것은 단지 배경 지식을 명확히 표시하기 위한 용도이며, OSLiC에서 직접 인용하지 않는다. 만약 우리가 오픈소스(라이선스) 주제에 대해 좋은 개요를 얻기 위해 두 권의 책을 제시해야 한다면 우리는 다음과 같이 표기할 것이다. (a) the 'Rebel Code' (for a Germanversion cf. Moody, Glyn: Die Software-Rebellen. Die Erfolgsstory von Linus Torvalds undLinux; transl. from the American [edition, 2000] by Annemarie Pumperling; Landsberg am Lech: verlag moderne industrie, 2001, ISBN 3-478-38730-2, passim—for an Englishversioncf. Moody, Glyn: RebelCode:LinuxAndTheOpenSourceRevolution:[NewYork]:BasicBooks, 2002, ISBN978-0738206707.p.passim) 그리고(b)the 'legalbasicconditions'(cf.Jaeger, Tilla.AxelMetzger:OpenSourceSoftware.Rechtliche RahmenbedingungenderFreienSoftware:3rdedition. Muñchen: VerlagC.H.Beck, 2011,passim). 하지만 우리는 다행이도 그렇게 하도록 강요당하지 않았다.(배경지식 이외에는 의존하지 않았다)

첫 번째 결정은 오픈소스 라이선스 전문가 위원회로 활동하도록 소규모의 직원들을 지원하는 것이었다. 그들은 회사 전체를 위한 서비스를 제공해야 한다. 프로젝트들, 매니저들, 그리고 개발자들은 그들이 특정 환경에 처해 있을 때 오픈소스 라이선스를 준수하기 위해 해야 할 일들에 대해 이 위원회에 질문을 할 수 있어야 한다. 그리고 이 위원회는 요청자가 해당 오픈소스 라이선스에 따라 행동하고 있음을 확신할 수 있는 신뢰성 있는 과업목록을 제공해야 한다. 이 결정의 배경은 간단했다. 모든 새로운 프로젝트에 대해 반복적으로 개발자를 선택(및 교육)해야 하는 대신 하나의 중앙 전문가 그룹이 있는 경우 비용을 절감하고 품질을 향상시킬 수 있다는 것이다. 그래서 OSRB(the Telekom Open Source Review Board, 텔레콤 오픈소스 검토 위원회)는 자기 구성적이고 상향식으로 운영되는 커뮤니티로서 초기의 전문가 그룹으로 설립되었다.

우리 회사가 취한 두 번째 결정은 텔레콤 오픈소스 검토 위원회가 체계적으로 위원회의 결과물을 재 사용 가능한 해설서의 형태로 수집하게 하는 것이다. 이 결정의 개념 또한 간단했다. 내부 서비스가 알 려질수록 작업량은 증가할 것이며 작업이 많을수록 리소스가 많아지고 비용이 높아진다. 따라서 본 해설서는 비용을 절감하고 요청자들이 라이선스 전문가가 되지 않고서도 스스로 그 답을 찾을 수 있게 만드는 것이다. 모든 기본 사례에서 그들의 작업에 대해 OSRB에서 분석한 결과를 요청하는 것 대신에 스스로 해설서에서 답을 찾아야만 했다. 그렇게 함으로서 기획된 텔레콤 오픈소스 해설서는 향후 위원회가 방대해 지는 것을 예방할 것이다.

우리 회사가 취한 세 번째 결정은 텔레콤 오픈소스 검토 위원회가 오픈소스 프로젝트를 일반적으로 사용하는 협력 방식으로 해설서를 작성하도록 하는 것이었다. 이 결정의 이유는 간단했다. 대체적으로 추후에 오픈소스 검토 위원회에 의해 검토되는 것이 아니라 간단한 매뉴얼을 통해 프로젝트, 프로그래머 및 관리자들의 오픈소스 라이선스 준수를 보장해야 한다면 이 해설서는 특히 더 신뢰성이 있어야 한다. 오픈소스 작업 모델에는 협력 공동체의 지속적인 검토가 품질을 향상시킨다는 이미 잘 알려진 특징이 있다. 따라서 내부 “텔레콤(Telekom) 유인물”을 작성하는 것 뿐 만 아니라 전체 커뮤니티가 더 광범위한 오픈소스 라이선스 해설을 사용, 수정 및 재배포 할 수 있도록 하는 결정은 품질 향상을 위한 것이었다. 결과적으로 오픈소스 검토 위원회는 OSLiC를 일련의 LATEXsources로 발표하기로 결정했으며 공개 저장소인 GitHub를 통해 출간하기로 했다.<sup>3)</sup> 그리고 OSLiC는 크리에이티브 커먼즈 저작자표시-동일조건변경허락 3.0 독일 라이선스를 적용하기로 했다.<sup>4)</sup>

하지만 OSLiC를 무료 책으로 출간하는 것은(적어도 텔레콤 오픈소스 검토 위원회에 있어서는) 수 년 동안 많은 직원과 회사의 삶을 풍요롭게 하고 단순화시킨 오픈소스 커뮤니티에 감사의 뜻을 전하는 또 다른 중요한 의미가 내포되어 있다.

3) 다음 링크를 통해 코드를 받을 수 있으며 <https://github.com/dtag-dbu/oslic>; 프로젝트 정보는 <http://dtag-dbu.github.com/oslic/> 또는 <http://www.oslic.org/>에서 찾을 수 있다.

4) 본 내용은 크리에이티브 커먼즈 저작자 표시-동일조건변경허락(Creative Commons Attribution-ShareAlike) 3.0 독일라이선스에 의해 사용이 허가된다. “본 문서와 동일하거나 유사한 라이선스에 의한 저작물을 배포”하고 “저자가 지정한 방식의 저작물로 중시” 한다면 제약 없이 “공유(복제, 배포 및 전송)” 또는 “수정(개작)”을 할 수 있다. 인터넷에서 재사용 시 적절한 방식으로 최초 저자들을 언급하고, 이들의 스폰서인 도이치텔레콤(주)의 명칭을 제시하며 <http://www.telekom.com> 링크를 적용하기 바란다. 문서 유형으로 재사용 시 문서에 <http://www.telekom.com>, 최초 저자 그리고 스폰서인 도이치텔레콤(주)에 대해 간략하게 소개하기 바란다. 일반적인 인용문의 경우에는 과학 표준 방식을 사용한다.

전체적으로 OSLiC 는 5가지의 원칙을 따른다.

**과업목록(To-do lists)을 중심으로 논의:** 오픈소스 소프트웨어 및 라이선스를 구체적으로 사용하는 법에 대한 정보 수집을 위해 간단한 양식에 기초하여 OSLiC에서는 각각의 과업목록(To-do lists)을 제공한다. 그리고, 요청자들이 라이선스 준수를 할 수 있도록 인도하는 사용이 용이한 파인더를 제공하도록 한다. 또한, OSLiC의 이러한 모든 요소는 사용법 자체를 방해하지 않고 이해하기 쉽도록 소개되고 논의되어야 한다.

**철저하게 명확한 출처를 포함하는 인용문:** OSLiC는 개정 가능하고 신뢰할 수 있어야 한다. 그것은 누가 왜 어떤 정보를 누구로부터, 어떤 버전에서 그리고 왜 사용하는지를 총체적으로 언급하고 명시해야 한다.<sup>5)</sup>

**숲을 개간하지 않지만 풀을 베어 길을 내기:** OSLiC는 라이선스와 그것의 법적인 특징에 대해 다루어야만 한다. 하지만 이는 모든 세부적인 측면을 다룬다는 것은 아니다. 다른 대안<sup>6)</sup>이 있을 수 있더라도 특정 사용 사례에서의 라이선스에 따라 가능한 이행 방법에 초점을 두어야 한다.

**라이선스 문구의 내용을 신중히 고려하기!:** OSLiC에서는 법적 논의에 관한 전반적인 설명을 제공하지 않아야 하며 여기에 관여하지 않아야 한다. 여기에서는 라이선스를 준수하기 위해 각각의 라이선스와 사용사례를 위한 신뢰할 수 있는 한 가지 방법만을 찾는 것이어야 한다. 이런 분석의 주요 출처에서는 오픈소스 라이선스를 정확히 이해하고 있어야 한다. (자선 변호사 및 합리적인 주장을 펼치는 소프트웨어 개발자의 해석에 의해 뒷받침되거나 그에 근거하는). OSLiC에서는 소프트웨어 개발자를 대상으로 때로는 개발자에 의해 오픈소스 라이선스가 작성된다는 점을 중요시한다.

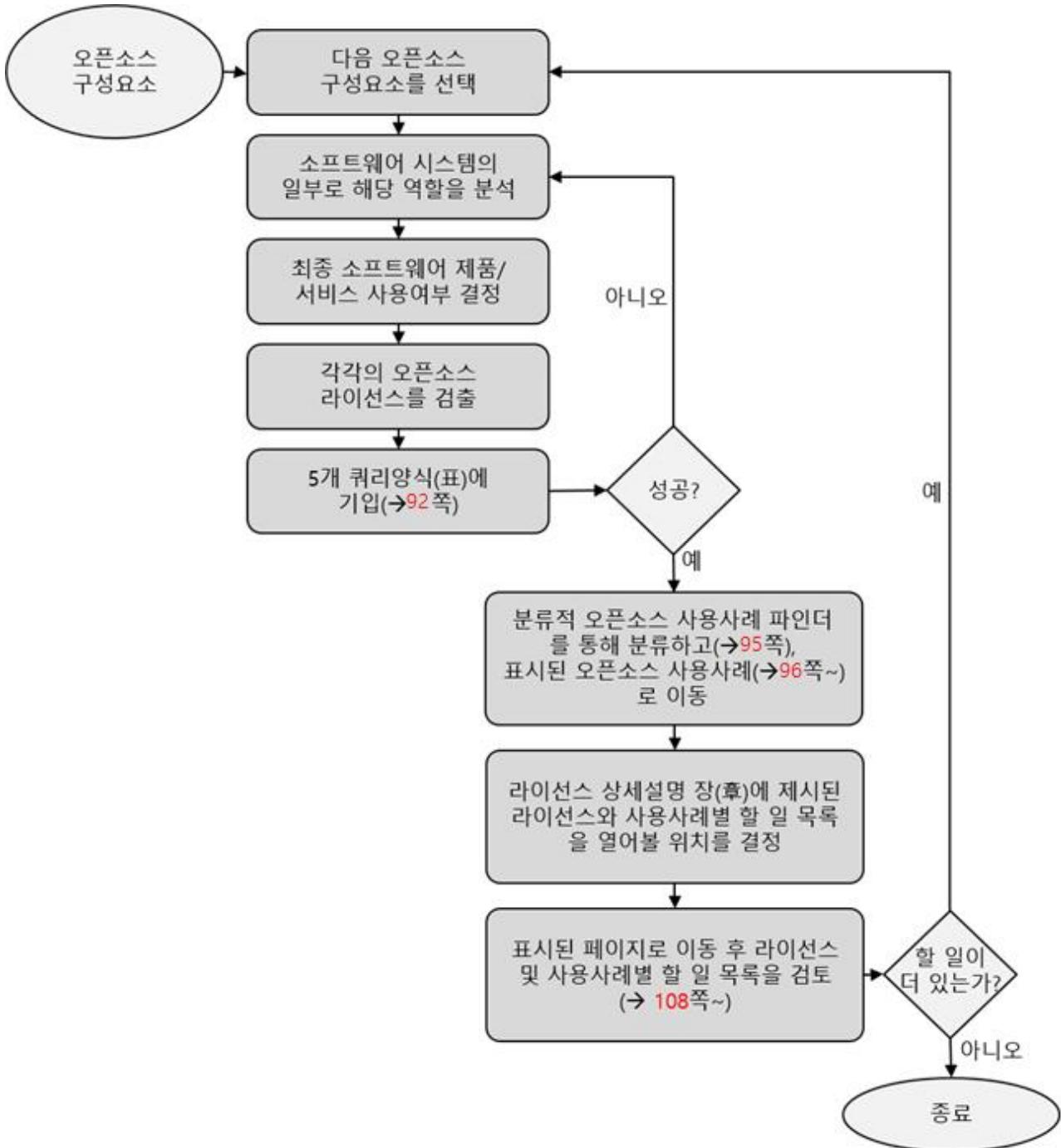
**군중에 대한 신뢰!:** OSLiC는 도이치텔레콤(주)의 직원이 아닌 다른 사람들에 의해 장려되고 고무되는 개선 및 조정을 위해 개방되어 있어야 한다.

이런 원칙에 입각하여 OSLiC를 사용하는 두 가지 방법을 제공한다.

5) 이를 위해 우리는 주석이나 힌트를 내포하는 대신에 ‘구식’문헌 스타일의 각주를 사용한다. 우리는 사용자가 원하는 대로 댓글과 힌트를 검토 또는 무시할 수 있도록 하고 싶지만 대규모의 내포된 댓글이나 짧은 페이지 전환으로 인한 방해를 받지 않고 모든 의견과 힌트를 검토하거나 무시할 수 있게 했다. 현대 작가 가이드에서는 덜 요란한 스타일을 선호한다는 점은 알고 있다.(pars pro toto cf. MLA: *MLA Handbook for Writers of Research Papers*; 7th edition. New York: The Modern Language Association of America, 2009, ISBN978-1-60329-024-1, passim)하지만 인터넷 출처의 짧은 변경 때문에 제기되는 신뢰성 있는 활용을 원할 경우 이 방법은 여전히 다소 부정확하다 (세부사항→ OSLiC, pp.308. OSLiC 인용 스타일에 관한 몇 가지 추가 설명에서 이 방법을 쓴 동기 참조) cf. Reincke, Karsten: *Classical Scholar Texts WithFootnotes based on LaTeX, BibTeX, Koma, jurabib and mykeds-CSR*: 2012 (URL: <http://www.fodina.de/en/closedprojects/latex-addons/classical-scholar.html>) - reference download: 2013-02-10, passim. For a more elaborated legitimizing version cf. Reincke, Karsten: (Geistes-) Wissenschaftliche Texte mit jurabib. Dienst am Leser, Dienst am Scholaren: Über Anmerkungsapparate in Fußnoten - aber richtig. [n.l.], 2012(URL: <http://download.fodina.de/fodina/ClassicalScholarFoNoDe.pdf>) - reference download: 2013-02-10, passim).

6) OSLiC는 특정 필요성에 관련하여 프로젝트에 대해 조언하지 않는다. 이는 변호사와 법률 전문가의 업무로 남아야 한다. OSLiC는 변호사의 법적 조언 또는 법적 검토를 대신할 수 없고 대체되지 않는다. 이 내용은 개발자, 관리자, 오픈소스 전문가 및 회사가 최종적으로 법률 고문이 검토해야 하는 해결법을 찾는데 기여하도록 한다. 독일 독자들에게 다시 반복하지만 OSLiC는 물론 독일의 ‘비사법적 법률 서비스에 관한 법률(Rechtsdienstleistungsgesetz)’을 존중한다. 단지 이 문서는 일반 대중에게 언급하는 법적 의견을 담고 있을 뿐이다. 다만 법적 문제에 대한 표현과 토론에 대한 단순한 보편적 접근을 지향하는 선에서 이해되기를 바란다.(“nur an die Allgemeinheit gerichtete Darstellung und Erörterung von Rechtsfragen”).

다른 무엇보다도 독자들은 자신들이 꼭 필요로 하는 과업 목록을 쉽고 빠르게 찾기를 기대한다. 다음 그림이 그 절차이다.<sup>7)</sup>



7) 잘 알려진 ‘빠르고 야비한 해커’에 관해서는 (우리 또한 그런 경향이 있지만) 여기서 편법(손쉬운 방법)을 포함시켰다. 사용하고자 하는 오픈소스 패키지 라이선스를 이미 알고 여기서 정의한 오픈소스 사용 사례의 의미에 매우 정통하다면 5.1장의 OSLiC 5가지 질의 양식, 분류적 오픈소스 사용 사례 파인더 또는 오픈소스 사용 사례로 고심하지 않고 6장에 있는 해당 라이선스에 대한 설명으로 바로 건너뛸 수 있다. 상세하게 오픈소스 라이선스만을 다른 일부 장은 일련의 라이선스별 사용사례를 제공하는 라이선스 특정 탐색기로부터 시작한다. 단, 라이선스의 복잡성에 따라 일부의 경우 골자만 남길 수는 있다. 이 방법의 단점은 OSLiC가 체계적으로 안내하는 과정 없이 스스로 사용사례에 관한 지식을 적용해야 하며 부작용이 있을 수 있다는 것이다.

두 번째, 독자들은 전체 분석 내용을 이해하고 싶어 할지도 모른다. 따라서 여기에서는 라이선스를 준수하는 행동의 기초로서 오픈소스 라이선스 분류체계에 대해 간략히 논의한다.<sup>8)</sup> 그리고, 오픈소스 라이선스에 따르는 일부 부작용을 고려한다.<sup>9)</sup> 최종적으로 오픈소스 사용 사례 구조에 대해 학습한다.<sup>10)</sup>

많은 사람의 많은 소망인 '행복한 (합법적인) 해킹!'이 담고 있는 사용, 변경, (재)배포를 강조하면서 이 서론을 마치고자 한다.

---

8) → OSLIC “2. 오픈소스: 동일한 관념, 여러 다른 라이선스”, 17쪽

9) → OSLiC “3. 오픈소스: 일부 부작용에 대하여”, 43쪽

10) → OSLiC “4. 오픈소스 사용 사례: 개념과 분류체계”, 88쪽

## 2 오픈소스: 동일한 관념, 다양한 라이선스

이 장에서는 자유 오픈소스 소프트웨어에 관해서 같은 목적을 갖는 다양한 라이선스 모델을 설명하고자 한다. 먼저 우리는 라이선스들을 그룹화 하는 기준 방법들의 한계에 관해 논하려 한다. 기준의 그룹은 실제 라이선스 사례에 따라 행동하기 위한 간단한 조건들을 제공하는 '가상의 프로토타입 라이선스'라는 의미로 종종 사용되었다. 하지만, 우리가 지켜야하는 것은 이런 그룹들의 일반적인 조건들이 아닌 각 라이선스들의 세부조건들이다. 그럼에도 불구하고 우리는 오픈소스 라이선스들을 구조화하는 새로운 방식을 제시하려한다. 각 라이선스들은 어떠한 대상(사람 또는 단체)으로부터 다른 대상을 지키기 위한 목적들을 갖게 되는데 우리는 이 목적들을 기준으로 공통된 그룹을 나눌 것이다. 이러한 패턴을 통해서 우리는 실제로 모든 오픈소스 라이선스들을 그룹화 할 수 있다.

오픈소스 라이선스 그룹화<sup>11)</sup>는 일반적으로 진행되었다. 일련의 오픈소스 라이선스<sup>12)</sup>는 그 자체로 이미 여러 그룹화 기준에 의해 정해진 한 무리이다. 오픈소스 라이선스가 되기 원하는 각 소프트웨어 라이선스들"[...]는 반드시 [...] 조건을 따라야함"은 오픈소스 정의(Open Source Definition, OSD)<sup>13)</sup>에 포함되는 "배포조건"을 가지게 된다. 각 라이선스들은 Open Source Initiative(Open Source Initiative)<sup>14)</sup>에 의해 관리가 되는 OSD 기준으로 분류가 된다. 따라서 OSD는 "비 오픈소스 라이선스 그룹"<sup>15)</sup>과 "잠재적인 오픈소스 라이선스 그룹"을 구분하게 된다.

자유 소프트웨어 라이선스(Free Software License)를 그룹화하기 위한 또 다른 방법은 "자유 소프트웨어 정의(Free Software Definition, FSD)"에 의해 지정된다. 이 자유 소프트웨어 정의는 자유 소프트웨어 라이선스에 의해 충족되어야 하는 4가지 조건이 포함된다. 자유 소프트웨어 정의에 준하는 라이선스에 의해 "어떠한 목적을 위해 프로그램을 실행하는 자유", "작동 방식을 학습하고 필요에 맞게 조정하는 자유", "복제물을 재배포하는 자유", 그리고 마지막으로 "프로그램을 개선하고 개선사항을 공개하는 자유"<sup>16)</sup>가 허용되어야 한다. 의외로 이 정의는 소스 코드는 공개적으로 이용이 가능해야 한다는 요구사항이 '유일하게' 도출되는 조건임을 내포하고 있다. "개선된 버전을 공표하는 자유와 변경할 자유"가 "의미가 있어야" 한다면 "프로그램의 소스코드 이용"은 전제 조건이 된다. "따라서 소스코드 이용 가능성은 자유 소프트웨어의 필수 조건이 된다."<sup>17)</sup>

11) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스; 2012 [n.y.]

(URL:<http://opensource.org/licenses/alphabetical>) - 참고자료 다운로드: 2013-01-22, wp. 참조

12) Open Source Initiative: Open Source Initiative; 2012 [n.y.] (URL: <http://www.opensource.org/about/>) - 참고자료 다운로드: 2013-01-22, wp. 참조

13) Open Source Initiative: 오픈소스 정의; 2012 [n.y.] (URL: <http://www.opensource.org/docs/osd>) - 참고자료 다운로드: 2012-06-21, wp. 참조

14) 더 정확하게는 : 오픈소스 소프트웨어 정의를 충족하는 것만이 오픈소스 라이선스가 되기 위한 필수 조건이다. 오픈소스 라이선스가 되기 위한 충분한 조건은 공식 오픈소스 라이선스 승인 과정을 제공하는 OSI로부터 승인을 받는 것이다(Open Source Initiative: [OSI] 라이선스 검토 프로세스; 2012 [n.y.]

(URL: <http://www.opensource.org/approval>) - 참고자료 다운로드: 2013-01-22, wp. 참조).

15) 조금 더 명확하게 하자면, 오픈소스 라이선스가 되기 위한 필요조건은 OSD를 만족시키는 것이다. 오픈소스 라이선스가 되기 위한 충분조건은 OSI가 제공하는 공식 절차를 통과한 오픈소스 라이선스 승인이다. (cf. Open Source Initiative: The [OSI] License Review Process; 2012 [n.y.] <URL:

<http://www.opensource.org/approval>> - reference download: 2013-01-22, wp).

16) 리처드 스톤만: 자유 소프트웨어 정의, 1996, 41쪽 참조

17) 참고문헌 참조

오픈소스 소프트웨어 정의와 자유 소프트웨어 정의간의 차이는 종종 강조의 차이로 설명되어왔다.<sup>18)</sup> 두 가지 정의는 “거의 정확히 동일한 범위의 소프트웨어를 다루지만” 자유 소프트웨어 재단(Free Software Foundation)은 “자유 사항을 강조하길 선호하고” Open Source Initiative(OSI)는 철학적으로 무관한 “개발 방법론”을 강조하길 원한다.<sup>19)</sup>

자유 소프트웨어 라이선스와 자유 소프트웨어 그룹의 세 번째 방법은 “데비안 사회 계약(Debian Social Contract)”에 포함된 “데비안 자유 소프트웨어 가이드라인”에 의해 지정되었다. 데비안 자유 소프트웨어 가이드라인은 데비안 자체로 말해 주듯이 9가지를 정의하는 기준을 포함하며 “오픈소스 정의 기준으로서 자유(문자 그대로!) 소프트웨어 커뮤니티에서 채택”해 왔다.<sup>20)</sup>

이 방법을 대략적으로 이해하면 이 3가지 정의가 외연적으로 동일하고 내면적으로만 다르다는 결론에도 달할 수 있다. 그러나 사실이 아니다. 차이를 밝히기 위해 여러 라이선스 군(群)의 차이를 정할 수 있는 것이 무엇인지에 대한 질문을 통해 Open Source Initiative 승인 라이선스, 오픈소스 소프트웨어 정의에 준하는 라이선스, 데비안 자유 소프트웨어 가이드라인에 준하는 라이선스 및 자유 소프트웨어 정의에 준하는 라이선스 군(群)을 외연적으로 비교하였다.<sup>21)</sup>

첫째, 가장 결정하기 쉬운 차이는 단방향으로 포함되는 것이다. 정의에 의해 OSI(Open Source Initiative 승인) 라이선스와 OSD(오픈소스 소프트웨어 정의)에 준한 라이선스는 OSD(오픈소스 소프트웨어 정의) 요구사항에 부합한다.<sup>22)</sup> 그러나 OSI(Open Source Initiative) 승인 라이선스만이 OSI(Open Source Initiative)의 프로세스를 성공적으로 통과하였고<sup>23)</sup> 이에 따라 정식으로 오픈소스 라이선스로 실리게 되었다.<sup>24)</sup> 한편으로는 OSI(Open Source Initiative) 승인 라이선스는 오픈소스 라이선스이며 그 반대의 경우도 마찬가지이다. 다른 한편으로는 OSI(Open Source Initiative) 승인 라이선스와 오픈소스 라이선스는 OSD(오픈소스 소프트웨어 정의)에 준한 라이선스이지만 그 반대의 경우는 아니다.

둘째, 유사한 논증을 통해 DFSG(데비안 자유 소프트웨어 가이드라인)에 준한 라이선스와 OSI(Open Source Initiative) 승인 라이선스를 구별할 수 있다. 언급한대로 OSD(오픈소스 소프트웨어 정의)는 “데비안 자유 소프트웨어 가이드라인에 기반을 두고 있고 하나의 정의에 부합하는 어떤 라이선스는 거의 다

18) 이는 또한 리처드 스톤만(Richard M. Stallman)의 관점이다. 한편으로 그는 “자유 소프트웨어 운동”과 “오픈소스 운동”은 일반적으로 “기본 원칙은 차이가 있지만 실제 권장사항은 다소 일치한다.” 그리고 “오픈소스 운동을 적으로 생각하지 않는다.”고 명시하였다. 다른 한편으로는 오픈소스 운동의 경우, 소프트웨어가 오픈소스가 되어야 하는 이슈는 실제적 문제이지 윤리적 문제가 아니며”, “자유 소프트웨어 운동의 경우 비자유 소프트웨어는 사회 문제이고 자유 소프트웨어가 해결책”이라고 언급하여 두 가지 운동에 대해 기술하였다.(‘자유 소프트웨어 자유사회’에 실린 리처드 스톤만의 ‘자유 소프트웨어가 ‘오픈 소프트웨어’보다 나은 이유(Why ‘Free Software’ is Better than ‘Open Software’), 1998년 최초 작성, 55쪽 참조). 스톤만은 간단한 방법으로 견해를 간략히 제시하였다. “‘오픈소스’는 사용자들이 자유를 누려야 한다는 의견을 제기하려는 목적으로 고안되지 않았다.” 그러나 친구들은 “자유에 대한 생각을 퍼뜨리고” 이에 따라 “자유 소프트웨어”를 고수하길” 원한다(상기 인용 글과 동일, 59쪽). 이 의견을 다시 짚어 보면 “오픈소스는 개발 방법론이고 자유 소프트웨어는 ”윤리적 의무“에 근간하는 사회 운동이다.”라고 표현하고 있다. Communications of the ACM 6월호에 실린 리처드 스톤만의 “관점 : “오픈소스”에는 자유소프트웨어 관점이 생략된 이유(Viewpoint: Why “Open Source” Misses the Point of Free Software”), 2009년 6월 참조. (URL: <http://doi.acm.org/10.1145/1516046.1516058>) - 참고자료 다운로드: 2011-12-29, 31쪽

19) Debian(데비안): 데비안 자유 소프트웨어 가이드라인(DFSG): 2013 [n.y.] (URL: [http://www.debian.org/social\\_contract#guidelines](http://www.debian.org/social_contract#guidelines)) 참조 - 참고자료 다운로드: 2013-01-22, p.wp.

20) Debian(데비안): 데비안 자유 소프트웨어 가이드라인(DFSG): 2013 [n.y.] (URL: [http://www.debian.org/social\\_contract#guidelines](http://www.debian.org/social_contract#guidelines)) 참조 - 참고자료 다운로드: 2013-01-22, p.wp.

21) 실제로 존재하는 라이선스의 부분집합은 확대될 수 있고 정의에 맞는 신규 라이선스로 개발될 수 있으므로 정의에 대한 외연적 기능을 분석하기 위해 가능한 표준에 준한 모든 라이선스를 고려해야 한다.

22) Open Source Initiative: 오픈소스 정의, 2012, wp. 참조

23) 참고문헌 참조

24) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

른 정의에 부합한다.”<sup>25)</sup> 그러나 정의에 부합한다는 것은 정식 오픈소스 라이선스가 되기에 불충분하다. 라이선스는 OSI(Open Source Initiative)로부터 승인을 받아야 한다.<sup>26)</sup> 따라서 Open Source Initiative 승인 라이선스 또한 데비안 자유 소프트웨어 가이드라인에 준한 라이선스가 되지만 반대의 경우는 아니라 결론이 나온다.

셋째, “수년간”<sup>27)</sup> 보여 왔던 “몇 가지 예외사항”을 무시함으로서 그들의 ‘친척같은(kinsmanlike)’ 관계 때문에 적어도 오픈소스 소프트웨어 정의에 준하는 라이선스는 데비안 자유 소프트웨어 가이드라인에 준하는 라이선스이며 그 반대의 경우도 마찬가지이다.

마지막으로 (잠재적) 자유 소프트웨어 라이선스 집합은 다른 모든 3 개집합 보다 커야 한다. 한편으로는 FSD(자유 소프트웨어 정의)에 따르면 자유 소프트웨어 라이선스는 소프트웨어를 판독할 수 있는 것뿐만 아니라 사용, 변경 및 배포가 허용되어야 한다.<sup>28)</sup> 이 조건은 “자유로운 재배포”, “소스코드”, “파생저작물 (Derived Work)”이라는 주제와 관련한 처음 3 개의 OSD(오픈소스 소프트웨어 정의) 단락에 포함되어 있다.<sup>29)</sup> 한편, OSD(오픈소스 소프트웨어 정의)는 FSD(자유 소프트웨어 정의)에 의해 언급되지 않는 요구사항과 OSD(오픈소스 소프트웨어 정의)에 준한 라이선스로서의 자격이 되기 위해 라이선스에 의해 충족되어야 하는 요구사항을 몇 가지 더 포함한다.<sup>30)</sup> 모든 FSD(자유 소프트웨어 정의) 조건은 충족하지만 OSD(오픈소스 소프트웨어 정의)에 있는 몇 가지 요구사항을 따르지 않는 라이선스가 존재할 수 있다는 결론이 나온다.<sup>31)</sup> 따라서 모든(잠재적) 자유 소프트웨어 라이선스 세트가 모든(잠재적) 오픈소스 라이선스보다 크고 OSD(오픈소스 소프트웨어 정의)에 준한 라이선스보다도 커야 한다.

이 상황을 아래 그림과 같이 나타낼 수 있다.

---

25) 포겔 : 오픈소스 소프트웨어 작성, 2006, 233쪽 참조

26) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

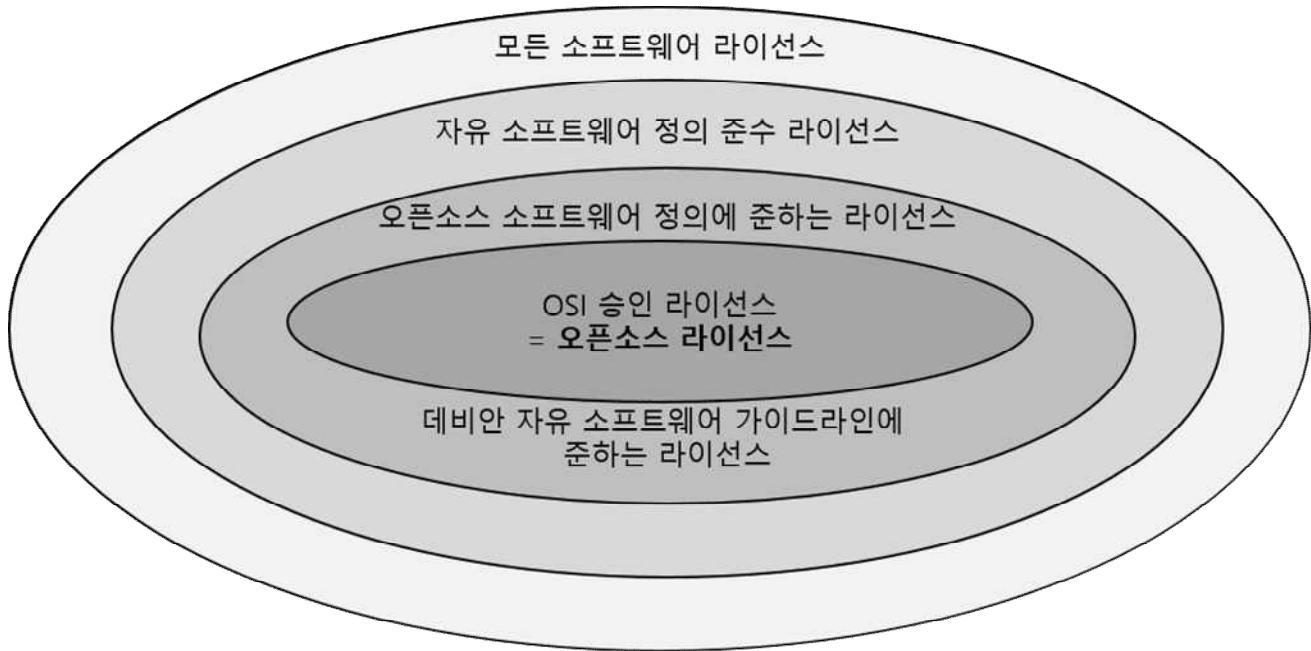
27) 포겔 : 오픈소스 소프트웨어 작성, 2006, 233쪽 참조

28) 스톤만 : 자유 소프트웨어 정의, 1996, 41쪽 참조

29) Open Source Initiative: 오픈소스 정의, 2012, wp. 참조

30) 예를 들어, “라이선스는 기술 중립적이어야 한다.”라는 조건을 참고한다 (위 책과 동일).

31) 다시 언급하자면 실제 기존의 라이선스 집합이 아닌 외연적 정의 가능성을 고려해야 한다. 여기서, GPL, LGPL 또는 AGPL과 같은 실제 기존의 모든 자유 소프트웨어를 오픈소스 라이선스로 분류하는 것은 부적절하다. 여기서는 OSD(오픈소스 소프트웨어 정의)가 아닌 FSD(자유 소프트웨어 정의)를 이행하는 라이선스를 야기할 수 있다는 사실을 언급하고 있다.



위 그림을 보면 오픈소스 라이선스를 분류하기 위해 더 추가할 군(群)이 없을 것이라는 것을 더 길게 설명하지 않아도 될 것이다. 한편, 가장 큰 군(群)에서는 오픈소스 라이선스에 관하여 나타내고 있지 않는다. 반면에 오픈소스 라이선스 군에서는 라이선스를 충족하기 위해 해야 할 행위가 구체적으로 명시되지 않은 단지 오픈소스 정의를 기준으로 한 요소들만 모았다.

다음 레벨의 오픈소스 라이선스 군(群)은 OSI(Open Source Initiative) 승인 라이선스의 내부 구조와 관계가 있다. 심지어, OSI(Open Source Initiative) 자체적으로 라이선스 목록을 놓고서 OSI 사이트 방문자들의 요구에 더 부합하는 보다 나은 유형의 그룹화에 대해 최근에 논의하였다.<sup>32)</sup> OSI(Open Source Initiative)는 최종적으로 “대중적이고 널리 사용되거나 영향력이 큰 커뮤니티에서 사용되는 라이선스”, “특수 목적의 라이선스”, “기타 라이선스”, “더 대중적인 라이선스들이 많이 포함된 라이선스”, “재사용 불가 라이선스”, “대체된 라이선스”, “임의로 폐기된 라이선스”, “비분류 라이선스”의 범주로 분류하였다.<sup>33)</sup>

오픈소스 라이선스 유형을 구성하는 또 다른 방법은 “학술 기관이 원래 지었던 명칭으로 이름 지은” 학술적 라이선스(*academic licenses*)<sup>34)</sup>, “파생저작물의 소스코드를 공개해야 하는 요구사항을 포함하여 동일한 라이선스에 의거 저작물을 파생저작물을 배포자가 공개해야”하는 특징으로 이름 지은 상호적 라이선스(*reciprocal license*)<sup>35)</sup>, ‘산업표준’의 재사용 가능성을 나타내기 때문에 이름 지은 표준 라이선스

32) Open Source Initiative: OSL 메일링 리스트. 라이선스-논의. 신규 Open Source Initiative 라이선스 시작 페이지 초안; 2012 [n.y.] (URL: <http://projects.opensource.org/pipermail/license-discuss/2012-April/000332.html>) - 참고 자료 다운로드: 2013-01-29, wp. 참조

33) Open Source Initiative: 카테고리별 오픈소스 라이선스; 2013 [n.y.] (URL: <http://opensource.org/licenses/category>) - 참고자료 다운로드: 2013-01-29, wp. 참조

34) Rosen, Lawrence: Open Source Licensing. Software Freedom and Intellectual Property Law; Upper Saddle River, New Jersey: Prentice Hall PTR, 2005, ISBN 0-13-148787-6, p. 69. 참조

35) id., l.c., p. 70. 참조

(*standard license*<sup>36)</sup>, 그리고 음악, 예술, 영화, 문학 작품 등 소프트웨어 이외의 것을 나타내기 위해 이를 지은 컨텐츠 라이선스(*content license*<sup>37)</sup>등으로 분야를 나누어 '오픈소스 라이선스 유형'을 고안한 것이다.

두 가지 유형의 분류체계는 새로운 (소프트웨어)프로젝트에 사용하고자 하는 적합한 라이선스를 찾는데 직접적인 도움을 준다. 그러나 이러한 분류를 통해 라이선스 준수 활동을 추론할 수 없다. 이 유형들은 대부분 라이선스 외부 기준만을 고려하여 정의되기 때문이다. 특정 기관에서 제작한 라이선스든 또는 산업 표준이나 소프트웨어가 아닌 다른 유형의 저작물을 다루는 라이선스든 상관없이 본질적으로 라이선스 준수 활동을 제공하지 못한다.

학술적 라이선스와 상호적 라이선스로 그룹화 하는 행위만이 라이선스 이행 업무의 개념을 고려하고 있다. 단, 이들 라이선스들이 "파생저작물의 소스코드를 배포하기 위해 적용되는 라이선스에 따른 의무사항 없이 어떠한 목적으로든 소프트웨어를 사용할 수 있도록 허가" 한다는 세부조항에 따라 "학술적 라이선스의 정의를 확대 하였을 경우"에 그렇다.<sup>38)</sup> 이 추가 세부조항과 관련하여 "학술적 라이선스 (*academic licenses*)와 상호적 라이선스(*reciprocal license*)군은 오픈소스 라이선스의 "주요 분류"이다.<sup>39)</sup> 정의에 의하면 이 라이선스들은 반대가 될 뿐 아니라 반대되는 모순을 구성한다. 그러나 오픈소스 라이선스 집합 내부에 본질적 대립과 이율배반을 형성하고 있다는 점을 명심해야한다<sup>40)</sup>.

학술적 라이선스(*academic licenses*)와 상호적 라이선스(*reciprocal license*)를 본질적 유사성에 따라 묶는 방법은 허용적 라이선스(*permissive license*)와 약한 카피레프트 라이선스(*weak copyleft license*), 그리고 강력한 카피레프트 라이선스(*strong copyleft license*)로 그룹화 하는 것이다. 심지어 위키피디아는 "그들을 "상호주의 및 동일 조건 변경허락"의 요구사항을 가진 "카피레프트 라이선스"와 대조되고 어떻게 소프트웨어가 재분배 되는지에 대한 최소한의 요구사항을 가지는 자유 소프트웨어 라이선스의 집단"이라는 의미로 "관대한(허용적) 자유 소프트웨어 라이선스(*permissive free software licence*)"라는 용어를 사용하였다.<sup>41)</sup>

몇몇의 다른 필자들은 내포된 "카피레프트" 효과의 연속으로서, 소스코드의 수정이 있을 시 반드시 공표해야하기 때문에, 학술적 라이선스(*academic licenses*)를 "허용적 라이선스"라고 명명하고, 상호적 라이선스(*reciprocal license*)를 "제한적 라이선스"라고 구분하였다. 그들은 파생저작물이 같은 라이선스 아래에서 공표되어야하는 "강력한 제한적 라이선스"의 일부분을 소개한다.<sup>42)</sup> 묶는 방법에 대한 다음 수정은

---

36) 참고문헌 참조

37) id., l.c., p. 71. 참조

38) 참고문헌 참조

39) id., l.c., p. 179. 참조

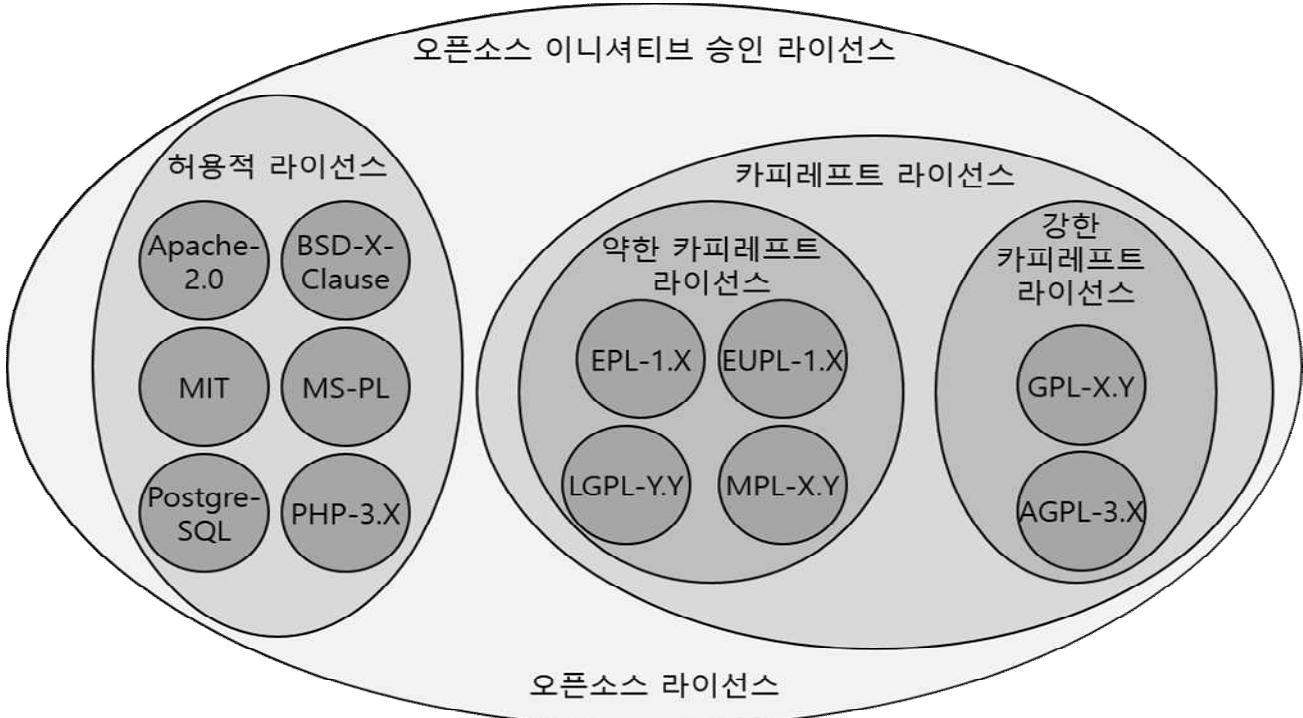
40) 이러한 이유로 "오픈소스 라이선스(OSL)는 상호적 라이선스"이고 "학술적 자유 라이선스(Academic Free License(AFL))는 상호주의 조항이 없는 정확한 동일한 라이선스"라고 한다면 약간의 혼란을 주게 된다. (id., l.c., p. 180 참조). 만약, BSD라이선스가 AFL이고 AFL이 OSL이 아니며 그리고 OSI가 OSL만 승인한다면 BSD라이선스는 승인된 오픈소스 라이선스가 될 수 없다. (Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조)

41) 위키피디아(영문판): 허용적 자유 소프트웨어 라이선스(*Permissive free software licence*): n.l., 2013 [n.y.] (URL: [http://en.wikipedia.org/wiki/Permissive\\_free\\_software\\_licence](http://en.wikipedia.org/wiki/Permissive_free_software_licence)) - 참고자료 다운로드: 2013-02-02, wp. 참조

42) pars pro toto cf. Buchtala, Rouven: Determinanten der Open Source Software-Lizenzwahl. Eine spieltheoretische Analyse: Frankfurt am Main, Berlin, Bern [... etc.]: Peter Lang, 2007 (=

“엄격한 카피레프트 조항에 따른 [오픈소스]라이선스”<sup>43)</sup>, “제한된 카피레프트 조항에 따른 [오픈소스] 라이선스”<sup>44)</sup>, 그리고 “어떠한 제약도 없는 카피레프트 [오픈소스]라이선스”<sup>45)</sup>로 분류하여 묶는 방법을 사용한다. 마지막으로 다음 관점은 약한 카피레프트(*weak copyleft*)와 강력한 카피레프트(*strong copyleft*)로 분류하는 방법과 연관된다. “약한 카피레프트가 적용된 소프트웨어는 변경될 때 카피레프트 조항을 준수해야 하고 그에 링크된 다른 소프트웨어는 변경에는 해당하지 않는다.” 반면에 “강력한 카피레프트는 이를 이용하여 만들어진 모든 종류의 파생저작물들 역시 카피레프트 조항의 영향을 받는다고 기술한다.”<sup>46)</sup>

적합한 분류와 명명에 대한 접근방식을 기반으로<sup>47)</sup> 다음과 같은 그림을 전개할 수 있다.



확장성에 기초한 오픈소스 라이선스 분류체계는 명확하게 알려져 있고 종종 혹은 덜 분명하게 거론되고

Informationsmanagement und strategische Unternehmensfu “hrung), [Vol./No.] 12), ISBN 978-3-631-57114-9, p. 57.

43) 최초에는 “Lizenzen mit einer strengen Copyleft-Klausel”로 명시됨. Jaeger a. Metzger : 오픈소스 소프트웨어.

Rechtliche Rahmenbedingungen der Freien Software, 2011, 24쪽 참조

44) 최초에는 “Lizenzen mit einer beschränkten Copyleft-Klausel”로 명시됨. 인용한 글과 동일, 71쪽 참조

45) 최초에는 “Lizenzen ohne Copyleft-Klausel”로 명시됨. 인용한 글과 동일, 83쪽 참조

46) 위키피디아(영문판): Copyleft; n.l., 2013 [n.y.] (URL: <http://en.wikipedia.org/wiki/Copyleft>) - 참고자료 다운로드: 2013-02-02, wp.참조

47) 마지막으로 다른 문맥에서 중요하게 생각되는 다른 분류방식도 존재한다. 예를 들어 ifross license는 주요 분류로 “Open Source Licenses” 와 그 아래 부문류로 “Licenses without Copyleft Effect,” “Licenses with Strong Copyleft,” “Licenses with Restricted Copyleft,” “Licenses with Restricted Choice,” 또는 “Licenses with Privileges” 포함한다. 그리고 이러한 분류를 OSI에 의해 목록화 된 것이 아닌 라이선스로 나타낸다. (ifross: License Center: 2011 [n.y.] < URL: [http://www.ifross.org/ifross\\_html/lizenzcenter-en.html](http://www.ifross.org/ifross_html/lizenzcenter-en.html) > - reference download: 2013-02- 26, wp 참조). 오픈소스 소프트웨어 정의 의미를 참고하면 이는 충분히 타당하다 (Open Source Initiative: The Open Source Definition, 2012, wp 참조). The OSLiC는 OSI(Open Source Initiative: The Open Source Licenses, alphabetically sorted, 2012, wp 참조)에 의해 목록화 된 향상된 오픈소스 라이선스를 참조하여 대상의 단순화를 원한다(cf. Open Source Initiative: The [OSI] Licence Review Process, 2012, wp 참조).

있다.<sup>48)</sup> 유감스럽게도 이 분류체계는 여전히 오해의 소지가 있는 다음과 같은 메시지를 일부 포함하고 있다.

Permissive(허용적)는 매우 긍정적인 의미를 내포하고 있다. 따라서 허용적 라이선스(*permissive license*)와 카피레프트 라이선스(*copyleft license*)를 비교할 때 허용적 라이선스는 카피레프트 라이선스보다 어떤 면에서 더 낫다는 것을 암묵적으로 들어낸다. 자연스럽게 확장된 정의를 혼란스러워 하게 되고 해당 명칭의 내포적인 영향력 때문에 이러한 결론을 내리게 된다. 하지만, 인간의 사고방식이기에 어쩔 수 없다.

이 기본 메시지가 반드시 '잘못'된 것은 아니다. '카피레프트'에 의해 도입된 무엇인가를 환원해야 하는 의무사항에 비해 제한 없이 오픈소스 소프트웨어를 사용하고 싶은 사람들이나 회사에게는 편리할 수 있다.<sup>49)</sup> 그러나 카피레프트 라이선스 보호를 강조하는 사람이나 기업이 있을 수 있다. 실제로 적어도 오픈 소스 라이선스<sup>50)</sup> GPL<sup>51)</sup>는 자유를 보호하기 위해 개발자들이 그들의 "동료(neighbours)" 및 아래의 수정 사항들을 위해 개발되었다.<sup>52)</sup> 따라서 "카피레프트"는 "프로그램을 자유 소프트웨어로 만들기 위해 그리고 모든 수정, 확장된 버전의 프로그램 또한 자유 소프트웨어로 만들기 위한 방법론"<sup>53)</sup>이다. 이 방법<sup>54)</sup>은 "코드와 자유는 법적으로 분리 불가능"<sup>55)</sup>을 위한 것이다. 제한되지 않고 보호되길 바라는 이러한 상이한 이해관계 때문에 더 나은 명칭—관대한 라이선스 군의 편견 없는 이름—을 찾는 것이 나을지도 모른다. 그러나 우리는 최소한 이 분류체계는 등급을 없애는 기본 메시지를 담고 있음을 알고 있어야 한다.

48) FSF(자유소프트웨어 재단)에서도 '허용적 비-카피레프트 자유 소프트웨어 라이선스(*permissive non-copyleft free software license*)'라는 용어를 사용하며 (pars prototo: cf. Free Software Foundation: Various Licenses and Comments about Them: 2013 [n.y.] hURL: <http://www.gnu.org/licenses/license-list.html> - reference download: 2013-02-08, wp/section 'Original BSD license') '약한 카피레프트' 및 '강력한 카피레프트'라는 용어와 이를 대조한다(pars pro toto: cf. id., l.c., wp/section 'European Union Public License' 참조)

49) 사실상 카피레프트는 카피레프트가 아니다. 정의를 떠나서 그 영향은 카피레프트 '방법'을 적용하기 위한 조건을 정하는 특정 라이선스에 좌우된다. 예를 들어 GPL에서 카피레프트 영향은 '배포되는' 기준에 묶여진다. 나중에 이 조건들을 체계적으로 다룰 것이다 (see chapter Open Source Use Cases: Concept and Taxonomy, pp. 103). 이에 따라, 여기서 '일반화하는' 형식으로 설명한다.

50) RMS는 이를 자유 소프트웨어 라이선스로 불리는 것을 선호한다. (s. p. 18)

51) 원본 소스 Free Software Foundation: GNU General Public License, version 2; 1991 [n.y. of the html page itself] hURL: <http://www.gnu.org/licenses/gpl-2.0.html> - reference download: 2013-02-05, wp. OSLiC 안에서 우리는 오픈소스 라이선스에 대해 문서적으로 접근하기 때문에 OSI에서 배포되는 버전을 지속적으로 참조한다. 'OSI-GPL'에 대하여Open Source Initiative: GNU General Public License, version 2 (GPL-2.0). Version 2, June 1991; 1991 [n.y. of the html page itself] hURL:<http://opensource.org/licenses/GPL-2.0> - reference download: 2013-02-05, wp 참조

52) GNU 프로젝트의 역사는 여러 번 언급됐다. GNU 프로젝트와 초기 설립에 대하여 pars pro toto Williams, Sam: Free as in Freedom. Richard Stallman's Crusade for Free Software: Beijing [... etc.]: O'Reilly, 2002, ISBN 0-596-00287-4, passim 참조. 광범위한 여론 조사 pars pro toto Moody: Die Software-Rebellen, 2001, passim 참조. 매우 짧은 버전에서 리처드 스톤만은 일찍이 자유 소프트웨어 재단이 봉괴되었던 시기에 "단지 실행 사본을 받기 위해 이웃을 도와주지 않는다는 명확한 약속을 서명해야 하는 '기밀유지협약서(nondisclosure agreement)'를 목격했다고 언급하였다": "협조하는 공동사회는 금지되었다." (Stallman, Richard M.: The GNU Project: originally published in 'Open Sources: Voices from the Open Source Revolution, O'Reilly, 1999'; In Stallman: Free Software, Free Society: Selected Essays, 2002, p. 16 참조).

53) 리처드 스톤만: What is Copyleft?(카피레프트란 무엇인가?) 1996년 최초 작성; In Stallman: Free Software, Free Society(자유 소프트웨어, 자유 사회): Selected Essays(선정 에세이), 2002, 89쪽 참조

54) 미국 법적 저작권 체계에 근거하여 이 방법에서는 두 가지 단계를 사용한다. 첫째는 "저작권으로 보호받는다."이며, 둘째는 "배포 조항이 변경되지 않는 경우에만 프로그램의 코드 또는 그 파생된 프로그램을 사용, 수정 및 재배포할 권한을 모든 사람에게 제공하는 법적 수단인 배포 조항"이 추가된다. (참고문헌 참조)

55) 참고문헌 참조

오해의 소지가 있는 다른 해석은 –직관에 반대되는—‘카피레프트’ 개념의 사용으로부터 나온다. 관대한 라이선스와 반대로 카피레프트 라이선스 군을 나타냄으로써 함축적으로 두 가지 메시지를 띠운다. 첫째, 갖고 있던 수정된 프로그램을 재배포하는 것은 카피레프트 라이선스를 충분히 준수하는 것이다. 둘째, 허용적 라이선스는 소프트웨어 사용권을 확보하기 위해 어떤 것도 요구하지 않는다. 그렇게 해석되지 않기를 바라더라도 –인간이 늘 하듯—가능한 단순하게 사물을 받아들이는 경향이 있다.<sup>56)</sup> 여러 측면이 존재하기 때문에 카피레프트 라이선스와 허용적 라이선스의 모순을 이와 같이 이해한다면 심각한 일반화를 하는 것으로 받아들이는 것이 된다.

반면 ‘강력한 카피레프트’를 적용한 GPL의 경우에도 파생저작물을 단지 재배포하는 것 이외의 임무를 부여한다. 예를 들어 “GPL 라이선스를 적용하는 프로그램을 받아서 사용하는 사람에게는 프로그램과 함께 라이선스 사본을 같이 제공하는 것”을 요구하고 있다.<sup>57)</sup> 또한 ‘약한 카피레프트’ 라이선스를 적용할 경우 이 라이선스에 따라 충족해야 하는 더 많은 다른 기준이 요구된다. 예를 들어 EUPL에서는 소스코드와 함께 바이너리를 직접 제공하지 않는 허가권자(licensor)는 무료로 자신의 저작물의 소스코드 버전을 제공해야 하지만<sup>58)</sup> 동일한 상황에서 MPL은 “배포하는 비용 이내의 수수료를 그 소스코드 형태의 사본을 확보하는 데 요구할 수 있다”<sup>59)</sup>는 것을 요구한다. 그리고 마지막으로 허용적 라이선스 준수 사용을 위해 작업을 완수해야하며 더 나아가 다양한 일을 요구한다. 예를 들어 BSD에서는 “배포/재배포 시 위의 저작권 고지 문을 포함하도록 하고 필요시 재작성하여 포함할 것을” 요구한다. “저작권 고지 문”的 구조 때문에 이 의무적 고지는 소프트웨어 저작자 / 저작권 소유자의 이름이 공표되어야 한다는 점을 암시한다.<sup>60)</sup> 이와 반대로 아파치 라이선스(Apache License)에서는 “저작물에 배포의 일환으로 “고지(NOTICE)” 텍스트 파일이 포함된다면 사용자가 배포하는 파생저작물에는 파일의 중심 부분(저작자/저작권 소유자 이름 이외에 더 많은 정보가 들어갈 수 있는 부분)을 공개적으로 자주 제시해야 하는<sup>61)</sup> 고지 파일 내에 들어 있는 판독 가능한 귀속 고지(attribution notices)가 포함되어야 한다.

따라서 (그리고 이 분류체계에 대한 직관적 해석에 반하여) 의심할 여지없이 각각의 오픈소스 라이선스는 가장 허용적인 라이선스이더라도 어떤 활동을 통해 요건이 충족되어야 한다. 이 작업을 확인하기 위해서 라이선스 분류체계의 일반화된 개념이 아닌 라이선스 자체를 살펴보아야 한다. 이와 같이 잘 알려진 유형으로 오픈소스 라이선스를 분류한다 하더라도 특정 개별 라이선스의 준수 활동을 알아 낼 수 없

56) 저자들의 경험에 비추어 가끔은 그와 같이 단순하게 받아들이기 때문에 자신의 독자적 존재를 부각시키고 관리수준에서 결정을 한다. 그러나 이는 관리자의 잘못이 아니다. 이들의 업무는 정보를 합치고, 개괄하여 간략히 나타내는 것이다. 세부내용으로 다른 사람들을 당황하게 하지 않고 보다 나은 관점을 제공하는 것이 전문가의 일이다.

57) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §1. 참조

58) 독일 버전의 EUPL에서는 “problemlos und unentgeltlich auf den Quellcode (zugreifen können)”라는 문구를 사용한다 (Europäische Gemeinschaft a. European commission Joinup: Open-Source-Lizenz für die Europäische Union: 2007 (URL: [http://joinup.ec.europa.eu/system/files/DE/EUPL\\_v.1.1 - Lizenz.pdf](http://joinup.ec.europa.eu/system/files/DE/EUPL_v.1.1 - Lizenz.pdf)) - 참고자료 다운로드: 2013-02-08, 3쪽, section 3). 영어 버전에는 “소스코드는 쉽고 자유롭게 이용할 수 있다”라는 상세내용을 담고 있다 (유럽 공동체 a. European commission Joinup: European Union Public Licence v. 1.1. 2007 (URL: [http://joinup.ec.europa.eu/system/files/EN/EUPL\\_v.1.1 - Licence.pdf](http://joinup.ec.europa.eu/system/files/EN/EUPL_v.1.1 - Licence.pdf)) - 참고자료 다운로드: 2013-02-08, 2쪽, section 3).

59) Open Source Initiative: Mozilla Public License 2.0 (MPL-2.0): 2013 [n.y.] (URL: <http://opensource.org/licenses/MPL-2.0>) - 참고자료 다운로드: 2013-02-07, section 3.2.a. 참조

60) Open Source Initiative: The BSD 2-Clause License: 2012 [n.y.] (URL: <http://www.opensource.org/licenses/BSD-2-Clause>) - 참고자료 다운로드: 2012-07-03, wp. 참조

61) Open Source Initiative: Apache License, Version 2.0: 2004 [페이지 n.y.] (URL: <http://opensource.org/licenses/Apache-2.0>) - 참고자료 다운로드: 2013-02-07, wp. section 4.4. 참조

다. 내포된 메시지와 일반화된 일부 개념을 가지고 살아가길 원한다면 분류체계만으로 충분할 것이다. 그러나 분류체계는 오픈소스 라이선스를 충족하기 위해 해야 하는 일을 결정하는데 적절한 수단이 아니다. 특정 오픈소스 소프트웨어의 사용권한을 확보하기 위한 라이선스 준수 활동은 라이선서가 소프트웨어에 부여한 특정 오픈소스 라이선스에 근거해야 한다. 여기에 지름길이란 없다.

인간은 결론적으로 특정 목적을 위한 큰 영역의 세부적인 사항에 대해 논의해야 할 경우에도 그 영역에 대한 설명을 위해 일반화되고 체계화된 관점을 필요로 한다. 오픈소스 라이선스의 영역에 대해 논의하고 구성 할 수 있는 더욱 영리한 방법이 있을 것이다. 따라서 여기서는 오픈소스 라이선스를 분류하는 다른 방법에 접근 하려 한다.<sup>62)</sup>

일반적으로 라이선스들은 공통 목적이 있다. 그 공통 목적은 라이선스를 통해 무언가로부터 사람 또는 대상을 보호하는 것이다. 이 임무의 구조는 3가(價)의 동사인 '보호하다(protect)'라는 단어의 성격에 기초한다. 이 구조는 보호하는 사람 또는 대상과 보호받을 사람 또는 대상과 연결된다. 즉, 보호물/보호자와 이에 대항하는 피보호물/피보호자와 연결시킨다. 일반적으로 라이선스가 이런 역할을 한다. 또한 라이선시의 "권한"을 "보호"하는 것은 GPL2.0<sup>63)</sup>, LGPL-2.1<sup>64)</sup> 그리고 GPL3.0<sup>65)</sup>에서 확실하게 언급되었다 (LGPL-3.0에서 이 목적이 계승 됐듯이)<sup>66)</sup>. 이 관점에서 일반적으로 오픈소스 라이선스는 보호하기 위해 고안된 것으로 가정한다. 라이선스는 소프트웨어 사용자(수령자), 참여자 그리고 개발자 및 배포자 그리고 더 나아가 소프트웨어 자체를 보호 할 수 있다. 그리고 아래와 같은 여러 위협으로부터 보호할 수 있다.

첫째, 오픈소스 소프트웨어라는 측면에서 사용자는 소프트웨어 사용, 수정 및 재배포할 권한을 잃어버릴 위험으로부터 보호받을 수 있으며 또한 특허분쟁으로부터 보호받을 수 있다 추정한다.

둘째, 오픈소스 참여자와 배포자는 코드 개선 및 파생 코드의 형태의 피드백 손실과 보증 청구 및 특허 분쟁으로부터 보호받을 수 있다 추정한다.

셋째, 오픈소스 프로그램과 그 특정 형태는 (배포 가능하거나 불가능할 수 있고 수정 가능하거나 불가능 할 수 있으며 바이너리 또는 소스코드로 배포될 수 있는) 추가 개발하는 부분의 재사유화와 독점에 대해 보호받을 수 있다 추정한다.

넷째, 오픈소스 커뮤니티에 기반 한 새로운 개발물 들은 소프트웨어를 자유롭게 사용하는 세상을 확대하기 위해 소프트웨어 사유화에 대항하여 보호받을 수 있다 추정한다.<sup>67)</sup>

62) 궁극적으로, 라이선스 자체를 항상 살펴보아야 한다는 사실을 인정할 수 밖에 없더라도

63) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. (Preamble(전문). 참조

64) Open Source Initiative: The GNU Lesser General Public License, 버전 2.1 (LGPL-2.1); 1999 [html 페이지 n.y.] (URL: <http://opensource.org/licenses/LGPL-2.1>) - 참고자료 다운로드: 2013-03-06, wp. Preamble(전문). 참조

65) Open Source Initiative: GNU General Public License, 버전 3 (GPL-3.0); 2007 [html 페이지 n.y.] (URL: <http://opensource.org/licenses/GPL-3.0>) - 참고자료 다운로드: 2013-03-05, wp. Preamble(전문). 참조

66) Open Source Initiative: The GNU Lesser General Public License, 버전 3.0 (LGPL-3.0); 1999 [html 페이지 n.y.] (URL: <http://opensource.org/licenses/LGPL-3.0>) - 참고자료 다운로드: 2013-03-06, wp. 표제(prefix). 참조

이러한 관점을 존중하며 라이선스 특정 보호 기능을 보다 더 세밀하게 그려보게 된다. 따라서 다음페이지에서 각각의 오픈소스 라이선스의 보호기능을 설명하고 추론할 것이다. 표 2.1은 빠른 참조를 위해 간략하게 나타낸 것이다<sup>68)</sup>

---

67) 더 엄격한 버전에서 라이선스의 기능은 일련의 오픈소스 소프트웨어 정체 현상이 방지되도록 커뮤니티를 보호하는 권한으로 확인할 수 있다.

68) → 27쪽(표 위치)

표 2.1: 보호자 관념의 오픈소스 라이선스

오픈소스 라이선스 <sup>a</sup>	보호하는 대상														추가 개발 물	
	사용자			기여자 (배포자)		오픈소스 소프트웨어										
						미배 포	배포 시 상태									
	이미 소스 또는 바이너리를 입수한 자			오픈소스 소프트웨어를 확산시키는 자			비수정	수정								
							소스	바이 너리	소스	바이 너리						
	다음 항목에 대항															
	권한 상실			특허 분쟁	피드백 손실	보증 청구	특허 분쟁	이미 공개된 소프트웨어 독점화 및 재사유화						사유 화		
	사용	수정	재배 포													
Apache	2.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
BSD	3-Cl	✓	✓	✓	¬	¬	✓	¬	¬	✓	¬	✓	¬	¬	¬	
	2-Cl	✓	✓	✓	¬	¬	✓	¬	¬	✓	¬	✓	¬	¬	¬	
MIT		✓	✓	✓	¬	¬	✓	¬	¬	✓	¬	✓	¬	¬	¬	
MS-PL		✓	✓	✓	✓	¬	✓	✓	¬	✓	¬	✓	¬	¬	¬	
PostgreSQL		✓	✓	✓	¬	¬	✓	¬	¬	✓	¬	✓	¬	¬	¬	
PHP	3.0	✓	✓	✓	¬	¬	✓	¬	¬	✓	¬	✓	¬	¬	¬	

CDDL	1.0	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—
EPL	1.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	¬
EUPL	1.1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	¬
LGPL	2.1	✓	✓	✓	¬	✓	✓	✓	¬	✓	✓	✓	✓	✓	¬
	3.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	¬
MPL	1.0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	1.1	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	2.0	✓	✓	✓	✓	✓	✓	✓	¬	✓	✓	✓	✓	✓	¬
MS-RL		✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—

AGPL	3.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
GPL	2.1	✓	✓	✓	✓	¬	✓	✓	¬	✓	✓	✓	✓	✓	✓
	3.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

a) '✓'는 열에 나타낸 대상에 대하여 라이선스에 의해 보호된다는 표시이며,

'¬'는 열에 나타낸 대상에 대해 라이선스에 의해 보호되지 않는다는 표시이며,

'—'는 해당 문구가 여전히 평가되어야 한다는 표시이다.

이탤릭체로 표현된 라이선스 명칭은 해당 마인드맵(→ 42쪽)에서 다뤄지지 않고 이 표에서만 열거됨

## 2.1 GNU Affero General Public License(AGPL)의 보호 기능

[TODO...]

## 2.2 아파치 라이선스(Apache-2.0)의 보호 기능

승인된 오픈소스로서<sup>69)</sup> 아파치 라이선스<sup>70)</sup>는 소스코드 또는 바이너리 파일의 사본을 사용, 수정 또는 배포할 권리의 상실로부터 사용자를 보호한다.<sup>71)</sup> 게다가 특히 조항에 있어서<sup>72)</sup> 아파치 2.0 프로젝트는 특히 분쟁으로부터 사용자를 보호한다.<sup>73)</sup> 이러한 특히 조항과 "보증 제한(disclaimer of warranty)" 그리고 "책임 제한(limitation of liability)"으로 인해 아파치 라이선스는 기여자와 배포자를 특히 분쟁과 보증 청구로부터 보호한다.<sup>74)</sup> 마지막으로 아파치-2.0은 저작물을 폐쇄된 소프트웨어로 변환하는 라이선스 변경으로부터 배포된 소스코드를 보호한다. 그 이유는 첫째, "저작물 또는 파생저작물을 받아서 사용하는 다른 사용자에게 (아파치) 라이선스 사본을 제공"해야 하며 둘째, "배포하는 파생저작물의 소스코드 형태"로, "모든 저작권, 특히, 상표 및 귀속 고지를 보유"해야 하고 셋째, "받아서 사용하였던 최초 패키지에 의해 공급되는 고지(NOTICE)파일의 판독 가능한 사본을 포함"해야 하기 때문이다.<sup>75)</sup>

그러나 아파치 라이선스에서는 '카피레프트' 소프트웨어가 아니기 때문에 피드백 손실로부터 기여자를 보호하지 않는다. 아파치 라이선스는 소스코드 배포를 요구하는 문구가 들어있지 않다. 같은 맥락에서 아파치-2.0은 배포되지 않은 소프트웨어나 배포된 바이너리에 대하여 독점화(수정 또는 수정되지 않은 형태 모두)로부터 보호하지 않는다. 그 이유는 아파치 라이선스는 바이너리가 배포자에 의해 수정된 소스에 좌우된다 할지라도 소스코드의 공급 없이 바이너리를 (재)배포하는 것을 허용하기 때문이다. 마지막으로, 아파치-2.0은 사유화로부터 추가 개발물을 보호하지 않는다.

## 2.3 BSD 라이선스의 보호 기능

승인된 오픈소스 라이선스<sup>76)</sup>로서 BSD 라이선스<sup>77)</sup>는 소스코드 또는 바이너리 파일의 사본을 사용, 수정

69) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

70) 아파치 라이선스 버전 2.0은 아파치 소프트웨어 재단이 관리한다 (Apache Software Foundation: Apache License, 버전 2.0; 2004 (URL: <http://www.apache.org/licenses/LICENSE-2.0>)- 참고자료 다운로드: 2011-08-31, wp 참조). Open Source Initiative은 아파치 라이선스 사본을 제공하며(Open Source Initiative: APL-2.0, 2004, wp 참조) 공식 승인된 오픈소스 라이선스로 싣고 있다(Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp 참조). Open Source Initiative은 아파치 라이선스 1.1을 "대체 라이선스"로 분류한다(Open Source Initiative: 카테고리별 오픈소스 라이선스, 2013, wp 참조). 아파치 소프트웨어 재단은 동일한 취지로 1.0 및 1.1 배포 버전을 "기록물"로 분류한다(Apache Software Foundation: 라이선스: 2013 [n.y.] (URL: <http://www.apache.org/licenses/>)- 참고자료 다운로드: 2013-02-25, wp 참조). 그래서, OS LIC는 가장 최신 버전인 아파치-2.0에만 초점을 맞춘다. 초기의 아파치 라이선스에 따라 이행해야 하는 자들은 이를 BSD-2-Clause 및 BSD-3-Clause 라이선스의 동류로 이해하는데 이는 도움이 될 것이다.

71) Open Source Initiative: APL-2.0, 2004, wp. §2. 참조

72) → OS LIC 47쪽(3.1.2)

73) 참고문헌 참조

74) 참고문헌 참조

75) 참고문헌 참조

76) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

77) BSD는 버클리 소프트웨어 배포의 약자이다. BSD 라이선스 공개 및 명명에 대한 자세한 설명인 Open Source Initiative: The BSD 3-Clause License: 2012 [n.y.] (URL: <http://www.opensource.org/licenses/BSD-3-Clause>) - 참고자료 다운로드: 2012-07-04, wp. 사설 참조

또는 배포할 권리의 상실로부터 사용자를 보호한다.<sup>78)</sup> 추가로 ‘무보증 조항(No Warranty Clause)’이라는 내용이 포함되었기 때문에 소프트웨어 사용자의 보증 청구로부터 기여자 그리고(또는) 배포자를 보호한다.<sup>79)</sup> 그리고 마지막으로 소스코드가 폐쇄되는 라이선스 변경으로부터 기존에 배포된 소스코드를 보호한다. 그 이유는 “해당 소스코드의 재배포 및 수정에는 저작권 고지, 이에 관련된 조건 리스트, 그리고 권리 포기 내용이 포함”되어야하기 때문이다.<sup>80)</sup> 그러므로 소스코드를 독점화하는 것과 관계없이 기존 BSD 라이선스가 적용된 코드를 다른 라이선스를 적용하여 배포하는 것은 올바르지 않다.<sup>81)</sup>

하지만 BSD 라이선스 프로젝트는 특허분쟁에 대하여 기여자 또는 배포자를 보호하지 않으며(어떠한 특허 조항도 포함되어있지 않기 때문에), 기여자를 피드백 손실로부터 보호하지 않는다(‘copyleft’ 소프트웨어가 아니기 때문에). 또한, 수정 또는 수정되지 않은 형태에 관계없이 배포되지 않는 소프트웨어와 배포된 바이너리의 독점화로부터 보호하지 않는다. 그 이유는 소스코드를 공급하지 않고도 바이너리만의 재배포를 허용하기 때문이다.<sup>82)</sup> 마지막으로 BSD 라이선스는 사유화로부터 추가 개발물을 보호하지 않는다.

## 2.4 CDDL의 보호기능

승인된 오픈소스 라이선스<sup>83)</sup>로서 CDDL(Common Development and Distribution License)은 소스코드 또는 바이너리 사본을 사용, 수정 또는 배포할 권리의 상실로부터 사용자를 보호한다.<sup>84)</sup>

[...]

## 2.5 Eclipse Public License(EPL)의 보호기능

승인된 오픈소스 라이선스<sup>85)</sup>로서 EPL(Eclipse Public License)<sup>86)</sup>은 소스코드 또는 바이너리 사본을 사용,

78) Open Source Initiative: 오픈소스 정의, 2012, wp. §1. 참조

79) Open Source Initiative: The BSD 2-Clause License, 2012, wp. 참조

80) 참고문헌 참조

81) 일반적인 상식에서 다른 자체 라이선스에 의거 저작물을 대체할 수 있는 BSD 라이선스에 대해 들어본 적이 있을 수 있다. BSD 라이선스가 모든 상황에서 유효하지 않다는 점을 받아들인다면 맞는 말이다. 수정된 저작물(파생저작물)의 소스코드를 재배포할 필요가 없다. 받은 버전을 수정하고 결과물을 바이너리로서만 배포하며 개선사항을 비밀로 해둘 수 있다. 그러나 수정 프로그램의 소스코드를 배포한다면 “수정되었든 수정되지 않았든 소스로 재배포하는 행위는 그 소스코드 재배포 시 위의 저작권 고지, 조건 리스트 및 뒤 따르는 권리 포기가 유지된다는 조건에서 허용되기” 때문에 라이선싱이 유지된다. (참고문헌 참조)

82) BSD-2CL 라이선스 (참고문헌 참조)와 BSD-3CL 라이선스 참조 (Open Source Initiative: The BSD 3-Clause License, 2012, wp)

83) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

84) Open Source Initiative: Common Development and Distribution License (CDDL-1.0); 2004 [n.y. of the html page itself] URL: <http://opensource.org/licenses/CDDL-1.0i> -reference download: 2013-04-19, wp. §?. 참조

85) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

86) 폰소스 이니시어티브: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

[86] Eclipse Public License, 버전 1.0은 이클립스 소프트웨어 재단(Eclipse Software Foundation)에서 관리한다(이클립스 재단: Eclipse Public License, 버전 1.0; 2005 [n.y.]URL: <http://www.eclipse.org/org/documents/epl-v10.php>) - 참고자료 다운로드: 2013-02-20, wp)) 물론 OSI도 복사본을 제공한다.( Open Source Initiative: Eclipse Public License, Version 1.0; 2005 [n.y. of the page itself] hURL: <http://opensource.org/licenses/EPL-1.0i> { reference download: 2013-02-20, wp 참조).

수정 또는 배포할 권리의 상실로부터 사용자를 보호한다.<sup>87)</sup> 또한 특히 조항에 따르면<sup>88)</sup> EPL은 특히 분쟁에 대해서도 사용자를 보호한다.<sup>89)</sup> 특히 조항 외에도 EPL은 “무보증(no warranty)”과 “책임 제한(disclaimer of liability)” 부분을 포함한다.<sup>90)</sup> 이 세 가지 요소는 모두 기여자와 배포자를 특허분쟁과 보증 청구로부터 보호한다. 마지막으로 EPL은 소프트웨어를 독점소프트웨어로 리셋하는 라이선스 변경으로부터 배포된 소스코드를 보호한다. 먼저, EPL은 EPL을 적용하여 배포된 어떠한 저작물이 “소스코드 형태로 배포된다면 (이 때) 이는 마찬가지로 (EPL) 협약 하에 배포되어야 한다” 이러한 배포가능 행위 시 “각 [배포된]프로그램의 복사본” 또는 프로그램 패키지에 “반드시” EPL의 “복사본”을 삽입해야한다.<sup>91)</sup> 하지만 허용적 라이선스와는 반대로 EPL은 수정된 또는 수정되지 않은 것과는 관계없이 배포된 소스코드만을 보호하는 것은 아니다. EPL은 수정된 또는 수정되지 않은 바이너리 또한 보호한다. EPL은 라이선스에 “프로그램을 위한 소스코드를 사용가능함”이 명시된 경우나 “라이선시”가 “소프트웨어 교환에 관례적으로 쓰이는 매체를 통해 또는 매체에서 합당한 방식으로 이를 확보할 수 있는” 경우 수정하는 “기여자”와 배포자가 “자체 라이선스 협약 하에 오브젝트 코드의 형태로 프로그램을 배포”할 수 있도록 허용한다.<sup>92)</sup> 그러므로 EPL은 카피레프트 라이선스라는 결론을 내린다.

하지만 EPL은 효력이 강력한 카피레프트를 포함한 라이선스는 아니다. EPL에서 약한 효력의 카피레프트 ‘만’ 있다.<sup>93)</sup> 실제로 EPL에서는 EPL 라이선스에 따른 “프로그램”(오브젝트 형태로 배포된)의 위치의 경우 일치하는 소스코드를 어디에서 얻을 수 있는지 알 수 있도록 해야 한다.<sup>94)</sup> ‘프로그램’이란 용어는 “EPL에 따라 배포된 기여(contribution)”로 정의되며 ‘기여’라는 용어는 (다른 요소 외에) “프로그램의 변경된 내용 및 프로그램에 추가된 내용”을 나타낸다.<sup>95)</sup> 이는 다음과 같은 정의가 내려질 수 있다. ‘프로그램’은 ‘기여’로 정의할 수 있고 ‘기여’는 ‘프로그램’으로 정의할 수 있다. 그럼에도 불구하고 라이선스는 호의적으로 해석해야한다. 논란의 여지가 없는 점은 여기에 있다. EPL 라이선스에 따라 수정된 프로그램, 라이브러리, 모듈 또는 플러그인을 배포해야 한다면 수정된 소스코드 또한 공개해야 한다. EPL 라이선스에 따른 프로그램(프로시저 호출을 추가하여 이를 사용하는 대신에 수정하였던)에 의해 사용되는 일부 자체 플러그인 또는 추가 라이브러리를 “추가”한다면 두 가지 부분의 코드를 공개해야 한다.(프로그램 및 추가된 요소의 코드). 이런 의미에서 EPL에서는 다른 약한 효력의 카피레프트를 활용하는 라이선스와 같이 독점화로부터 바이너리가 확실히 보호된다. 그러나 EPL 라이선스에 따르지 않는 다른 상위 프로그램의 구성요소로 사용되는 EPL 기반 라이브러리만을 배포한다면, 이 라이브러리는 상위 개발물을 따르지 않는다. (단, 라이브러리 자체가 지배적인 추가 개발물에 의해 전달되는 프로시저 또는 (프로그램) 함수를 불러내지 않는 경우). 이러한 이유로, 라이브러리에는 추가되는 것이 없으며, 해당 라이브러리 외의 코드는 공개하지 않아야 한다. 따라서 EPL에서는 예를 들어 GPL과 같은 취지의 효력이 강력한 카피레프트를 사용하지는 않는다.

87) 참고문헌 참조, wp. §2a.

88) OSLiC 49쪽 참고

89) 참고문헌 wp. §2b & §2c.

90) 참고문헌 wp. §5 & §6. 참조

91) 참고문헌 wp. §3. 참조

92) Open Source Initiative: EPL-1.0, 2005, wp. §3, 특히, §3.b.iv. 참조

93)인터넷에서 상반되는 세부내용을 발견할 수 있더라도 그러하다. 전체를 대표하는 부분 ifross: ifross Lizenzz-Center, 2011, wp 참조: 이 페이지는 “강력한 카피레프트 효력”이 있는 다른 라이선스” 부문의 EPL를 열거하고 있다.

94) Open Source Initiative: EPL-1.0, 2005, wp. §3, 특히, §3.b.iv. 참조

95) 참고문헌 wp. §1. 참조

## 2.6 European Union Public License(EUPL)의 보호 기능

승인된 오픈소스 라이선스<sup>96)</sup>로서, European Union Public License(EUPL)<sup>97)</sup>는 받은 소스코드 또는 바이너리 사본을 사용, 수정 또는 배포하는 권리의 상실로부터 사용자를 보호한다.<sup>98)</sup> 또한 해당 특허조항<sup>99)</sup>에 근거하여 EUPL은 특허분쟁에 대해서도 사용자를 보호한다.<sup>100)</sup> 이 특허조항 외에도 EUPL에는 추가로 “보증 제한”과 “책임 제한”에 대한 내용을 담고 있다.<sup>101)</sup> 이 세 가지 요소에 의해 특허분쟁과 보증 청구에 대해 기여자와 배포자가 보호된다. 끝으로, EUPL은 또한 독점화 및 재사유화로부터 배포된 소스를 그리고 피드백 손실로부터 기여자를 보호한다. 이런 보호는 두 가지 조치에 근거한다. 첫째, EUPL에는 “원저작물과 원저작물기반 파생작의 사본”을 반드시 “EUPL 조항에 의거”하여 배포해야 한다는 사항이 규정된 “카피레프트 조항”이라는 제목의 단락이 포함되어 있다.<sup>102)</sup> 둘째, EUPL은 (“저작물을 지속적으로 배포 또는 전달하는” 한) 각 라이선시는 “소스코드를 쉽고 자유롭게 이용할 수 있는 저장소(repository)를 (표시하여)” 또는 직접적으로 “소스코드를 제공”해야 한다.<sup>103)</sup> 이 조건은 EUPL에게 매우 중요하기 때문에 라이선스에서 메시지가 반복되고 있다. 또 다른 단락에서 EUPL은 “저작물이 실행코드(Executable Code)로 제공된다면 라이센서는 저작물 복사본과 함께 저작물의 기계 판독형 소스코드 복사본을 추가적으로 제공하거나 라이센서가 저작물을 지속적으로 배포하는 동안 소스코드를 쉽고 자유롭게 이용할 수 있는 저장소를 고지문에 표시”할 것을 요구하고 있다.<sup>104)</sup> EUPL에 의해 “원저작물 또는 이의 파생작”으로서 정의되는 “저작물”的 의미에서 볼 때<sup>105)</sup> EUPL은 카피레프트 라이선스임에 틀림없다.

그러나 EUPL은 강력한 카피레프트를 포함하는 라이선스는 아니다. EUPL의 핵심을 받아들인다면 해당 라이선스는 보통 두 가지의 상황에서 모두 소스코드를 공개해야하기 때문에 (재)사유화 및 독점화로부터 원 저작물의 수정을 뿐만 아니라 추가 개발물을 보호하는 것으로 보인다. 이와 같이 이해한다면, EUPL은 ‘강력한 카피레프트 라이선스’일 것이다. 그러나 EUPL은 추가로 “호환성 조항(Compatible clause)”을 포함함에 따라 “라이선시가 호환성 라이선스 하의 원 저작물과 다른 저작물에 기초한 파생작 또는 사본을 배포한다면, 호환성 라이선스 조항에 기반하여 배포가 가능하다”는 것을 명시하는 조항을 포함한다.<sup>106)</sup> (“호환성 라이선스”라는 용어는 예를 들어, Eclipse Public License와 같은 호환 가능한 라이선스 리스트에 의해 명확히 정의된다.<sup>107)</sup> 이런 호환성 조항에 근거하여 추가 개발물의 코드를 공개

96) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

97) European Union Public License(EUPL) 버전 1.1은 유럽 연합(European Union)에서 관리하며, “Joinup”이란 명칭 하에 주관하고 있다(European Community(유럽 공동체) a. European commission Joinup: EUPL-1.1/EN, 2007, wp 참조). EUPL은 공식적으로 여러 언어를 독일어로 번역하였다((Europäische Gemeinschaft a. European commission Joinup: EUPL-1.1/DE, 2007, wp 참조). 이와 같이 다중 언어로 되어 있기 때문에, Open Source Initiative는 자체 버전을 제공하지 않지만 유럽 호스트 “Joinup”的 랜딩 페이지에 링크된 랜딩(시작) 페이지만을 제공 (Open Source Initiative: European Union Public License, version 1.1 (EUPL-1.1: 2007 [html 페이지 n.y.] )

(URL: <http://opensource.org/licenses/EUPL-1.1>) - 참고자료 다운로드: 2013-03-04, wp 참조).

98) Europäische Gemeinschaft a. European commission Joinup: EUPL-1.1/DE, 2007, wp. §2. 참조

99) OSLiC 50쪽 참고

100) 유럽 공동체 a. European commission Joinup: EUPL-1.1/EN, 2007, wp. §2 참조, 말미에서.

101) 참고문헌 wp. §7 & §8. 참조

102) 참고문헌 wp. wp. §5. 참조

103) 참고문헌 참조

104) 참고문헌 wp. §3. 참조

105) Open Source Initiative: EUPL-1.1 (OSI), 2007, wp. §1 참조

106) 참고문헌 wp. §5. 참조

107) Appendix 참조

하는 의무 사항은 뒤집어질 수 있다. 첫 단계에서 EUPL을 따르는 라이브러리를 사용하면서 EPL<sup>108)</sup>을 따르는 다소 무익한 추가 기능 어플리케이션 정도를 공개할 수 있다. 두 번째 단계에서 EUPL 라이선스를 유지하는 대신에 EPL에 기반하여 배포할 수 있는 'EUPL 라이브러리'가 추가한다. 그리하여 최종적으로 약한 효력의 카피레프트 라이선스인 Eclipse Public License에 기반하여 동일한 저작물이 확보된다.<sup>109)</sup> EUPL-1.1의 보호 기능은 라이선스 내용 자체를 기준으로 추정할 만큼 포괄적이지 않으므로<sup>110)</sup> EUPL 저자가 강력한 카피레프트 라이선스를 작성하고자 한다는 인상이 독자에게 심어진다 하더라도 기껏해야 약한 효력의 카피레프트 라이선스가 될 수 있다. 그러나 EUPL 라이선스에서는 사유화에 대해 추가개발물이 보호되지 않는다.

## 2.7 GNU General Public License(GPL)의 보호 기능

GPL로 알려진 GNU General Public License는 자유 소프트웨어 재단에서 관리하고 제공되는 라이선스로서 "GNU 운영체제 홈페이지"의 일부로 운영되고 있다.<sup>111)</sup> 현재, OSI가 승인하는 오픈소스 라이선스에는<sup>112)</sup> GPL-2.0<sup>113)</sup>, GPL-3.0<sup>114)</sup> 두 가지 버전이 있다. 두 가지 GPL 버전이 동일한 결과와 취지를 목표로 하고 있지만 문맥과 논증 구조에 대해선 다른 경향이다. 따라서 이 두 라이선스는 별도로 다루어야 할 것이다.

### 2.7.1 GPL-2.0

GPL-2.0의 보호 기능은 쉽게 결정될 수 있다. 첫째, 소프트웨어를 받은 사용자가 수정되지 않은 "소스코드의 사본"<sup>115)</sup>을 "복사 및 배포"하고, "사본 또는 그 일부분을 수정하여 그 수정본을 배포"<sup>116)</sup> (소스코드 형태뿐만 아니라 바이너리 형태로도<sup>117)</sup>) 할 수 있다. 그러므로 승인된 오픈소스 라이선스<sup>118)</sup>에 따라

108) 라이선스 내용을 따른다면 다소 무익한 어플리케이션은 EUPL 라이브러리의 함수를 호출하여 EUPL 라이브러리에 의존해야 할 필요가 없다. 라이선스 내용은 "양립성 라이선스에 의한 다른 [그 밖의] 저작물은 "파생작"과 함께 배포될 수 있음을" 제시할 뿐이다. 이 표현에 의해 라이선스에서는 파생작과 다른 저작물간에 대비를 이루고 있다 (다른 저작물은 반드시 파생작이 될 필요는 없음을 제시).

109) OSLiC 49쪽 참고

110) EUPL의 보호 기능을 규정하는 유형은 FSF에서 처음에 제시하고 있다.(Free Software Foundation: Various Licenses and Comments about Them, 2013, pp. wp. section 'European Union Public License'). The EU answers that publishing such a trick will comprise its user in the eyes of the open source community (cf. European Community a. European commission Joinup: New FSF statements on the EUPL are a step in the right direction; 2013 [n.y] hURL: <https://joinup.ec.europa.eu/community/eupl/news/new-fsf-statements-eupl-are-step-right-direction> { reference download: 2013-03-05, p. wp 참조}. 이는 의심할 여지없이 사실이다. 하지만, 이러한 논쟁은 EUPL에 의해 제기된 보호막의 구멍을 막지 못한다.

111) 자유소프트웨어 재단: GNU 운영체제[:] 라이선스; 2011 (URL: <http://www.gnu.org/licenses/>) - 참고자료 다운로드: 2013-03-25, wp. 참조

112) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

113) 최초 버전은 FSF에서 제공, Free Software Foundation: The GPL-2.0 License (FSF), 1991, wp. Open Source Initiative에서 제공하는 버전 참조, Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. 참조

114) 최초 버전은 FSF에서 제공 Free Software Foundation: GNU General Public License [version 3]; 2007 [html 페이지 n.y. ] (URL: <http://www.gnu.org/licenses/gpl.html>) - 참고자료 다운로드: 2013-03-06, wp 참조. Open Source Initiative에서 제공하는 버전 Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. 참조

115) Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. §1. 참조

116) 인용한 글과 동일, wp. §2. 참조

117) 인용한 글과 동일, wp. §3. 참조

118) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

GPL-2.0은 배포 받은 소스코드나 바이너리 사본의 사용, 수정, 배포할 권리상실로부터 사용자를 보호한다. 둘째, GPL2.0은 보증 청구<sup>119)</sup>와 피드백의 손실로부터 (카피레프트에 의해<sup>120)</sup>) 사용자를 보호한다. 셋째, GPL2-0은 사유화에 대해서 거의 완전히 소스코드 그 자체를 보호한다. 원 저작물의 일부를 새 파생작에 복사하여<sup>121)</sup> 다시 제작한(원저작물을 “기반으로한 작업”) 바이너리 수정 버전만을 처음에 배포하더라도 소스코드<sup>122)</sup>(즉, 수정된 전체 저작물<sup>123)</sup>)를 제공해야 한다. 이 수정된 “[원본]프로그램을 기반으로 한 저작물”은 매우 넓은 의미로 해석해야 한다. 이는 “copyright에 의거한 프로그램 또는 파생작”을 의미한다. 다시 말해, 글자 그대로 또는 수정되거나 다른 언어로 번역되든 간에 프로그램 또는 그 일부를 포함하는 저작물을 뜻한다.”<sup>124)</sup> 이러한 이유로, 소프트웨어 배포에서, GPL-2.0은 소프트웨어의 재사유화뿐만 아니라 추가 개발물을 사유하는 것에 대해서도 대항하여 보호한다.

그러나 GPL-2.0에 의해 특히분쟁에 대해서는 사용자나 기여자 또는 배포자가 보호받지 못하며<sup>125)</sup>, (재)사유화에 대해 배포되지 않은 (수정된) 소프트웨어는 보호되지 않는다.<sup>126)</sup>

## 2.7.2 GPL-3.0

GPL-3.0에서 중요한 변경사항은 “GPL 적용 저작물(covered work)”의 “유포(propagate)” 또는 “전송(convey)”라는 문구의 사용이다. “GPL 적용 저작물”은 “수정되지 않은 프로그램 또는 그 프로그램을 기반으로 한 저작물”을 나타낸다. “특정 프로그램을 기반으로 하는 저작물”은 “초기 저작물의 “수정된 버전”으로 정의되는데 여기서 “수정된 버전”이란 특정 프로그램을 전체 혹은 일부분을 (복제)(수정)하는 것으로부터 만들어지는 수정된 저작물을 의미한다.<sup>127)</sup> 반면에 “저작물 유포”는 대중이 이용할 수 있게 하는 “복사, 배포(수정본, 원본)와 “개인 사본을 수정하거나 컴퓨터에서 실행하는 것을 제외한” 기타 유형의 저작물을 다루는 행위를 나타낸다.<sup>128)</sup> 셋째, GPL-3.0에서는 저작물을 “convey”하는 것은 “다른 사람들이 복사본을 만들거나 배포 받을 수 있게 하는 모든 유형의 배포 행위를 의미한다”라고 명시하고 있다.<sup>129)</sup> 이 설명은 받는 사람이 활발히 프로그램을 복사하거나 패치(fetch)하는 경우도 배포행위라는 점을 명확히 하는 것에 도움이 된다.

119) Open Source Initiative: GPL-2.0 라이선스 (OSI), 1991, wp. §11, §12. 참조

120) 인용한 글과 동일, wp. §3. 참조

121) 인용한 글과 동일, wp. §2. 참조

122) Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. §1. 참조

123) 인용한 글과 동일, wp. §3. 참조

124) 인용한 글과 동일, wp. §0. 참조

125) → OSLiC, 50쪽

126) 이는 AGPL이 비슷해지려는 GPL의 ‘부족한’ 부분이다. GPL-2.0 라이선스에 따른 서버 소프트웨어를 수정하여 사회에 다시 개선된 소프트웨어를 제공할 의무 없이 수정된 소프트웨어 기반 서비스 제공을 위해 자체 기계에 설치 할 수 있다. 그러나 적어도 독일에서는 이 점에서 엄격한 제한을 중시하는 것으로 보인다. 보유 컴파트먼트로 소프트웨어를 배포하는 것은(GPL-2.0 라이선스 소프트웨어의 경우) 소스코드도 배포해야 하는 의무를 유발하는 배포 행위이다. [중요: 인용문 필요함!]

127) Open Source Initiative: GPL-3.0 라이선스 (OSI), 2007, wp. §0. 참조

128) Open Source Initiative: GPL-3.0 라이선스 (OSI), 2007, wp. §0. 참조. GPL 3.0은 특정 제약사항을 직접 다루지 않고 전 세계 국가의 저작권 체계에 적용할 필요가 있다. 따라서 이 라이선스에서는 FSF 취지에서 “저작물을 유포”라는 문구의 의미는 일반적으로 “개인 사본을 수정하거나 컴퓨터에서 실행하는 것을 제외하고” 이 문구가 적용되어야하는 특정 저작권 체계라고 명시하고 있다.

129) 위 책과 동일 참조

새로운 문구를 언급하며 GPL-3.0은 “기본적 허가(basic permission)”로 “라이선스에 달리 효력이 남아있는 한 조건 없이 표준이 적용된 저작물을 만들고, 실행하며, 배포”하는 것을 허용한다.<sup>130)</sup> 이는 중시해야 하는 어떠한 규정이 존재하지 않는다면 제한 없이 무엇이든 허용되는 것으로 해석될 수 있다. 이 설명을 기준으로 볼 때 GPL-3.0 프로그램을 개인적인 목적으로 사용 및 수정하는 것은 제한되지 않는다.<sup>131)</sup>

따라서 보통 (OSD로 정의되는 다른 오픈소스 라이선스와 같이<sup>132)</sup>) GPL은 사용자가 받은 소스코드나 바이너리를 사용, 수정, 배포할 수 있는 권리를 지켜준다.<sup>133)</sup> 더 나아가 해당 특히 조항을 기반으로 GPL-3.0은 소프트웨어 사용자와 기여자를 특허분쟁으로부터 보호한다.<sup>134)</sup> 게다가 GPL-3.0에선 적어도 소스코드 배포 시 명확히 알려야 한다고 잘 알려진 “보증 제한<sup>135)</sup>”과 “법적 책임 배제<sup>136)</sup>”를 통해 기여자와 배포자를 보증 청구로부터 보호하려고 한다.<sup>137)</sup> 마지막으로 GPL-3.0의 가장 강력한 보호 내용은 피드백 손실과 사유화에 대한 보호에 관한 것이다. GPL-3.0 라이선스 프로그램을 바이너리 형태로 배포할 때마다 그 소스에도 접근 가능하게 해야 한다.<sup>138)</sup> 또한, 위 의무는 모든 GPL 적용 저작물에 해당된다. 그러므로 수정되지 않은 원 저작물 뿐만 아니라 모든 수정본과 원저 작물의 전체 또는 일부를 복사하여 재제작하거나 수정한 “결과물”등은 모두 위 의무를 따라야한다.<sup>139)</sup> “이 라이선스 조항에 기반하여 기계 판독 가능한(machine readable) 해당 소스를 배포한다면 조항 4 그리고 5에 따라 오브젝트 코드 형태로도 GPL 적용 저작물을 배포 할 수 있다.”<sup>140)</sup> 따라서 의심할 여지없이, GPL은 추가 개발물들의 기반으로 사용된 수정 프로그램 또는 라이브러리의 코드 뿐만 아니라 모든 추가개발물의 코드가 공개하기를 요구한다.

GPL이 사유화로부터 보호하지 않는 단독 사용 형태는 오직 개인 혼자서 사용하는 경우만이다.<sup>141)</sup>

## 2.8 Lesser GNU General Public License(LGPL)의 보호 기능

LGPL은 자유 소프트웨어 재단(Free Software Foundation)에서 관리하고 제공하며, 잘 알려진 “GNU 운영체제 홈페이지”的 일부분으로 주관하고 있다.<sup>142)</sup> LGPL의 의미는 시간이 경과되면서 변하였다. 첫째, 1991년에는 “GNU Library General Public License”로서 설명되었고, “최초의 Library GPL 공개 버전”으로

130) 인용한 글과 동일, wp. §2. 참조

131) 일반적으로, 단독으로 오픈소스 소프트웨어 부품을 사용 중이라면 부과된 일을 이행할 필요가 없음을 추론하게 된다(즉, 특정 라이선스 규정은 단독사용이 아니라 소프트웨어 배포에만 초점을 둔다는 사실에 기준). 그러나 GPL-3.0에서 이와 같은 단독사용과 관련하여 내세우는 단언은 보다 명확해진다. 이런 단언은 “라이선스의 효력이 남아 있는 한 조건 없이 전달하지 않는 GPL 적용 저작물을 만들고, 실행하며, 유포하도록 하는” “기본 허가” 중 하나이다. “저작물 유포”는 “개인 사본을 수정하거나 컴퓨터에서 실행하는 것을 제외한” 어떤 행위를 나타낸다(인용한 글과 동일, wp. §2, §0 참조). 따라서 GPL-3.0은 자신의 컴퓨터(machine)에 대한 완전한 자유를 뒷받침한다. 제3자에게 결과물을 전달하지 않는 한 하고 싶은 일을 하기 바란다.

132) Open Source Initiative: 오픈소스 정의, 2012, wp. 참조

133) Open Source Initiative: GPL-3.0 라이선스 (OSI), 2007, wp. §3, §4, §5, §6. 참조

134) → OSLiC, 51쪽

135) 인용한 글과 동일, wp. §15. 참조

136) 인용한 글과 동일, wp. §16. 참조

137) 인용한 글과 동일, wp. §4. 참조

138) Open Source Initiative: GPL-3.0 라이선스 (OSI), 2007, wp. §6. 참조

139) 인용한 글과 동일, wp. §0. 참조

140) 인용한 글과 동일, wp. §6. 참조

141) 이와는 정반대로 GPL-3.0에서는 제3자에게 수정물을 넘기고 “단독 사용을 위해 수정할 수 있게 할 목적으로 타인에게 소스코드를 작업의 기초로서 배포할 수 있다 (인용한 글과 동일, wp. §2 참조).

142) Free Software Foundation: GNU OS 라이선스, 2011, wp. 참조

표현될 수 있는데, “정상적인 GPL 2버전과 병행했기 때문에 2라는 번호가 따라붙기도 했다. 오늘날, 이 라이선스는 “GNU Lesser General Public License로 대체된” 것으로 표시된다.<sup>143)</sup> 1999년에 더 새로워진 LGPL 버전은 “버전 번호 2.1, GNU Library Public License 버전 2를 계승한 라이선스”로 공개되었다.<sup>144)</sup> 마지막으로 2007년 6월에 최신 버전, 즉 새로운 구조를 포함하는 LGPL이 공개되었다. GPL-2.0과 LGPL-2.1은 유사하면서도 독자적인 라이선스들이지만, LGPL-3.0은 GPL-3.0의 추가사항으로 해석된다. LGPL-3.0 라이선스 서두에는, 그에 부합하는 GPL-3.0의 내용이 다음의 문장으로 포함되어 있다. “GNU Lesser General Public License에는 아래에 열거한 추가 허가사항이 보강된 GNU General Public License 버전 3의 조항과 조건을 포함한다.”<sup>145)</sup> 이런 차이점에 기초하여 여러 LGPL을 구분하여 다루어야 할 것이다.

### 2.8.1 LGPL-2.1

다른 버전의 GPL 또는 LPGL과 같이 LGPL-2.1은 그 목적을 소프트웨어 사용자의 “권한”을 “보호”하기 위한 과정으로 명확히 기술하고 있다.(일반적으로 모든 “GNU General Public License에서는 자유 소프트웨어를 공유 및 변경하는 자유를 보장하기 위한 것이다”라고 명확히 밝히고 있다.)<sup>146)</sup> 따라서 LGPL-2.1은 받은 소스코드 또는 바이너리의 사본을 사용, 수정 또는 배포하는 권한 상실에 대해 사용자를 보호하는 승인된 오픈소스 라이선스<sup>147)</sup>이다.<sup>148)</sup> 그러나 LGPL-2.1에서는 소프트웨어 사용자의 특히 권한을 허가한다고 추정되는 문장을 제시하진 않는다.<sup>149)</sup> 따라서 기여자는 배포자는 특허분쟁에 대해 보호되지 않는다. 그 대신 LGPL-2.1에는 보증 청구에 대해 기여자와 배포자를 보호하기 위해 이를 확증하는 “보증 없음”이라는 제하의 특별한 두 문단을 포함하고 있다.<sup>150)</sup> 마지막으로, LGPL-2.1에서는 재폐쇄 및 재사유화에 대해 이미 배포된 소스 그리고 피드백 순실에 대해 기여자가 보호된다. 이런 목적에서, 한편으로 LGPL-2.1에서는 수정결과가 “(LGPL-2.1) 조항에 의거 제3자에게 무료로 허가 된다”는 조건으로, 받는 사람은 “라이브러리 또는 일부의 사본 또는 사본들을 수정하고 그 수정물을 복사하고 배포할 수 있다”라고 명시하고 있다.<sup>151)</sup> 다른 한편으로는, 이 LGPL 버전에서는 “목적코드 또는 실행 가능한 형태로” 수정물 배포를 허용한다. 단, LGPL-2.1 조항에 의해 완전히 동일한 기계가 읽을 수 있는 소스코드”로 배포되어야 한다.<sup>152)</sup>

그러나 GPL과는 반대로 LGPL에서는 전체 프로그램 또는 추가개발의 코드 공개를 요구하지 않는다. 여기서는 “라이브러리를 사용하는 저작물”을 “라이브러리 기반 저작물”과 구분하고 있는데 그 이유는 다음과 같다. 첫째, 라이브러리를 LGPL-2.1에 의해 허가받은 “소프트웨어 라이브러리 또는 저작물”로 정의하며, “라이브러리 기반 저작물”이란 저작권법에 기반하는 라이브러리 또는 파생작을 의미한다고 덧붙인다.<sup>153)</sup>

143) Free Software Foundation: GNU Library General Public License [버전 2.0]; 1991 [html 페이지 n.y.] (URL: <http://www.gnu.org/licenses/old-licenses/lgpl-2.0.html> - 참고자료 다운로드: 2013-03-25, wp. 참조)

144) Free Software Foundation: GNU Lesser General Public License [버전 2.1]; 1999 [html 페이지 n.y.] (URL: <http://www.gnu.org/licenses/lgpl-2.1.html>) - 참고자료 다운로드: 2013-03-06, wp. 참조

145) Free Software Foundation: GNU Lesser General Public License [버전 3]; 2007 [html 페이지 n.y.] (URL: <http://www.gnu.org/copyleft/lesser.html>) - 참고자료 다운로드: 2013-03-06, wp. 참조

146) Open Source Initiative: LGPL-2.1 라이선스 (OSI), 1999, wp. 서문. 참조

147) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2-12, wp. 참조

148) Open Source Initiative: LGPL-2.1 라이선스 (OSI), 1999, wp. §1, §2, §4. 참조

149) → OSLiC, 51쪽

150) 인용한 글과 동일, wp. §15, §16. 참조

151) 인용한 글과 동일, wp. §2. 참조

152) 인용한 글과 동일, wp. §4. 참조

둘째, “라이브러리를 사용하는 저작물”은 “라이브러리 일부의 파생물을 포함하지 않지만 결합되거나 링크되어 라이브러리와 함께 작업하도록 고안된 프로그램”으로 정의한다. 반면에, “별개로” 다뤄지는 “라이브러리를 사용하는 저작물”은 명백히 “라이브러리의 파생작이 아니다”라고 정의한다.<sup>154)</sup> 셋째, (위의 섹션에 대한 예외사항으로서) LGPL-2.1에서는 “완전히 동일한 기계가 읽을 수 있는 라이브러리 소스코드를 포함한 저작물을 첨부”한다면 “스스로 선택한 조건 하에 ‘라이브러리를 사용하는 저작물’을 라이브러리와 결합하거나 링크하여 라이브러리 일부를 포함하는 저작물을 생성하여 배포”할 수 있다. 이 세 가지 설명은 전부 수정 여부에 상관없이 또한 코드를 직접적으로 공개/배포하든 별도로 라이브러리 소스코드를 수신하기 위한 제안서‘만’ 공개/배포하는 상관없이 라이브러리의 소스코드를 공개할 것을 명확히 요구하고 있다.<sup>155)</sup> 그러나 이들 설명에서는 라이브러리 또는 (OSLiC에서 언급하기 위해 사용하는 것으로서) 가장 상위의 개발을 사용하는 저작물의 소스코드를 공개/배포할 것을 요구하지 않는다.

따라서 놀랄 일은 아니지만 LGPL에서는 가장 상위의 개발을 사유화로부터 보호하지 않는 것으로 추정된다. 이것이 GNU Lesser General Public License라고 불리는 이유가 된다.

### 2.8.2 LGPL-3.0

LGPL-3.0은 GPL-3.0의 확장으로 해석되기를 원한다. 이를 위해, 여기서는 명확히 “(일부) 추가 허가조항이 보강되는 GNU General Public License 버전 3의 조항 및 조건이 포함”되었다.<sup>156)</sup> 따라서 LGPL-3.0에서는 전체 차원의 상위 개발을 다루는 부분을 제외하고 GPL-3.0의 보호 기능 대부분이 계승되었다. GPL-3.0과는 반대로 LGPL에서는 LGPL-3.0 라이선스가 적용된 라이브러리를 더 복잡한 라이브러리<sup>157)</sup>, 기능 추가 어플리케이션<sup>158)</sup>, 그리고 재구성된 라이브러리 시스템<sup>159)</sup>에 삽입할 수 있다. 또한 LGPL-3.0에서는 “(자체) 선택한 조항에 의거” 이 전체 차원의 유닛을 “전달”할 수 있다.<sup>160)</sup> 이에 따라, 이 가장 상위 개발의 소스코드 또한 공개할 의무는 없지만<sup>161)</sup> (수정되어) 삽입된 라이브러리의 소스코드를 공개할 의무가 있다.

이미 설명한 GPL-3.0의 보호 기능<sup>162)</sup>과 LGPL-3.0에 대한 추가 세부사항에 근거하여 이와 같이 LGPL-3.0의 보호 기능을 요약할 수 있다.

첫째, LGPL은 받은 소프트웨어를 사용, 수정 또는 배포할 권한의 상실로부터 사용자를 보호한다. 또한

153) Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp. §0, emphasis. Ours. 참조

154) 인용한 글과 동일, wp. §5, emphasis ours 참조. 정확하게는 LGPL에서는 라이브러리에 링크된 특정 상황 하에 저작물이 파생작이 될 수 있다고 또한 명시하고 있다. 그러나 이때, LGPL에서는 일부 추가 조건을 이행한다면 파생작이 아닌 저작물로서 ‘파생작’을 취급할 수 있다. 이 관점과 관련하여, 링크하여 비파생작이 파생작이 되는 LGPL에서 던지는 암시는 기대할 수 있는 중요한 것으로 보이지 않는다.

155) Open Source Initiative: LGPL-2.1 라이선스 (OSI), 1999, wp §6 참조.

156) Free Software Foundation: LGPL-3.0 라이선스 (FSF), 2007, wp. 참조, §0 바로 앞

157) 인용한 글과 동일, wp. §3. 참조

158) 인용한 글과 동일, wp. §4. 참조

159) 인용한 글과 동일, wp. §5. 참조

160) 인용한 글과 동일, wp. §4. 참조

161) 엄밀히 말하자면, LGPL-3.0에서는 “결합된 저작물”은 삽입된 라이브러리 신규 버전을 바탕으로 재결합할 수 있음을 보장한다. 이를 위해, 더 큰 유닛을 재링크하지 않고 삽입된 라이브러리를 교체할 수 있는 “적합한 공유된 라이브러리 메커니즘”을 사용해야 하거나 적어도 사용자가 삽입된 라이브러리 신규 버전을 바탕으로 지배적인 유닛을 다시 링크할 수 있는 바이너리 세트와 “최소한의 상응하는 소스 코드”를 공개해야 한다 (위 책과 동일 참조)

162) → OSLiC, 51쪽

특허분쟁에 대해서도 보호한다. 둘째, 피드백 손실, 보증 청구 및 특허분쟁에 대해 기여자와 배포자는 보호된다. 마지막으로 재사유화에 대해 이미 배포된 소프트웨어 자체가 보호된다.

그러나 LGPL-3.0에서는 미 배포한 소스코드가 보호되지 않고 사유화에 대해 가장 상위의 개발이 보호되지 않는다.

## 2.9 MIT 라이선스의 보호 기능

승인된 오픈소스 라이선스 중 하나인<sup>163)</sup>, MIT 라이선스<sup>164)</sup>는 사용자의 배포 받은 소스코드 또는 바이너리 복사본을 사용, 수정 또는 배포하는 권한을 보호 한다<sup>165)</sup>. 또한, MIT 라이선스는 '보증 없음 조항'이 포함되기 때문에 기여자 또는 배포자는 소프트웨어 사용자의 보증 청구로부터 보호된다.<sup>166)</sup> 마지막으로 MIT 라이선스는 소스가 폐쇄되는 라이선스 변경으로부터 기존에 배포된 소스를 보호한다. 왜냐하면, "사용, 복사, 수정, 배포하는 허가는 저작권 고지와 이 허가고지가 소프트웨어의 모든 복사본 또는 상당 부분에 포함되어야 하는 조건에 준하기" 때문이다.<sup>167)</sup>

하지만 MIT 라이선스는 특허분쟁으로부터 사용자, 기여자, 배포자를 보호하지 않는다. (왜냐하면 MIT 라이선스는 특히 조항이 포함되지 않기 때문이다.) 또한, (카피레프트가 적용되지 않기 때문에) 피드백 손실로부터 기여자를 보호할 수 없다. 또한 MIT 라이선스에선 (수정 여부에 관계없이) 재 폐쇄로부터 배포되지 않은 소프트웨어 또는 배포된 바이너리가 보호되지 않는다. 왜냐하면 소스코드를 공급하지 않고 바이너리만 재배포할 수 있기 때문이다.<sup>168)</sup> 마지막으로 MIT라이선스로 온탑 개발의 사유화로부터 보호 받을 수 없다.

## 2.10 Mozilla Public License(MPL)의 보호 기능

승인된 오픈소스 라이선스로,<sup>169)</sup> Mozilla Public License<sup>170)</sup>는 배포 받은 소스코드 또는 바이너리 복사본

163) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

164) 'MIT'는 "메사추세츠 기술 협회("Massachusetts Institute of Technology)"의 약어이다 (cite[(cf.][wp]wpMitLic2011a).

165) Open Source Initiative: 오픈소스 정의, 2012, wp 1ff. 참조

166) Open Source Initiative: MIT 라이선스; 2012 [n.y.] (URL: <http://opensource.org/licenses/mit-license.php>) - 참고자료 다운로드: 2012-08-24, wp. 참조

167) Open Source Initiative: MIT 라이선스; 2012, wp. 바이너리가 아닌 소스코드가 보호되는 이유는 BSD 라이선스의 논점에 따른다. 이 요구사항에 의해 수정된(파생) 저작물의 소스코드를 재배포할 의무는 지지 않는다. 그리고 받은 버전을 수정하고, 바이너리 형태로만 결과물을 배포하며, 개선사항을 비밀로 해둘 수 있다. 그러나 MIT "허가고지(permission note)"에는 소프트웨어의 모든 카피 또는 상당 부분이 포함되므로", 수정물의 소스코드를 배포한다면 라이선성이 유지된다.

168) 위 책과 동일 참조

169) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

170) 2012년 Mozilla Public License 2.0 (Mozilla Foundation: Mozilla Public License 2.0 (MPL-2.0); 2012 (URL: <http://www.mozilla.org/MPL/2.0/>) - 참고자료 다운로드: 2013-03-05, wp 참조)는 장기간의 "개정 프로세스"를 통해 공개되었다(Mozilla Foundation: About MPL 2.0: Revision Process and Changes FAQ: 2013 [n.y.] (URL:<http://www.mozilla.org/MPL/1.1/>) - 참고자료 다운로드: 2013-03-05, wp 참조). 이로 인해, Mozilla Public License 1.1 (Mozilla Foundation: Mozilla Public License Version 1.1: 2013 [n.y.] (URL: <http://www.mozilla.org/MPL/1.1/>) - 참고자료 다운로드: 2013-03-05, wp 참조)는 무효가 되었다. Open Source Initiative은 또한 MPL-2.0 (Open Source Initiative: MPL-2.0 라이선스 (OSI), 2013, wp 참조) 버전을

을 사용, 수정 또는 배포할 권한을 보호한다.<sup>171)</sup> 또한 그 취지와 배포된 특허조항에 입각하여<sup>172)</sup> MPL에서는 특허분쟁으로부터 사용자가 보호된다.<sup>173)</sup> 이 특허 부분 이외에도 MPL에는 추가로 “보증 제한”과 ‘법적 책임 배제’사항이 포함된다.<sup>174)</sup> 위 세 요소에서는 모두 특허분쟁과 보증 청구로부터 기여자/배포자를 보호한다. 마지막으로 MPL은 또한 배포된 코드가 재폐쇄/재사유화되는 것을 보호하고 배포자가 피드백 손실하지 않도록 보호한다. 이 라이선스에서는 “수정물을 포함하여 소스코드 형태의 소프트웨어를 모든 표준이 적용되게 배포하는 것이 라이선스 조항에 포함되어있지만”<sup>175)</sup>, 반면에 “실행 파일 형태로 배포된 소프트웨어는 반드시 소스코드 형태로 이용할 수 있어야 한다”라고 명시하고 있다.<sup>176)</sup> 따라서 MPL은 카피레프트 라이선스라고 추측된다.

그러나 MPL이 강력한 카피레프트 라이선스는 아니다. MPL은 최상위 개발의 사유화를 보호하지 않는다. 첫째, MPL에서는 파생작이라는 단어가 쓰이지 않는다.<sup>177)</sup> 대신 MPL에서는 “(초기) 소스코드 형태 그리고 수정된 소스코드 형태”를 “MPL 적용 소프트웨어(Covered Software)”라는 라벨로 표시한다.<sup>178)</sup> (“수정”이라는 용어는 “추가, 삭제, 수정이 이루어진 MPL 적용 소프트웨어 또는 라이선스 적용 소프트웨어에 추가, 삭제, 수정으로 인한 소스코드 형태의 결과물을 모두 포함한다.”<sup>179)</sup>) 둘째, MPL에서는 “복합 저작물(Larger Work)”은 “개별 파일 또는 파일들은 MPL 적용 소프트웨어가 아님”으로 명시하며 소스코드 형태 그리고 그 수정본과의 차이를 둔다.<sup>180)</sup> 마지막으로, 이 라이선스에서는 “라이선스가 적용된 소프트웨어 라이선스 요구사항에 따른다면 선택한 조항에 의거 복합 저작물을 작성 및 배포할 수 있다”라고 명시하고 있다.<sup>181)</sup> 이 설명에 따르면, MPL 라이선스가 적용된 라이브러리의 함수 호출에 따라 기능이 달라지는 추가 저작물은 확실히 파생작이지만,<sup>182)</sup> 해당 라이브러리와 추가 저작물이 여러 파일로 배포된다면 기능 추가 어플리케이션의 코드를 공개할 필요가 없으므로 MPL 취지가 담긴 복합 저작물이라고 분류된다.<sup>183)</sup> 이러한 이유로, MPL은 약한 카피레프트 라이선스이고 최상위 개발의 사유화를 보호하지 않는다.

---

주관하고 이를 Open Source Initiative 승인 라이선스(Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조)로서 실고 있으면서 MPL-1.1을 “대체 라이선스”(Open Source Initiative: 카테고리별 오픈소스 라이선스, 2013, wp 참조)로 분류하고 있다. Mozilla Foundation(모질라 재단)은 “가장 중요한 라이선스 부분(파일-레벨 카피레프트)이 MPL 2.0과 MPL 1.1(Mozilla Foundation: MPL 2.0: 개정 프로세스 및 변경, 2013, wp 참조)에서 기본적으로 동일한 두 가지 라이선스간의 차이에 대해 언급한다. MPL-1.1을 검토하면 MPL-2.0의 조건 이행이 MPL-1.1에 따라 행동하는 것을 뜻한다는 인상을 받을 수 있다. 따라서 OSLiC에서는 적어도 당분간 MPL-2.0에 중점을 둔다. 그럼에도 불구하고, 여기 이 섹션에서는 MPL-2.0과 MPL-1.1의 보호 기능과 관련하여 같다는 뜻을 나타내기 위해 공개번호 없이 일반 라벨 ‘MPL’을 사용하고자 한다.

171) Open Source Initiative: MPL-2.0 라이선스 (OSI), 2013, wp. §2.1.a. 참조

172) → OSLiC 52쪽

173) 인용한 글과 동일, wp. §2.1. 참조b, §2.3, §5.2.

174) 인용한 글과 동일, wp. §6 & §7. 참조

175) 인용한 글과 동일, wp. §3.1. 참조

176) 인용한 글과 동일, wp. §3.2. 참조

177) 인용한 글과 동일, wp 참조. MPL-1.1에서는 라이선싱 소프트웨어 문맥이 아닌 신규 “라이선스 버전”을 기술하는 문맥에서만 용어 ‘파생작’이 사용된다(Mozilla Foundation: Mozilla Public License 1.1, 2013, wp. §6.3).

178) Open Source Initiative: MPL-2.0 라이선스 (OSI), 2013, wp. §1.4 참조

179) 인용한 글과 동일, wp. §1. 참조10. Mozilla Foundation(모질라 재단)에서는 ‘파일-수준 카피레프트’라는 용어로 이 내용을 표시한다 (Mozilla Foundation: MPL 2.0: 개정 프로세스 및 변경, 2013, wp 참조).

180) Open Source Initiative: MPL-2.0 License (OSI), 2013, wp. §1.7. 참조

181) 인용한 글과 동일, wp. §3.3. 참조

182) 이는 친절한 소프트웨어 개발자가 이해할만한 용어로서 일반적 의미의 “파생작”에 기인한다(→ OSLiC, 57쪽). 그러나 MPL에서는 이 일반적 측면에 중점을 두지 않으며, 복합 저작물이라는 자체 개념을 사용한다.

183) 이는 해당 헤더 파일을 포함시켜 온탑 개발에 함수, 클래스 또는 메소드의 선언(declaration)을 통합하는 것은 “수정물”로 간주하는 파일로 “(소스코드 형태의) 부분을” 포함하는 것을 의미하는 여부에 대해 논의하고 있는지도 모른다(모질라 재단: Mozilla Public License Version 1.1, 2013, wp. §1.4 참조). 친절한 개발자의 관점에서 볼 때, 선언부의 (헤더) 파일을 포함하는 것만으로 파생작을 만들 수 있다는 주장을 하기는 어렵다. 이는 종속성을 정

## 2.11 Microsoft Public License(MS-PL)의 보호 기능

승인된 오픈소스 라이선스로서<sup>184)</sup>, Microsoft Public License는 받은 소스코드 또는 바이너리 복사본을 사용자가 사용, 수정 또는 배포할 수 있게 보호한다.<sup>185)</sup> 더 나아가 특허조항에 의해<sup>186)</sup> 이 라이선스는 특허분쟁으로부터 사용자를 보호한다.<sup>187)</sup> 또한 이 특허조항과 간결한 보증 제한에 의해 MS-PL은 특허분쟁과 보증 청구로부터 기여자와 배포자를 보호한다.<sup>188)</sup> 마지막으로, Microsoft Public License는 공개 소프트웨어를 폐쇄 소프트웨어로 바꾸는 라이선스 변경으로부터 이미 배포된 소스코드와 “소스코드 일부분”을 보호한다. 첫 번째 이유로는, “소프트웨어의 일부인 모든 저작권, 특히, 상표 및 귀속 고지를 보유해야”하고<sup>189)</sup>, 둘째 이유로는, 소스코드를 배포한다는 전제 조건 하에 “MS-PL 라이선스 전체”를 배포물에 포함시켜야하기 때문이다.<sup>190)</sup>

그러나 Microsoft Public License는 “카피레프트” 소프트웨어에 해당하지 않으므로 피드백 손실로부터 기여자를 보호하지 않는다. 또한 이 라이선스에는 소스코드 공개를 요구하는 내용이 들어있지 않다<sup>191)</sup>. 동일한 취지로, MS-PL에서는 비수정 또는 수정된 형태이든 재 폐쇄로부터 배포되지 않은 소프트웨어 또는 배포된 바이너리 파일이 보호되지 않는다. 왜냐하면 이 라이선스에서는 배포자가 수정한 소스코드에 의해 바이너리파일이 달라지더라도 소스코드를 제공하지 않고 바이너리파일을 배포 및 재배포 할 수 있기 때문이다. 마지막으로 MS-PL은 사유화로부터 최상위 개발을 보호하지 않는다.

## 2.12 Postgres License(PostgreSQL)의 보호 기능

승인된 오픈소스 라이선스<sup>192)</sup>로서, PostgreSQL 라이선스는 배포 받은 소스코드 또는 바이너리 파일 복사본을 사용자가 사용, 수정 또는 배포할 수 있게 보호한다.<sup>193)</sup> 보증 제한 내용에 의해, 또한 보증 청구

---

하게 되는 코드의 함수를 불러내는 것이다. 그러나 이 부분은 MPL의 요지가 아니다. 이 라이선스에서는 정의하는 스니펫 코드를 문맥상 재사용하는 것을 목표로 한다. 이런 이유로 선언부 헤더 파일의 부분을 문맥에 통합하는 것을 무시할 수 있다. (모질라 재단 시각에서는 이 저작물은 수정물으로 보지 않기 때문에) 그러나 당연하게도, 헤더 파일에 정의부 코드를 숨기고 자신의 컴파일로 코드를 재사용하는 경우에는 MPL의 개념을 우회할 수 있게 된다. 이는 확실히 부분 통합이며, 이로써 통합 파일은 MPL 라이선스가 적용된 초기 저작물의 수정물이 된다.

184) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

185) Open Source Initiative: Microsoft Public License (MS-PL); 2013 [n.y.] (URL: <http://opensource.org/licenses/MS-PL>) - 참고자료 다운로드: 2013-02-26, wp. §2. 참조

186) → OSLiC 53쪽

187) 인용한 글과 동일, wp. §2B, §3B 참조

188) 인용한 글과 동일, wp. §2B, §3B, §3D 참조

189) 인용한 글과 동일, wp. §3C 참조

190) 인용한 글과 동일, wp. §3D 참조

191) 인터넷 상에 오해가 있는 것 같다. 영문 위키피디아에서는 MS-PL을 permissive 라이선스로, MS-RL을 Copyleft 라이선스로 표기한다. (cf. Wikipedia (en): Shared source; n.l., 2013 [n.y.] hURL: [http://en.wikipedia.org/wiki/Shared\\_source](http://en.wikipedia.org/wiki/Shared_source) - reference download: 2013-02-26, wp). 독일 위키피디아에서는 MS-PL을 약한 copyleft 라이선스라고 표기한다. (cf. Wikipedia (de): Microsoft Public License; n.l., 2013 [n.y.] hURL: [http://de.wikipedia.org/wiki/Microsoft\\_Public\\_License](http://de.wikipedia.org/wiki/Microsoft_Public_License) - reference download: 2013-02-26, wp). MS-RL 또한 약한 copyleft 라이선스라고 표기한다. (cf. Wikipedia (de): Microsoft Reciprocal License; n.l., 2013 [n.y.] hURL: <http://de.wikipedia.org/wiki/Ms-RL> - reference download: 2013-02-26, wp). 그러나 “ifross 라이선스 센터”에 의하면, MS-RL은 약한 copyleft, 그리고 MS-PL은 선택 가능한 옵션이 있는 permissive라이선스이다. (cf. ifross: ifross Lizenz-Center, 2011, wp). 라이선스 원문 자체와 다른 자료들을 참고하여, 우리는 MS-PL을 permissive 라이선스로 표기하기로 하였다.

192) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

193) Open Source Initiative: PostgreSQL Licence (PostgreSQL); 2013 [n.y.] (URL:

드시 “모든 복사본에 표시되어야”하므로 폐쇄된 소프트웨어로 저작물이 초기화되는 라이선스 변경으로부터 이미 배포된 소스코드도 보호한다.<sup>194)</sup>

그러나 PostgreSQL 라이선스는 ‘카피레프트’ 소프트웨어가 아니므로 피드백 손실로부터 기여자를 보호하지 않는다. 또한 이 라이선스에는 소스코드 공개를 요구하는 내용이 없다. 동일한 취지에서 PostgreSQL 라이선스는 (수정여부에 관계없이) 재폐쇄로부터 배포되지 않은 소프트웨어 또는 배포된 바이너리파일을 보호하지 않는다. 이유는 이 라이선스에서는 배포자가 수정한 소스코드에 의해 바이너리파일이 달라지더라도 소스코드를 제공하지 않고 바이너리파일을 배포/재배포 할 수 있기 때문이다. 마지막으로, PostgreSQL은 사유화로부터 최상위 개발을 보호하지 않는다.

## 2.13 PHP 라이선스의 보호 기능

승인된 오픈소스 라이선스로서<sup>195)</sup>, PHP-3.0 라이선스는 받은 소스코드 또는 바이너리 파일의 복사본을 사용자가 사용, 수정 또는 배포할 수 있게 보호한다.<sup>196)</sup> 또한 보증 제한 조항에 의해 보증 청구로부터 기여자와 배포자를 보호한다.<sup>197)</sup> 마지막으로, 이 라이선스는 “소스코드 재배포 시 저작권 고지, 해당 라이선스의 조건 목록 및 권리포기사항을 보유해야”하므로 공개 소프트웨어가 폐쇄 소프트웨어로 초기화하는 라이선스 변경으로부터 기존에 배포된 소스코드를 보호한다.<sup>198)</sup>

그러나 PHP-3.0 라이선스는 “카피레프트” 소프트웨어에 해당하지 않으므로 피드백 손실로부터 기여자를 보호하지 않는다. 또한 이 라이선스에는 소스코드 공개를 요구하는 내용이 들어있지 않다. 동일한 취지에서 PHP라이선스는 비수정 또는 수정된 형태이든 간에 재폐쇄와 관련되는 배포되지 않은 소프트웨어 또는 배포된 바이너리 파일을 보호하지 않는다. 왜냐하면 이 라이선스에서는 배포자가 수정한 소스코드에 의해 바이너리 파일이 달라지더라도 소스코드를 제공하지 않고 바이너리 파일을 배포/재배포 할 수 있기 때문이다.

## 2.14 요약

이와 같은 모든 세부사항은 표<sup>199)</sup>뿐만 아니라 이 장 마지막에 나타낸 마인드맵으로도 간략하게 나타낼 수 있다. 또한 이런 세부사항을 근거로 하여 ‘사용자 보호 라이선스’와 같이<sup>200)</sup> 신규 오픈소스 라이선스 그룹, 클래스, 보다 복잡한 분류체계에 이르는 ‘특허 분쟁 보호 라이선스(patent disputes fending license)’를 만들 수 있다.

---

199) <http://opensource.org/licenses/PostgreSQL>- 참고자료 다운로드: 2013-02-27, wp. 참조

200) 위 책과 동일 참조

195) Open Source Initiative: 알파벳 순서로 분류한 오픈소스 라이선스, 2012, wp. 참조

196) Open Source Initiative: PHP License 3.0 (PHP-3.0): 2013 [n.y.] (URL: <http://opensource.org/licenses/PHP-3.0>)- 참고자료 다운로드: 2013-02-27, wp. 참조

197) 위 책과 동일 참조

198) 위 책과 동일 참조

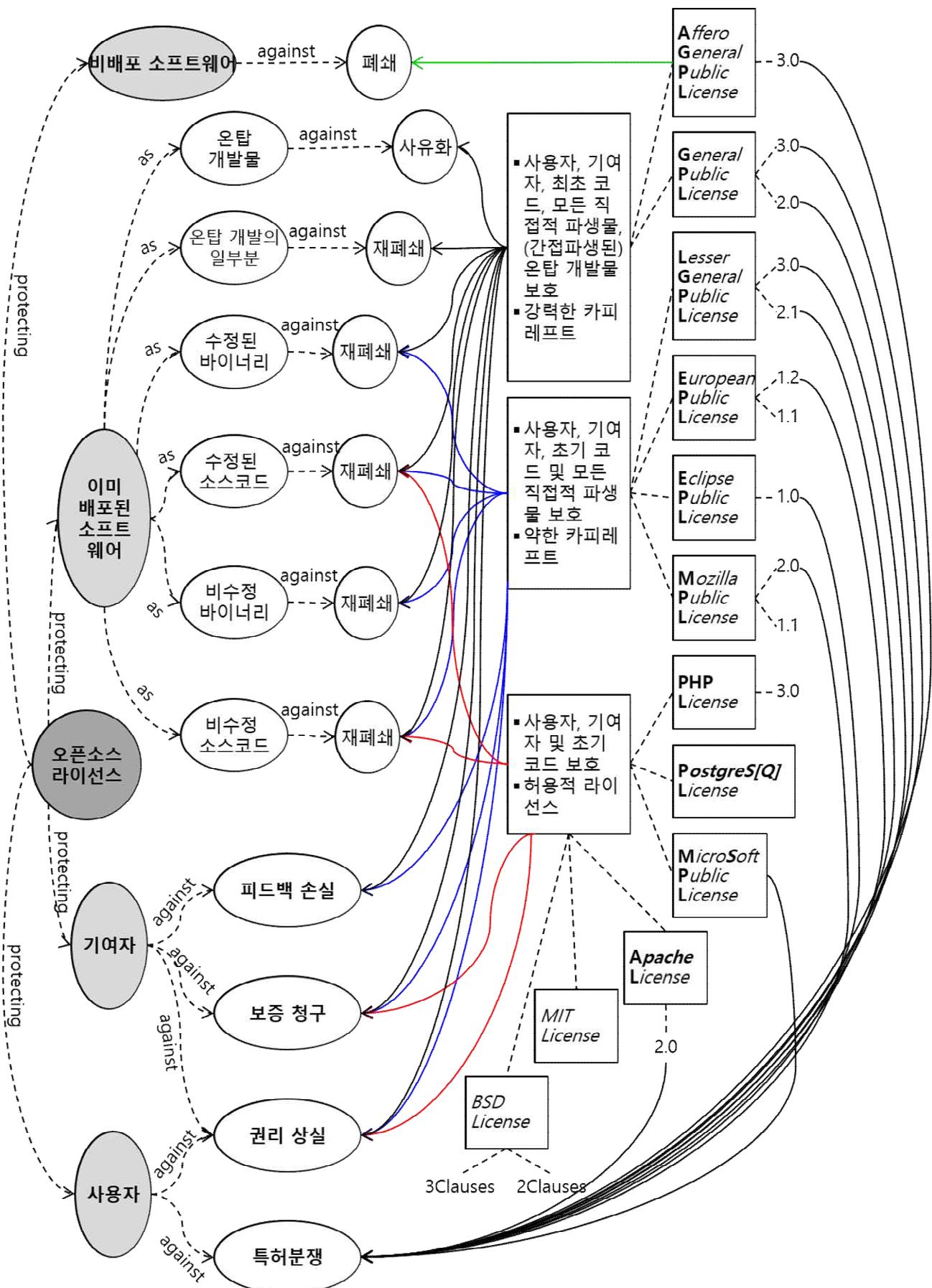
199) → OSLiC 27쪽

200) 모든 라이선스가 오픈소스 소프트웨어 정의를 충족해야 하므로 모든 라이선스를 뜻한다.

그러나 이 모든 분류에 대한 관점이 그룹의 모든 멤버가 동일한 요구사항을 충족하기 때문에 높이 평가되지 못한다는 점을 명심해야 한다. 이는 분류 기준이 성취과제를 직접 참조한다면 가능하다. 실제로 거의 모든 오픈소스 라이선스는 이 기준이 다르며, 차이가 매우 미미하더라도 이를 무시 할 수 없다<sup>201)</sup>. 따라서 가능한 오픈소스 라이선스 클래스를 반영하는 것은 오픈소스 라이선스 영역을 잘 알게 되는 좋은 방법이다. 그러나 이 방법이 소프트웨어 사용 권한을 결정하지는 못한다. 이를 위해 모든 라이선스를 개별적으로 고려해야 한다.

---

201) 전체를 대표하는 부분: BSD 라이선스와 아파치 라이선스에서는 여러분이 어플리케이션 개발자에게 표시(indication)를 제공할 것을 요구한다. 그러나 BSD 라이선스의 경우, 저작권 고지 / 문구를 발행하고 아파치 라이선스의 경우, 어플리케이션과 함께 배포된 고지 파일 내용을 정확히 제시해야 한다.



### 3 오픈소스: 일부 부작용에 대해

#### 3.1 불명확하게 특허를 공개하는 문제

이번 장에서는 오픈소스 소프트웨어를 배포하여 '암묵적 특허 사용 허용'으로도 알려진 오픈소스 라이선스의 특허조항의 효력을 일반적이지는 않지만 라이선스를 충족하는 과제와 관련하여 간략하게 분석한다.

적어도 자유 소프트웨어 운동에서는 소프트웨어 특허를 반기지 않는다.<sup>202)</sup> 그러한 태도의 잘 알려진 증거 중 하나는 GPL이다. 해당 라이선스의 서문에서는 "모든 자유 프로그램은 소프트웨어 특허에 의해 끊임 없이 위협을 받고 있다"고 주장하고 있다.<sup>203)</sup> 오픈소스 커뮤니티는 다음 세 가지 위험 요소에 대해 우려하고 있다. 첫째, 개발하지도 않은 오픈소스 소프트웨어 일부의 아이디어를 해당 특허로 등록한 후 특허 소송에 관여시켜 소프트웨어 사용을 막아 수익을 올리려 하는 것을 우려한다.<sup>204)</sup> 둘째, 오픈소스 소프트웨어 개발을 법적으로 금하게 되는 가혹한 일반 소프트웨어 특허를 우려한다.<sup>205)</sup> 셋째, 이들은 (꽤나 호의적이지 않은) 오픈소스 개발자가 오픈소스 원칙을 약화시키기 위해 특허를 등록할 수 있다는 가능성에 대해 걱정한다.<sup>206)</sup>

그러나 소프트웨어 특허에 대항하려는 시도와 관계없이 소프트웨어 특허는 실현되었다. 법을 준수하려면 특허 제약사항을 적절히 다루어야 한다. 오픈소스 라이선스들은 이 필요성을 인식하고 존중한다. 또한 이들 중 일부는 특정 특허조항<sup>207)</sup>으로 또는 라이선스 내용 속에서 여러 문장들로<sup>208)</sup> 소프트웨어 특허 효력을 다루려고 한다. 그러나 OSLiC에서 일반적인 논의에 참여를 원하지 않는다면 OSLiC는 이 주제를 왜 다루겠는가?

오픈소스 라이선스의 다른 조건과 반대로, 일반적인 특허조항 또는 제안에서 라이선스에 따른 필요한

202) 앞서 설명한 자유 소프트웨어 운동에 관한 관점에 기초한 소프트웨어 특허 효력에 관한 상세 사항은 Stallman, Richard M.: Free Software: Freedom and Cooperation(자유 소프트웨어 자유와 협력); (2001년 3월 29일에 뉴욕 대학에서 제공한 강의기록; In Stallman: Free Software, Free Society: 선정 에세이, 2002, wp.)을 참조한다. 이 강의는 1회 이상 제공되었으며, 나중에 (Stallman, Richard M.: The Danger of Software Patents(소프트웨어 특허의 위험성); (2002년 3월 25일 런던 캠프리지 대학교에서 제공한 강의기록; In Stallman: Free Software, Free Society: 선정 에세이, 2002, wp 참조)에 간행되었다. 200년대에 들어서 첫 십 년 동안 소프트웨어 특허에 대해 정치적으로 더욱 대항하는 일로 초점이 전환되었다(Stallman, Richard M.: Fighting Software Patents(소프트웨어 특허 논쟁) - Singly and Together(함께); n.st. [2004] (URL: <http://www.gnu.org/philosophy/fighting-software-patents.html>) - 참고자료 다운로드: 2013-02-18, wp 참조). 그러나 최근에는, 소프트웨어 특허를 다루는 일로 또 다시 바뀐 것으로 보인다(특허에 맞서는 것이 아니라 그 효력을 완화하는 일). 그 제안은 '일반적으로 사용되는 컴퓨팅 하드웨어에서 프로그램을 개발, 배포 또는 실행하는 것은 특허 침해가 성립되지 않는다는 점을 (입법화) 하자는' 제안이다((Stallman, Richard M.: Let's Limit the Effect of Software Patents, Since We Can't Eliminate Them; in: Wired, n.st. January (2012) (URL: <http://www.wired.com/opinion/2012/11/richard-stallman-software-patents/>) - 참고자료 다운로드: 2013-02-18, ISSN n.st., wp 참조)

203) Open Source Initiative: GPL-2.0 License (OSI), 1991, p.wp. 참조

204) Jaeger a. Metzger : 오픈소스 소프트웨어. Rechtliche Rahmenbedingungen der Freien Software, 2011, p. 234. 참조

205) 위 책과 동일 참조

206) Jaeger a. Metzger 오픈소스 소프트웨어. Rechtliche Rahmenbedingungen der Freien Software, 2011, p. 235. 참조

207) 전체를 대표하는 부분 Open Source Initiative: APL-2.0, 2004, wp §3. 참조

208) 전체를 대표하는 부분 Open Source Initiative: EPL-1.0, 2005, wp. 참조

일련의 특정 행동을 직접적으로 언급하지 않는다. 오픈소스 특허조항은 일반적으로 '행위로써 지불하는 (paying by doing)' 방식을 따르지 않는다. 따라서 실질적으로 여기에서 특허조항을 언급할 필요가 없는 것처럼 보인다.

유감스럽게도, 특허조항들은 '상황에 따른 대처'를 직접적으로 언급하지 않지만, 일부는 오픈소스 소프트웨어 배포자가 오픈소스 소프트웨어의 한 부분을 실제로 배포하고 있을 때 이미 무언가가 있음을 암시하게 하는 부작용을 유발한다. 이런 암묵적 부작용은 실리주의적인 OSLiC에서조차도 특허조항을 다루게끔 한다.

오픈소스 라이선스의 특허조항에는 두 가지 효과적인 지침이 있다. 이 조항에서는 오픈소스 소프트웨어 사용자를 보호하는 두 가지 방법을 사용하는데 어떤 경우에는 이 방법들을 조합하기도 한다.

- 첫째, 오픈소스 라이선스에서는 오픈소스 소프트웨어 각 부분의 모든 기여자와 배포자가 오픈소스 소프트웨어 자체를 사용하는 권한뿐만 아니라 소프트웨어를 법적으로 사용하는데 특허가 필요함에 따라 (기여자와 배포자가 소유한) 그 특허를 모두 사용하는 권한을 모든 사용자와 배포 받는 사람에게 저절로, 암묵적으로 허가하도록 보장할 수 있다.<sup>209)</sup> 따라서 다음 몇 페이지에 걸쳐 간단하게 이러한 라이선스를 권한부여 라이선스(granting license)라는 명칭으로 제시할 것이다.
- 둘째, 오픈소스 라이선스에서는 특히 침해와 관련하여 사용자가 기여자와 배포자에 대해 소송을 제기한다면 자동으로 소프트웨어를 사용, 수정 및 배포하는 권한을 상실 할 수 있다. 이는 초기에 부여된 권한을 취소하는 것으로 볼 수 있다. 따라서 이 라이선스를 권한취소 라이선스(revoking license)라고 한다.

이후에, 다음 분류를 입증하기 위해 OSLiC에서 논의된 모든 라이선스의 구체적 특허조항을 요약할 것이다.

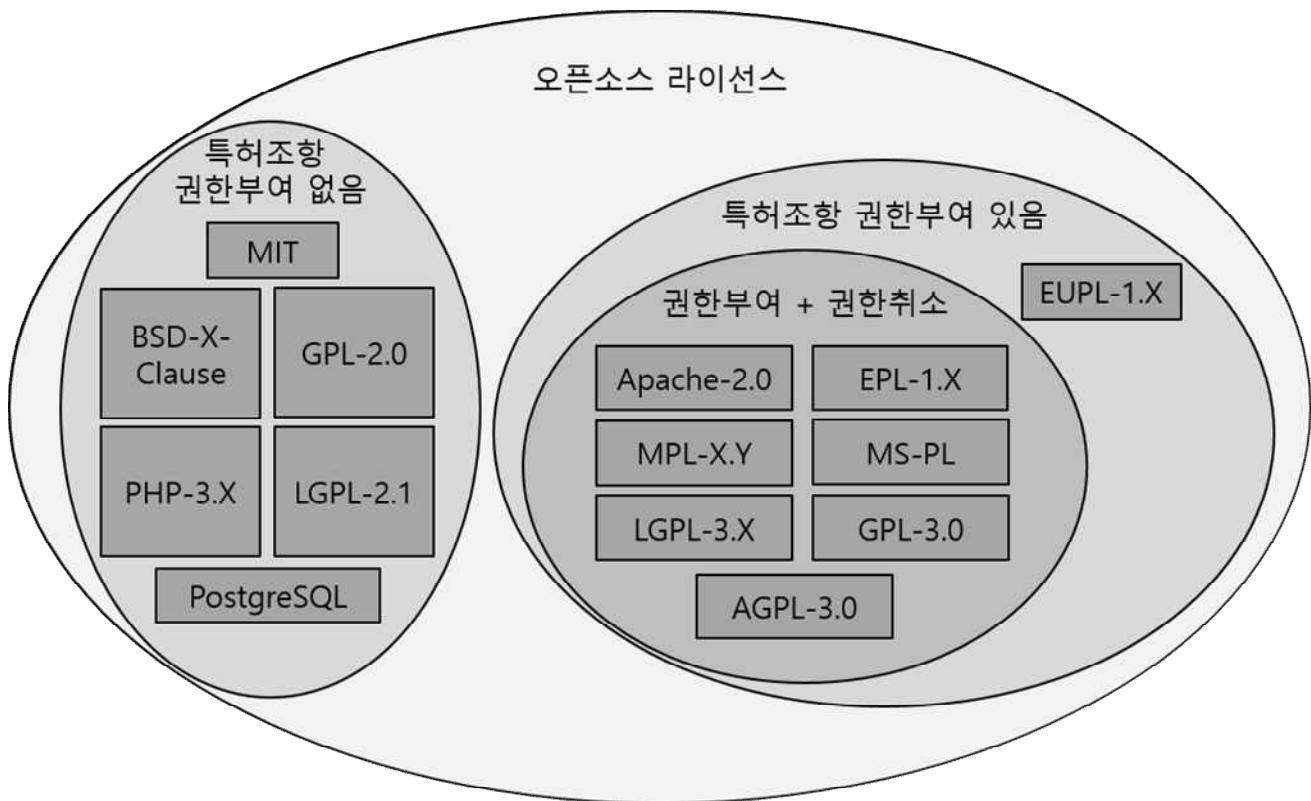
최종 문맥 형태와 상관없이, 어떠한 상황에서도 사용자가 이러한 측면을 고려하도록 하기 위해 라이선스는 권한의 부여 또는 취소에 대한 입지를 표현한다.

- 전체적으로, 사용 허가권자, 기여자, (그리고/또는) 배포자 중 어느 누구도 자기가 소유하지 않은 특허를 사용하는 권한을 양도할 수 없다는 점을 명심해야한다.(특허 조항이 있는 오픈소스로 배포하려하는 상황에라도)<sup>210)</sup> 개발에 기여하지 않았거나 배포에 관여하지 않은 제3자가 암묵적으로 무단으로 사용한 특허는 (오픈소스) 특허조항(소유권이 없으면 배포할 권한도 없다)에 기반하여 불명확하게 그리고 자동적으로도 양도될 수 없다.<sup>211)</sup> 따라서 오픈소스 라이선스가 사용자를 보호

209) 소프트웨어 개발에 기여하지 않은 배포자가 또한 필요한 특허 프트폴리오 권한을 허가해야 하는 여부는 법적 논의가 발생할 수 있다. OSLiC는 이런 논의에 참여하길 꺼려한다. 여기에서는 간단하게 실리적인 입장을 취하겠다(그런 특허조항을 포함한 오픈소스 라이선스에 따르고 있다는 점을 확실히 해두고). 만약 여러분이 이런 입장이 합당하지 않다고 본다면 소프트웨어를 법적으로 사용하는 또 다른 방법을 찾게 할 수 있는 법률 전문가에게 상담을 받는 것이 바람직할 수 있다.

210) EPL은 이런 측면이 주장되는 라이선스 중 하나이다. 특허조항 후반에서 다음과 같이 강조한다. “[…] 프로그램은 특허 또는 다른 실체의 지적재산권을 침해하지 않음을 기여자는 보장하지 못한다.” 또한 추가로 “[…] 받는 사람이 프로그램 배포에 제3자의 특허 라이선스가 필요하다면, 프로그램을 배포하기 전에 그 라이선스를 확보하는 것은 받는 사람의 책임이다”라고 명확히 하였다(Open Source Initiative: EPL-1.0, 2005, wp §2c 참조).

211) 이는 프로그래머가 가끔 고려하지 않는 중요한 측면이다. DTAG에서 우리에게 이 제약사항을 제기하고 인내심



하려 한다고 하더라도, 결과적으로 사용자는 코드가 작동하는 방식이나 소프트웨어가 사용되는 프로세스에 대해 제멋대로 손대지 않는 제3자의 특허를 침해하지 않아야 한다.<sup>212)</sup>

- 권한부여 라이선스(granting license)의 문맥에는 소프트웨어 일부분에 기여하고 배포하는 행위는 기여자, 배포자의 모든 특허가 전체 소프트웨어(다소 깊게 내포된 라이브러리를 포함) 사용에 필요한 “주어진 자유”라는 상황을 불명확하게 유발한다는 점을 고려해야 한다는 내용이 있다. 따라서 특허 포트폴리오의 핵심 특허 일부가 특허조항에 의해 피해를 받는지, 그리고 이에 따라 해당 오픈소스 소프트웨어 일부를 사용 또는 배포하지 말아야 하는지를 점검하고 싶다면 삽입된 라이브러리 역시 반드시 점검해야 한다.
- 마지막으로 권한부여 라이선스의 문맥에는 특허조항에 근거하여 ‘배포된 소프트웨어를 사용 가능하게 하기 위한 사용을 허용한다.’는 취지로 특허 사용만 허가되는지를 고려해야 한다. 특허조항은 일반적으로 특허를 양도하지 않는다. 이 때문에 오픈소스 소프트웨어에 의해 (고의적이지 않은) 특허 양도의 위협이 크지 않을 때(특허의 사용이 단지 소프트웨어의 결합에서만 부여되는 경우)도 있다. 특허 사용은 소프트웨어와 결합될 때만 권리가 주어진다. 한편 오픈소스 소프트웨어가 대규모의 프로세스에 내포 여부에 상관없이 오픈소스 소프트웨어를 사용하기 위해서는 특허 사용이 필요하기 때문에, 특허 사용 권리 얻지 못하고는 오픈소스 소프트웨어를 사용하지 못할 수 있다. 반면 오픈소스 라이선스에 근거하여 허가된 오픈소스 소프트웨어가 없이 오픈소스 라이선스

있게 설명한 특허 변호사인 Mr. Stephan Altmeyer와 유익하게 논의를 나누었다.

212) 제3자의 특허를 침해한 고의성 여부에 대한 문제는 오픈소스 소프트웨어의 취약점으로 보인다. 그러나 이는 사실이 아니다. 모든 소프트웨어는 취약점이 있다. 심지어 상업적 사용 허가권자(개발자)는 실제 소유하거나 특허를 재허가하기 위해 ‘구입’한 특허 사용을 허가하는 권리만을 갖는다. 또한 상업적 사용 허가권자는 타인의 특허를 고의적이든 고의적이지 않은 침해할 수 있다.

의 특허조항에 의해 양도된 특허를 사용할 수는 없다. 그 이유는 특허조항만이 오픈소스 소프트웨어 사용가능 여부를 나타내기 때문이다.

- 요약하자면 오픈소스 라이선스를 허가하는 것이 법적으로 소프트웨어를 사용하는데 필요한 모든 권한을 명시하지 않으면서도 저절로 허가하고 있는지 고려해야 한다. 오픈소스 기여자와 배포자는 그 사실을 알아야한다<sup>213)</sup>.
- 권한취소 라이선스와 관련하여, 특허조항에 금지로 이해될 수 있는 부정적 조건이 있는지 고려해야 한다. OSLiC는 이런 조건을 해야 과업 목록(to-do list)의 '금지' 부분에 포함시킬 것이다.
- 끝으로 취소 라이선스에 의한 취소 형태는 어떤 경우에는 소프트웨어 사용 취소를, 또 어떤 경우에는 특허 사용 취소를 나타내는 것인지를 언급해야 한다. 그러나 자비로운 오픈소스 소프트웨어 사용자의 실용적 관점에서 볼 때 두 번째 경우의 특허 취소는 암묵적으로 소프트웨어 사용 권한을 해지한다고 판단할 수 있다. 특허 사용이 소프트웨어의 일부분을 법적으로 사용하기 위해 필요하다면 역시 특허 사용 권한을 갖지 않고 소프트웨어 사용이 허용될 수는 없다(그리고 특허 사용이 소프트웨어를 사용하는데 필요하지 않다면 이 특허는 특허조항이 적용되지 않는다). 따라서 이 유형의 특허조항은 소프트웨어 사용/배포 또는 수정하는 권한을 해지하는 것으로 보인다. 이런 이유로, 기업 또는 조직뿐만 아니라 단일 사용자들은 오픈소스 소프트웨어를 확실히 법률을 준수하면서 사용하고자 한다면 그런 특허조항을 중시해야 한다.

OSLiC에서는 일반적으로 독자들이 라이선스뿐만 아니라 특허조항에 따라 이행하도록 지원하려고 한다. 따라서 여기서는 특정 특허조항의 취지를 간략하게 인용하고 개괄하겠다.

### 3.1.1 AGPL 특허 관련 조항

(preliminary text)

AGPL-3.0은 GPL-3.0으로부터 파생된 라이선스이다. 서론의 §11과 §13 문단을 제외하고 그 둘은 거의 같은 글자를 포함하고 있다.<sup>214)</sup> §13에서, AGPL은 "원격 네트워크 상호작용"(물론 그에 상응하는 소스코드의 전달)에 초점을 맞춘다는 것을 명백히 언급하고 있다. 그리고 §11에서는 AGPL은 특정 특허조항을 명시한다.(Open Source Initiative: AGPL-3.0 License (OSI), 2007, §11 그리고 §13 참조)

GPL-3.0과 같이 AGPL-3.0은 특허 분쟁으로부터 모든 라이선스 사용자(licensee)를 보호하려 한다. 이러한 보호는 다음의 3가지 절차에 의해 기술된다.

213) 개발에 어떤 것도 기여하지 않은 기여자에게도 유효한지는 논란의 여지가 있을 수 있다. 이 점이 OSLiC가 참여를 꺼리는 법적인 부분이다. 오픈소스 소프트웨어를 사용하는 신뢰성 있고 안정적인 방법만을 원하는 오픈소스 사용자의 관점에서 차이가 없음을 추정하게 된다.

214) Open Source Initiative: GNU Affero General Public License, Version 3 (AGPL-3.0); 2007 [n.y. of the html page itself] hURL: <http://opensource.org/licenses/AGPL-3.0> {reference download: 2013-04-05, 와 Open Source Initiative: The GPL-3.0 License (OSI), 2007, §1...§11 비교}

먼저, AGPL-3.0은 “각각의 기여자는 저작물을 만들고, 사용하며, 판매 제공, 판매, 내포 및 전송할 수 있는 영구적, 전 세계적, 비독점적(non-exclusive), 무료, 로열티 없는 특히 라이선스를 부여한다.”<sup>215)</sup>고 단언한다. 또한 특히 라이선스는 기여자에게 부여한 이 특히 라이선스가 제2자에 의해 간접적으로 수신되더라도 나중에 모든 버전을 받는 모든 하위 배포자에게 자동으로 적용되며 프로그램에 따라 해당 저작물이나 서비스를 받는 경우에도 적용된다.<sup>216)</sup>

두 번째로 AGPL은 “기여자”로 인한 특히 라이선스 부여를 강요하는 것뿐만 아니라 라이선스는 프로그램을 수정하지 않은 채로 배포하는 라이선스 사용자에게 다음과 같이 똑같은 것을 요구하고 있다. “단일 거래 또는 약정에 의거하여 또는 이와 관련하여, 해당 저작물을 전달하거나 전달함으로써 전파하고, 해당 대상을 받는 일부 당사자에게 특히 사용권을 부여하여 사용, 배포, 수정하도록 허가한 경우 또는 특히 라이선스를 부여한 경우 해당 대상 저작물 그리고 이를 기반으로 한 저작물의 모든 수신자에게 자동으로 연장된다.”<sup>217)</sup>

마지막으로 AGPL-3.0은 다음과 같이 명시하며 라이선스 사용자의 취소 조항을 도입한다. “특히 청구가 프로그램, 그 일부 또는 전부를 제작, 사용, 판매, 판매 제공 또는 내포함으로써 침해되었다고 주장하는 소송(교차 청구 또는 소송에서의 반소 포함)을 시작할 수 없음”<sup>218)</sup> 그리고 이 라이선스 사용자는 AGPL-3.0의 제한에 반하여 해당 저작물을 전파 또는 수정하려고 시도하는 경우 “자동으로” “그 어떤 특히 라이선스들을 포함하여” AGPL-3.0이 부여한 권리들을 상실한다.<sup>219)</sup>

이에 따라 AGPL-3.0은 GPL-3.0의 권한 부여 그리고 권한 취소 라이선스와 유사하다. 우선 법적으로 소프트웨어 사용에 필수적인 모든 기여자들의 모든 특히 사용 권한을 부여한다. 하지만 특히의 위반에 관련된 소송을 진행하는 경우, 부여된 모든 권한은 취소된다.

### 3.1.2 Apache-2.0 특허 관련 조항

“특허 라이선스 허가(Grant of Patent License)”라는 제목을 단 아파치 라이선스 2.0에는 매우 장황한 두 개의 요약된 문장으로 구성되는 특허조항이 들어있다.<sup>220)</sup> 이 특허조항 이외에 ‘특허’라는 단어는 “(소스)에서 모든 특히 유의 사항을 보유할 것을” 요구하는데 사용될 뿐이다.<sup>221)</sup>

Apache-2.0 특허조항의 핵심 메시지 하나는 “이로써 기여자는 여러분에게 저작물을 만들(었)고, 사용하며, 판매 제공, 판매, 내포 및 전송할 수 있는 영구적, 전 세계적, 비독점적(non-exclusive), 무료, 로열티 없는, 취소불능 특허 라이선스를 부여한다는” 설명이다.<sup>222)</sup>

215) Open Source Initiative: The AGPL-3.0 License, Version(AGPL-3.0); 2007, wp §11 참조

216) 위 책과 동일 참조

217) 위 책과 동일 참조

218) 인용한 글과 동일, wp §10 참조

219) 인용한 글과 동일, wp §8 참조

220) Open Source Initiative: APL-2.0, 2004, wp §3. 참조

221) 인용한 글과 동일, wp §4.3. 참조

222) Open Source Initiative: APL-2.0, 2004, wp §3 참조. “기여자(Contributor)”, “저작물(Work)” 및 “여러분(You)”에 대한 정의 §1: 기여자는 최초 사용 허가권자 및 저작물에 포함되었던 제공물의 기타 소유자를 말한다. 저작물은 형태와 관계없이 개발 과정의 결과물을 나타낸다. 여러분은 라이선시(licensee)를 뜻한다.

Apache-2.0 특허조항의 두 번째 핵심 메시지는 “여러분의 저작물이 특히 침해가 성립한다고 주장하는 실체(entity)에 대항하여 특히 소송을 제기한다면 여러분에게 부여되었던 특히 라이선스가 해지된다.”는 설명이다.<sup>223)</sup>

이 조항의 세 번째 메시지는 “라이선스는 기여자가 라이선스를 부여할 수 있는 특히 청구에만 적용할 수 있는데, 기여자의 단독 제공물 또는 제공물을 결합하여 만든 산출물이 침해받았다는 사실이 필수적이다.”는 설명이다.<sup>224)</sup>

따라서 Apache-2.0은(이 장에서 언급하기 위해 사용 중인) 허가(granting) 및 취소(revoking) 라이선스다. 우선 여러분은 소프트웨어를 법적으로 사용하는데 필요한 특히 사용을 기여자에 허가 받는다. 그러나 소프트웨어와 관련하여 여러분이 특히 침해와 관련한 소송을 제기한다면 여러분에게 허가된 권한이 취소된다.

### 3.1.3 CDDL의 특히 관련 조항

CDDL의 특허 조항은 다음과 같이 Apache License의 정신과 유사하다. 라이선스는 소프트웨어를 배포하거나 사용함으로 인해 필연적으로 침해당하는 각 기여자들의 특허에 대한 권한을 부여한다. 라이선스는 또한 기여자에 대한 소프트웨어와 관련하여 특히 소송을 제기한 사람은 부여받은 모든 권한을 상실한다. CDDL은 소송 당사자의 권리가 자동으로 즉시 취소하지 않고 60일의 유예기간을 준다는 점에서 다른 라이선스들과 차이가 있다. 이 기간 동안 청구를 철회하면, 부여된 라이선스는 종료되지 않는다.

CDDL에서 사용된 실제 표현에서 CDDL이 “초기 개발자”와 다른 “기여자”를 구분한다는 점에서 복잡해진다. “기여자”는 소프트웨어 버전을 받고 “수정 버전”을 추가하여 “기여자 버전”을 만든다. 모든 실질적인 목적을 위해 “초기 개발자”는 소프트웨어를 받지 못하고 “기여자 버전”(일반적으로 “원본 소프트웨어라고 불리는”)이 자신의 “수정본”과 일치하는 다른 제공자로 취급할 수 있다.

특히 라이선스는 “초기 개발자 부여(The Initial Developer Grant)”<sup>225)</sup> 및 “기여자 부여(Contributer Grant)”<sup>226)</sup> 조항의 (b) 조항에서 부여된다. 각 기여자는 라이선스 사용자에게 “해당 기여자가 단독으로 또는 기여자 버전과 조합하여 만든 수정본(이 수정본이란 (1) 해당 기여자에 의해 작성된 수정본 (2) 기여자 버전으로 해당 기여자가 만든 수정본 조합)을 작성, 사용 또는 판매함으로 인해 침해받은 특허 청구에 따라 라이선스 소유자에게 “전 세계적으로, 로열티 없는 비독점적 라이선스”를 부여한다.

이것은 기여자 버전에 있는 코드에 의해 침해된 특허로 특히 라이선스를 제한한다. 그리고 (d) 조항은 기여자가 만든 수정 사항이 없는 경우 “기여자 버전의 제3자 수정”<sup>227)</sup> 또는 해당 소프트웨어에 의해 야기된 “침해”를 제외시키기 위해 라이선스 부여를 더욱 더 제한한다.<sup>228)</sup> 이로 인해 기여자가 자신이 기여를

---

223) 위 책과 동일 참조

224) 위 책과 동일 참조

225) Open Source Initiative: CDDL-1.0, 2004, wp §2.1(b) 참조

226) 인용한 글과 동일, wp §2.2 참조

227) 인용한 글과 동일, wp §2.2 참조

228) 위 책과 동일 참조

한 후에 수정된 코드로 인한 특허 침해를 용인할 필요가 없으며, 특히 특허 정구 범위의 다른 모든 침해 원인을 제거하지 않고 기여자가 수정을 완전히 제거할 수는 없다. 왜냐하면 특허 라이선스가 소프트웨어의 그러한 사용으로 이이지지 않기 때문이다.

“종료(TERMINATION)”라는 제목의 섹션에는 다음과 같이 정의함으로써 특허 침해 청구에 대한 일반적인 방어가 포함되어 있다. “참여자 소프트에어가 직접 또는 간접적으로 모든 특허를 침해했다고 주장하는 참여자<sup>229)</sup>에 대하여 귀하(You)<sup>230)</sup>에 모든 권리를 직접 또는 간접적으로 부여한다는 라이선스의 섹션 2.1 및/또는 2.2에 의거하여 이에 대한 주장을 일방적으로 또는 참여자와의 서면 합의에 따라 철회하지 않는 이상 통보 없이 60일 기간이 만료되면 앞으로 그리고 자동적으로 종료된다.”

따라서 참여자는 라이선스 종료를 적극적으로 시작하는 것뿐만 아니라 참여자와 계약을 통해 사건을 해결하거나 청구를 철회하는데 60일이 소요된다.

### 3.1.4 EPL의 특허 관련 조항

Eclipse Public License에서는 다른 모든 권한과 같이 “권한 허가(Grant of Rights)”라는 제목에서 그리고 동일한 부분에 나와 있는 프로그램 사용에 필요한 특허가 다루어진다. EPL에서는 “기여자는 (받는 사람에게) 사용 허가되는 특허(Licensed Patent)에 근거하여 소스코드와 목적코드 형태로 기여자가 제공하는 제공물을 작성, 사용, 판매, 판매 제공, 내포(import) 및 전송할 수 있는 비독점적, 전 세계적, 로열티 없는 특허 라이선스를 부여한다.”라고 명시하고 있다.<sup>231)</sup> 이때 EPL에서는 다음과 같이 허가행위 허용 범위에 제한이 있다. 기여자의 하드웨어 특허와 초기 소프트웨어 조합 영역을 벗어나서 관계되는 특허는 이 공개 특허조항이 적용되지 않는다.<sup>232)</sup> 마지막으로 EPL에서는 기여자가 소유하는 제3자의 특허는 그 특허조항에 의해 공개될 수 없다는 일반적 사실을 시사하고 있다. 또한 “받는 사람에 의한 프로그램 배포 허용에 제3자 특허 라이선스가 필요하다면 프로그램을 배포하기 전에 그 라이선스를 확보하는 것은 받는 사람의 책임이다”라는 예증을 제기하였다.<sup>233)</sup>

다른 오픈소스 라이선스와 같이, EPL 역시 “받는 사람이 누군가의 프로그램이 자신의 특허를 침해한다고 다른 대상에게 제기하는 특허 소송을 시작하면 그 받는 사람에게 허가된 권한은 해지 된다.”고 말미에 제시하고 있다.<sup>234)</sup>

따라서 EPL은 이 장에서 언급한 허가 라이선스이자 취소 라이선스다. 먼저 여러분은 소프트웨어를 법적으로 사용하는데 필요한 모든 기여자의 특허 사용이 허가된다. 그러나 소프트웨어와 관련하여 특허 침해관련 소송을 시작한다면 여러분에게 허가된 권한은 취소된다.

---

229) 청구를 제기당한 “기여자” 또는 “초기 개발자”

230) 특허 침해 청구를 주장하는 당사자

231) Open Source Initiative: EPL-1.0, 2005 wp §2.b 참조

232) 위 책과 동일 참조

233) 인용한 글과 동일, wp §2.c 참조

234) 인용한 글과 동일, wp §7 참조

### 3.1.5 EUPL의 특허 관련 조항

European Union Public License에는 매우 간결한 특허조항이 들어있다. 여기서는 “사용 허가권자는 라이선스 사용자(licensee)에게 이 라이선스에 의한 저작물에 허가된 권한을 활용하는데 필요한 범위까지 사용 허가권자가 보유한 모든 특허를 로열티 없이, 비독점적으로 사용할 수 있는 권한을 허가한다.”고만 설명 한다.<sup>235)</sup> 또 이 라이선스에서는 특허에 특정한 취소 조항을 포함하는 것이 아니라 “이에 따라 허가된 모든 권한은 라이선스 사용자가 라이선스 조항을 위반 시 자동적으로 해지된다.”는 조항만을 포함한다.<sup>236)</sup> 따라서 EUPL은 이 장에서 언급 중인 취소 라이선스가 아닌 허가 라이선스다.

### 3.1.6 GPL의 특허 관련 조항

GPL 버전 2.0과 3.0이 동일한 목적을 추구하고는 있지만 문맥 및 논의 구성에 대해서는 상당한 차이점이 있다. 따라서 이 두 가지 라이선스를 별도로 다루어야 할 것이다.

#### 3.1.6.1 GPL-2.0

GPL-2.0에는 법적 특허 체계에 따라 소프트웨어 사용에 필요하고 기여자가 소유한 특허권을 허가 및 취소하게 되는 특정 특허조항이 들어있지 않다.

대신 GPL-2.0 서문에 “무료 프로그램은 소프트웨어 특허에 의해 끊임없이 위협받고 있다”와 이런 위협에 대처하기 위해 GPL 저작자는 “모든 특허는 모든 사람들의 무료 사용에 대해 사용이 허가되거나 전혀 사용이 허가되지 않아야 한다고 분명히 하였다”는 점이 주장된다.<sup>237)</sup> 그러나 이 세부설명은 특허분쟁에 대해 보호 차원에서 많은 논의가 필요한 간접 주장일 뿐이다. 그래도 이 단락 내용에서는 소프트웨어 사용자에게 필요한 특허를 사용하는 권리가 직접적으로 허가되지 않는다.

특허 문제와 관련하여 GPL-2.0은 GPL 라이선시에게 ‘부과된’ 기존의 라이선스에 의해 “이 라이선스 조건과 모순되는” 특허침해가 있어서 기존 소프트웨어 사용 권리를 포기하는 것이 제약사항을 충족하는 유일한 방법이라 하더라도 GPL-2.0 라이선시는 GPL-2.0 조건을 완전히 충족해야 한다는 사실을 설명하고 있다.<sup>238)</sup> 궁극적으로 이 라이선스에서는 원래의 저작권자가 “특허에 의해 라이브러리 배포 또는 사용을 제한하는 국가를 배제한 지리적 배포 한계를 부가”할 수 있다.<sup>239)</sup> 이 설명에 입각하여 GPL-2.0에 의해 직접적이든 간접적이든 소프트웨어 사용자에게 특허 권리가 허가된다는 점을 생각할 수 없다.

따라서 GPL-2.0은 허가 라이선스, 취소 라이선스 모두에 해당하지 않는다.

---

235) Open Source Initiative: EUPL-1.1(OSI), 2007 wp §2 마지막 참조

236) 인용한 글과 동일, wp §12 참조

237) Open Source Initiative: GPL-2.0 라이선스 (OSI), 1991 wp 서문 참조

238) 인용한 글과 동일, wp §11 참조

239) 인용한 글과 동일, wp §12 참조

### 3.1.6.2 GPL-3.0

GPL-3.0에서는 “모든 프로그램이 소프트웨어 특허에 의해 끊임없이 위협을 받는 것”이 “자유 프로그램에 적용된 특허에 의해 효과적으로 사유화될 수 있는 위험”으로 보이는 점을 유감스럽게 생각한다. 이에 따라 이 라이선스에서 특정 규정을 개괄하듯이 “GPL에서는 프로그램을 유료로 제공하는데 특허를 사용할 수 없음”을 분명히 하고 있다.<sup>240)</sup> 이 유형의 보호는 세 가지 단계로 구성된다. 첫째, GPL-3.0에서는 “기여자는 라이선스 사용자에게 그의 주요 특허 조항에 근거하여 기여자 버전의 콘텐츠를 작성, 사용, 판매, 판매 제공, 내포 및 실행, 수정 및 유포할 수 있는 비독점적, 전 세계적, 로열티 없는 특허 라이선스를 허가한다.”라고 규정하고 있다.<sup>241)</sup> 둘째, 이 라이선스에서는 기여자가 허가한 특허 라이선스는 “자동적으로” 심지어 제3자에 의해 받는 사람이 간접적으로 받고 “라이선스 적용 저작물” 또는 “그에 의한 저작물”을 받더라도 어떠한 버전의 저작물을 나중에 받는 사람까지 “확대된다.”고 정의하고 있다.<sup>242)</sup> 또 GPL-3.0에서는 배포된 소프트웨어를 사용하기 위해 필요한 특허 사용 권한을 갖지만 제3자에게 이 특허를 다시 사용 허가할 수 없는 “GPL 적용 저작물”의 배포자는 소스코드를 이용할 수 있게 하고 “권리 행사를 허용하지 않거나” “최종(downstream) 받는 사람들까지 특허 라이선스를 확대함으로써” 이 문제를 해결해야 한다고 규정하고 있다.<sup>243)</sup> 그리고 마지막으로 이 라이선스에서는 라이선스 사용자는 “프로그램 또는 그 일부를 작성, 사용, 판매, 판매 제공 또는 내포로 인해 특히 청구항이 침해되었다는 소송을 제기할 수 없고<sup>244)</sup> GPL-3.0 규정에 반하여 라이선스 적용 저작물을 유포하거나 수정하려 한다면 이 라이선스는 “특허 라이선스를 포함하여” GPL-3.0에 의해 허가된 권한을 “자동으로” 상실한다는 점을 설명하는 취소 조항을 소개하고 있다.<sup>245)</sup>

따라서 GPL-3.0은 허가 라이선스 및 취소 라이선스다. 우선, 소프트웨어를 법적으로 사용하는데 필요한 기여자의 특허를 사용하는 권한이 부여된다. 그러나 소프트웨어와 관련하여 특히 침해 관련 소송을 제기한다면 허가된 권한이 취소된다.

### 3.1.7 LGPL의 특허 관련 조항

이미 위에 언급한 바대로 LGPL 버전 2.1과 3.0은 문맥 및 논의 구성에 관해서는 심한 차이가 있다. 이에 따라, 이 라이선스들을 별도로 다루어야 한다.

#### 3.1.7.1 LGPL-2.1

GPL-2.0과 같이 LGPL-2.1에는 법적 특허 체계에 따라 소프트웨어 사용에 필요하고 기여자가 소유하는 특허를 사용하는 권리가 허가 및 취소되는 특정 특허조항이 들어 있지 않다.

---

240) Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. 서문 참조

241) 인용한 글과 동일, wp. §11. 참조

242) 위 책과 동일 참조

243) 위 책과 동일 참조

244) 인용한 글과 동일, wp. §10. 참조

245) Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. §8. 참조

그 대신 LGPL-2.1의 서문에서는 “소프트웨어 특허가 자유 프로그램의 존재를 끊임없이 위협하고 있고 위협을 해결하기 위해 LGPL의 저작자는 “라이브러리 버전을 위해 확보한 특허 라이선스는 이 라이선스에 규정된 완전한 사용 자유와 일치해야 한다는 주장을 한다.”고 언급하였다.<sup>246)</sup> 그러나 불행히도, 이 세부 사항은 특허분쟁으로부터의 보호 차원에서 많은 논의가 필요한 간접적 주장일 뿐이다. 그래도 LGPL-2.1의 이 단락에서는 필요한 특허를 사용하도록 소프트웨어 사용자에게 권한을 직접적으로 허가하지 않는다.

특허 문제에 관해서 LGPL-2.1에서는 또한 기존의 특허가 (LGPL 라이선스 사용자에게 “부과된”) 이 라이선스의 조건을 위반하는 경우라도 라이선스 사용자는 LGPL-2.1의 조건들을 완전히 충족시켜야한다고 명시한다.<sup>247)</sup> 그리고 마지막으로 이 라이선스에서는 원래 저작권자가 “특허에 의해 라이브러리 배포 또는 사용을 제한하는, 국가를 배제한 지리적 배포 한계를 부가”할 수 있다.<sup>248)</sup> 이 설명에 입각하여 LGPL에 의해 직접적이든 간접적이든 소프트웨어 사용자에게 특히 권한이 허가된다는 점을 생각할 수 없다.

따라서 LGPL-2.1은 허가 라이선스 또는 취소 라이선스에 해당하지 않는다.

### 3.1.7.2 LGPL-3.0

LGPL-3.0은 GPL-3.0이 확대된 라이선스다. “추가 정의” 부분으로 시작하기 전에 LGPL-3.0에서는 “GNU GPL-3.0의 조항 및 조건을 담고 있고 일부 “추가 허가사항”으로 GPL-3.0 내용을 “보강”한다고 명시하고 있다.<sup>249)</sup> LGPL-3.0에는 ‘특허’라는 단어를 담고 있지 않지만 GPL-3.0에서는 담고 있다.<sup>250)</sup> 따라서 LGPL-3.0은 이미 언급한 바와 같이<sup>251)</sup> 허가 라이선스 및 취소 라이선스인 GPL-3.0의 특허조항을 계승하고 있다.

### 3.1.8 MPL의 특허 관련 조항

MPL에서는 세 단락에 걸쳐서 허용되는 특허 사용과 관련한 설명을 하고 있다. 첫째, “기여자는 라이선스 사용자에게 기여자의 특허 조항에 기반하여 기여 또는 그 기여자 버전을 작성, 사용, 판매, 판매 제공, 내포 및 전송할 수 있는 전 세계적, 로열티 없는, 비독점적 라이선스를 허가한다”라고 명시하고 있다<sup>252)</sup>. 둘째, 일부 “제한사항”을 강조한다.<sup>253)</sup> 그리고 마지막으로 MPL에서는 라이선스 사용자에게 허가된 허가는 라이선시가 “기여자에 의해 기여자 버전이 직간접적으로 특허를 침해한다는 주장을 하며 누군가에게 소송을 제기”한다면 라이선시스 사용자에게 허가된 권한은 “기여자에 의해 해지된다.”는 취소 조항을 제시하고 있다.<sup>254)</sup>

---

246) Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp, 서문 참조

247) Open Source Initiative: GPL-2.0 License (OSI), 1991, wp, §11. 참조

248) 인용한 글과 동일, wp. §12. 참조

249) Open Source Initiative: LGPL-3.0 License (OSI), 2007, wp. 참조

250) Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. §11. 참조

251) → OSLiC, 51쪽

252) Open Source Initiative: MPL-2.0 License (OSI), 2013, wp. §2.1, 특허, §2.1.b. 참조

253) 인용한 글과 동일, wp. §2.1, 특허, §2.3. 참조

254) 인용한 글과 동일, wp. §5.2. 참조

따라서 MPL은 허가 라이선스 및 취소 라이선스다.

### 3.1.9 MS-PL의 특허 관련 조항

첫째, MS-PL에는 “기여자는 소프트웨어 사용자에게 사용 허가 특허에 기반하여 소프트웨어 제공물 또는 소프트웨어 제공물의 파생저작물을 작성, 사용, 판매, 판매 제공, 내포 또는 처리(dispose)할 수 있는 비 독점적, 전 세계적, 로열티 없는 라이선스를 허가한다.”는 설명을 담고 있다.<sup>255)</sup> 둘째, 이 라이선스에서는 “기여자에게 특허 소송을 제기한다면 소프트웨어에 대한 기여자에 속하는 여러분의 특허 라이선스는 자동적으로 종료된다.”라고 언급하고 있다.<sup>256)</sup>

따라서 MS-PL은 허가 라이선스 및 취소 라이선스다. 소프트웨어를 법적으로 사용하는데 필요한 기여자의 모든 특허를 사용하도록 허가받는다. 그러나 만약 소프트웨어와 관련하여 특허 침해 관련 소송을 시작한다면 여러분에게 허가된 권한은 취소된다.

## 3.2 부가사항: 링킹이 부차적 기준이 되는 이유

정적 또는 동적 링크된 소프트웨어 배포는 종종 법을 준수하기 위한 해결법이 되기도 하고 문제로서 논의 되기도 한다. 이 장에서는 이런 측면이 대부분 간과되는 이유와 파생저작물의 존재를 결정하는데 도움이 되지 않는 이유를 간략하게 논의한다.

이 문서(OSLiC)의 초기 버전 일부에서는 어플리케이션이 더 큰 유닛(larger unit)으로서 ‘구성’되었던 방식과 관련하여 해당 파인더에 의해 일부 사용사례(use cases)를 하위분류하였다. 필수 정보를 모으기 위한 이전의 형태에서 이 문서(OSLiC) 사용자는 전부 정적 및 동적 링크하여 또는 오픈소스 소프트웨어 일부를 더 큰 유닛으로 문자대로 통합하여 받은 오픈소스 소프트웨어와 다른 소프트웨어 구성요소를 결합하려 했는지 답해야만 했다. 현재, 이 질문은 완전히 삭제되었다. 저작자는 이 측면을 고려하는 것이 불필요하다는 점을 확신할 수 있다.

물론, 정적 또는 동적 링크는 라이선스 전문가에 의해 종종 심도 깊게 논의되는 편이다<sup>257)</sup>. 이는 중요한 측면으로 보인다. (향후 논의 예정)

이에 따라, 부인할 수 없는 사실 몇 가지를 거론하겠다. 이 문서(OSLiC)에서는 허용적 라이선스의 사례로서 Apache-2.0 라이선스<sup>258)</sup>, BSD-2-Clause 라이선스<sup>259)</sup>, BSD-3-Clause 라이선스<sup>260)</sup>, MIT 라이선스<sup>261)</sup>,

---

255) Open Source Initiative: MS-PL, 2013, wp §2.B. 참조

256) Open Source Initiative: MS-PL, 2013, wp §3.B. 참조

257) 2013년 암스테르담에서 개최된 European Legal and Licensing Workshop(유럽 적법성 및 라이선싱 워크숍)에서 조차도 elf 파일 링크 성격과 결과와 관련한 뛰어난 강의가 제공되었다.

258) Open Source Initiative: APL-2.0, 2004, wp. 참조

259) Open Source Initiative: BSD 2-Clause License, 2012, wp. 참조

MS-PL<sup>262)</sup>, PostgreSQL<sup>263)</sup> 및 PHP 라이선스<sup>264)</sup>를 다뤘다. 또한 약한 효력의 카피레프트 라이선스로서 EPL<sup>265)</sup>, EUPL<sup>266)</sup>, LGPL<sup>267)</sup>, MPL<sup>268)</sup>을 취급한다. 마지막으로, 효력이 강력한 카피레프트 라이선스로서 GPL<sup>269)</sup>과 AGPL<sup>270)</sup>에 대해 상세히 논의하고 있다.<sup>271)</sup>

단지 이 라이선스들 중 3가지만이 '링크(또는 그 변형(variant))'를 언급하고 있다. 쉘(일련의 html포맷 라이선스 파일에서 연산으로서 실행되는)에서 명령어 grep -i link \* | grep -v "<link \|links \|skip-link"의 사용은 AGPL-3.0, Apache-2.0, GPL-2.0, GPL-3.0, LGPL-2.1, LGPL-3.0에서는 '링킹(linking)'이란 단어의 변형(mutation)을 사용하는 것을 직접적으로 보여준다. 또한 명령어 grep -i statical \* 결과는 '정적(statical)'이란 용어의 사용만을 보여주지만 명령어 grep -i dynamical \*를 사용하는 것은 AGPL-3.0과 GPL-3.0만을 시사하는 것이다. 마지막으로, 명령어 grep -i "shared" \*(동일한 일련의 파일에서 실행되는)는 이 라이선스들에 의해서만 "공유 라이브러리(shared library)"라는 용어가 사용된다는 것을 뜻한다.

이 분석은 이미 정적 또는 동적 링크가 종종 제시되는 법 준수 행위를 위해 중요한 것은 아닐 수도 있다는 것을 나타낸다. 구체적인 설명을 검토한다면 법 준수 행위는 '조합된' 것에 의해 다소 드물게 좌우된다.

**Apache-2.0:** 아파치 라이선스의 이 버전에서는 "파생저작물(Derivative Work)"은 저작물 및 그 파생저작물의 인터페이스에 링크되는 저작물이 포함되지 않는 것으로" 설명하기 위해 단 한번 '링크'란 단어를 사용한다.<sup>272)</sup> 따라서 Apache-2.0에서는 전체적으로 링크되지 않거나 정적 또는 동적으로 링크되지 않은 파생저작물을 결정하기 위해 링크되는(being linked) 기준이 사용되지 않는다. 그러므로 Apache-2.0에 따르기 위해 속성 클래스를 완전히 무시할 수 있다.

**GPL-3.0:** GPL-3.0에서는 '링크'란 단어를 3회 사용한다. 처음에는 '상응하는 소스(Corresponding Source)'를 정의하기 위해 사용하고 있는데, 이를 "목적코드를 생성, 설치 및 실행하고 저작물을 수정하는 데 필요한 소스"로서 정의하고 있다. 또 이 정의는 "특별히 저작물을 필요로 하는 동적 링크된 서브프로그램

---

260) Open Source Initiative: BSD 3-Clause License, 2012, wp. 참조

261) Open Source Initiative: MIT License, 2012, wp. 참조

262) Open Source Initiative: MS-PL, 2013, wp. 참조

263) Open Source Initiative: PostgreSQL License, 2013, wp. 참조

264) Open Source Initiative: PHP-3.0, 2013, wp. 참조

265) Open Source Initiative: EPL-1.0, 2005, wp. 참조

266) Open Source Initiative: EUPL-1.1 (OSI), 2007, wp. 참조

267) LGPL-2.1에 관해서는 Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp. 참조. LGPL-3.0에 관해서는 Open Source Initiative: LGPL-3.0 License (OSI), 2007, wp. 참조.

268) Open Source Initiative: MPL-2.0 License (OSI), 2013, wp. 참조

269) GPL-2.0에 관해서는 Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. 참조. GPL-3.0에 관해서는 Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. 참조

270) Open Source Initiative: GNU Affero General Public License, 버전 3 (AGPL-3.0); 2007 [html 페이지 n.y.] (URL: <http://opensource.org/licenses/AGPL-3.0>) - 참고자료 다운로드: 2013-04-05, wp. 참조

271) OSLiC 딕토리 '라이선스'에서 이들 라이선스 사례에 근거한 html을 찾을 수 있다. Open Source Initiative 페이지에서 다운로드된 것이다. 후속 설명은 전부 이 파일을 나타낸다.

272) Open Source Initiative: APL-2.0, 2004, wp. §0. 참조

과 공유된 라이브러리의 소스코드"를 포함한다.<sup>273)</sup> 두 번째는 "GNU Affero General Public License 버전 3를 기반으로 사용 허가된 저작물과 대표 저작물을 결합하거나 링크하여 결합된 단일 저작물을 만들고 그에 따른 저작물을 전달"할 수 있다는 표현에서 사용되었다.<sup>274)</sup> 마지막으로는 "GNU General Public License에서는 사유 프로그램(proprietary Program)에 여러분의 프로그램을 통합시킬 수 없고", LGPL은 "서브루틴 라이브러리를 작성하였던 사용 허가권자에게 보다 나은 라이선스일 수 있으며", 특히 어플리케이션을 라이브러리와 링크할 수 있는데 더 유용한 것으로 간주할 수 있다는 표현에 사용되었다.<sup>275)</sup>

따라서 '정적 링크' 또는 '동적 링크'라는 기능은 라이선스를 충족하는 행위를 유발하는데 사용되지 않는다. "수정된 버전 배포(Conveying Modified Version)" 조건은 "초기 저작물의 수정 버전"을 나타내는 "프로그램에 근거한<sup>276)</sup> 저작물"을 말한다.<sup>277)</sup> 또한 수정자, 배포자 및 후속(subsequent) 라이선서로서 라이선시는 GPL-3.0에 의해 "사본을 소유하게 되는 자에게 [GPL-3.0 요소의 기준으로 개발되었던] 전체 저작물을 사용 허가(license)"해야 한다.<sup>278)</sup> 이 라이선스에서는 특히, 링크된 모드를 나타내지 않음으로써 이 주장에 제한을 두지 않는다. 이런 이유로 이 라이선스에 대해 소프트웨어를 법적으로 준수하면서 사용하는 방법을 결정하기 위해 소프트웨어 기능과 그 사용 및 이를 배포하는 행위를 완전히 간과할 수 있다.

**AGPL-3.0:** '동적으로(dynamically)'와 '링킹(linking)'이란 문구의 사용과 의미와 관련하여 AGPL-3.0에서는 GPL-3.0의 구조를 정확히 따른다. 이 용어는 "Corresponding Source"를 정의하는 문맥에 기인하며<sup>279)</sup>, '링크 (link)'이란 단어는 AGPL과 GPL이 적합한 라이선스라는 점을 뒷받침한다.<sup>280)</sup> 또 링크란 단어는 LGPL을 시사하는데 사용한다.<sup>281)</sup> 따라서 다시 말하지만 소프트웨어를 법적으로 준수하면서 사용하는 방법을 결정하길 원한다면 정적 또는 동적 링크 기능을 간과할 수 있다.

**GPL-2.0:** 이 라이선스에서 링크란 단어는 LGPL을 시사하는 문맥에서 기인한다.<sup>282)</sup> 또한 '정적' 및 '동적'이란 단어는 이 텍스트에서 전혀 사용되지 않는다. GPL의 카피레프트 기능은 프로그램 전체, 일부 프로그램, 파생된 프로그램 및 그들의 일부분 등에 기초한 저작물"을 나타내는 세부설명에 '만' 의존한다.<sup>283)</sup> 따라서 기존의 GPL 버전에서 조차 링크에 대한 기준은 소프트웨어를 법적으로 준수하면서 사용하기 위한 부가임무를 유발하지 않는다.

**LGPL-3.0:** 이 라이선스에서는 링킹이란 단어의 변이(variant)는 "어플리케이션과 라이브러리를 결합하거나 링크하여 생성되는 저작물"의 이름인 "결합 저작물(Combined Work)"의 개념을 정의하는데 사용된다.<sup>284)</sup>

---

273) Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. §0. 참조

274) 인용한 글과 동일, wp. §13. 참조

275) 인용한 글과 동일, wp. §13. 마지막 단락 참조

276) 인용한 글과 동일, wp. §5. 참조

277) 인용한 글과 동일, wp. §0. 참조

278) 인용한 글과 동일, wp. §5. 참조

279) Open Source Initiative: AGPL-3.0 License (OSI), 2007, wp. §0. 참조

280) 인용한 글과 동일, wp. §13. 참조

281) 인용한 글과 동일, wp. §5. 참조

282) Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. 마지막 단락 참조

283) Open Source Initiative: GPL-2.0 License (OSI), 1991, wp. §2. 참조

LGPL-3.0 말미에서는 받는 사람이 전체 생성물을 나중에 입력되는 새 라이브러리 버전과 재링크하는데 필요한 모든 자료를(지배적인 기능 추가 개발물의 목적 파일을 포함하여) 배포하거나 또는 받는 사람이 바이너리 라이브러리 파일을 교체하여 단지 라이브러리를 업데이트할 수 있도록 “적합한 공유 라이브러리 메커니즘”을 사용한다면 “자신이 선택한 조항에 근거하여 결합 저작물을 전달하는 것을” 허용한다.<sup>285)</sup> 이런 조건을 충족하려면 배포자는 동적 링크가 가능한 형태로 라이브러리와 기능 추가 개발물을 배포하거나 적어도 3년간 유효한 제안문서와 함께 정적 링크된 어플리케이션을 배포하고 사용자에게 라이브러리 및 기능 추가 개발물의 모든 목적 파일을 제공하여 사용자가 자신의 이익을 위해 어플리케이션을 재링크할 수 있게 하면 족하다.

**LGPL-2.1:** LGPL-2.1에서 다른 모든 라이선스보다 더 복잡하게 논의하더라도 해당 서문에서 기대하는 상황을 명시하고 있다(여러분이 라이브러리와 다른 코드를 링크한다면 받는 사람에게 목적 파일을 전부 제공하여 받는 사람들이 라이브러리에 변경을 가하고 다시 컴파일한 후에 라이브러리와 이 파일을 재링크할 수 있도록 해야 한다고 명시하고 있다.<sup>286)</sup> 이를 위해 LGPL-2.1에서는 만약 “정적으로 또는 공유 라이브러리를 사용하든 간에 프로그램이 라이브러리와 링크된다면 이 두 개의 결합은 법적으로 말하면 초기 라이브러리의 파생물인 결합 저작물이다”라고 시작부분에 정의하고 있다.<sup>287)</sup> “컴파일 또는 링크함으로써 라이브러리로 작업하도록 하는” “라이브러리를 사용하는 저작물은” “별개로 라이브러리의 파생저작물은 아니다.” 반면에, “라이브러리를 사용하는 저작물’을 링크하는 것은 (라이브러리 부분을 포함하므로) 라이브러리의 파생물인 실행 파일(executable)을 생성하는 것으로 정의한다.<sup>288)</sup> 그러나 예외사항으로서 LGPL-2.1에 의해 ‘라이브러리를 이용한 저작물’을 라이브러리와 결합 또는 링크하여 라이브러리 일부를 포함하는 저작물을 생성하고 선택한 조항에 근거하여 그 저작물을 배포할 수 있다. 배포자가 ‘적합한 공유 라이브러리 메커니즘’ 사용을 예상하거나 받는 사람이 전체 생성물을 나중에 입력되는 새로운 라이브러리 버전과 재링크하는데 필요한 전체 자료(지배적인 기능 추가 개발물의 목적 파일을 포함하여)를 또한 배포한다는 조건 하에 이 권한이 허가된다.<sup>289)</sup> 이들 조건을 충족하려면 배포자는 동적 링크가 가능한 형태로 라이브러리와 온탑 개발물을 배포하거나 적어도 3년간 유효한 제안문서와 함께 정적 링크된 어플리케이션을 배포하고 사용자에게 라이브러리 및 기능 추가 개발물의 모든 목적 파일을 제공하여 사용자가 자신의 이익을 위해 어플리케이션을 재링크할 수 있게 하면 족하다.

따라서 이와 같은 분석에 관련하여, 일반적으로 소프트웨어를 법적으로 준수하면서 배포하는데 필요한 과제를 달성하기 위해서 정적 또는 동적 링크된 바이너리 형태로 소프트웨어를 배포하기를 원하는지 여부를 다소 복잡하게 추정할 필요는 없다. 대신에 충분한 행위로서 알아내었던 LGPL의 작업 리스트에 이 행위들을 직접 포함시킬 수 있다. 또 LGPL의 작업 리스트에만 이 설명을 삽입하는 것으로도 충분하다. 이 논의를 일반화할 필요는 없다. 그러므로 정보 수집을 위해 제공된 형태를 단순화하여 적합한 라이선스를 충족하는 작업 리스트를 알아낼 수 있다.

284) Open Source Initiative: LGPL-3.0 License (OSI), 2007, wp. §0. 참조

285) 인용한 글과 동일, wp. §4. 참조

286) Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp. preamble 참조

287) 인용한 글과 동일, wp. 서문 참조

288) 인용한 글과 동일, wp. §5. 참조

289) Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp. §6, §6b, §6c. 참조

### 3.3 부가사항: 오픈소스의 기본 개념 - '파생저작물'이란?

이 장에서는 동적/정적 링크와 같은 기술적 측면의 파생 저작물에 대해 정의하려는 기존의 노력에 대해 간략하게 논의한다. 우리는 링크란 용어를 통해 뚜렷한 기준을 전달할 수 없음을 입증할 것이다. 1) 모듈은 단순히 압축 해제된 라이브러리다. 2) 프로그램을 실행하기 전에 모든 모듈을 정적으로 링크하는 스크립트에 의해 추가되는 모듈로서 소프트웨어를 배포할 수 있다. 3) 파이프 커뮤니케이션 기준은 양호하지만 불충분하다. 4) 이런 모든 시도는 스크립트 언어의 구성 기능과 일치하지 않는다. 그러므로 우리는 모글렌(Moglen)을 따르고 개발자 관점을 근거로 논쟁할 것이다. 그것은 라이선스에 의해 보호되는 기능 추가 개발물과 보유한 파생저작물에 의해 사용 허가되었던 코드 일부를 호출하거나(call) 이동하는(jump into) 함수, 메소드 또는 어떤 것의 문제일 뿐이다.

일단 논쟁의 개요를 제시해본다.

'파생저작물(derivative work)'의 의미를 알도록 한다! 대부분의 오픈소스 라이선스에서는 저작물 '수정' 형태로 직접적이든 간접적이든 '파생저작물'<sup>290)</sup>이란 용어를 사용한다.<sup>291)[표를 조사(調査)로 작성]</sup> 그리고 '파생저작물' 등의 용어를 사용하는 거의 모든 라이선스는 해당 라이선스에 따라 수행되어야 하는 작업을 파생저작물이 되는 전제 조건에 연결시킨다. 이런 이유로, 이 라이선스에 따르려면 '파생저작물'이 무엇인지 알아두어야 한다.

그러나 그 의미는 확고하지 않다. '파생저작물'이란 용어에 대한 여러 가지 다른 해석이 있다[차이점을 구체화하고 출처를 인용]. 따라서 파생저작물이 무엇인지 바라는 만큼 명확하지 않다.

따라서 포용적인 개발자의 관점에 근거하여 논의하겠다. 오픈소스 라이선스는 소프트웨어 개발자를 대상으로 이들의 자유를 지키면서 소프트웨어를 개발하도록 작성되었다. 그리고 이 라이선스는 또한 소프트웨어 개발자에 의해 직접 또는 적어도 이들의 도움을 받아 작성되기도 한다. 따라서 여러분은 소프트웨어 일부분이 두 가지 원칙을 기준으로 소프트웨어의 다른 부분의 '파생저작물'인 상황 하에 질문에 답할 수 있어야 한다.

- 숨겨진 이해관계나 복안을 갖고 있지 않은 자선적이고 중립적인 소프트웨어 개발자를 기준으로 논의한다.
- 의심되는 경우에는 두 부분을 소스와 파생저작물로서 서로 관계된다고 가정한다(OSLiC에서 필요 한 과제를 이행할 것을 권장하기 위해).

기본적으로 여기서 우리는 LGPL 특정 관점을 일반화한다. 그 일반화에는 다음과 같은 세 가지 용어를 사용한다.

"라이브러리"는 "어플리케이션 프로그램과 쉽게 링크되기 위해 작성된 소프트웨어 함수와 데이터의 집

290) 소스를 인용

291) 소스를 인용

합”으로 정의된다.<sup>292)</sup>

“라이브러리 기반 저작물”은 “라이브러리 또는 파생저작물”로 정의된다.<sup>293)</sup>

“라이브러리를 사용하는 저작물”은 처음에는 “라이브러리의 파생저작물은 아니라고” 정의했으며 “사용하는 라이브러리에 결합/링크되어 파생저작물이 될 수 있는” 것으로 정의된다.<sup>294)</sup>

이러한 세부설명에 따라 두 가지 방법으로 라이브러리의 파생저작물이 있을 수 있다는 결론을 내리게 된다. 첫째, 라이브러리는 라이브러리가 되는 특성을 바꾸지 않고도 자체적으로 개선될 수 있다. 이에 따른 라이브러리는 초기 라이브러리의 파생저작물이 된다. 둘째, 지배적인 프로그램(overarching program)은 라이브러리가 제공하는 함수, 메소드 또는 데이터를 호출하여 라이브러리를 사용할 수 있다. 이 경우 그 프로그램은 라이브러리에 링크되자마자 라이브러리에 기능적으로 의존하는 파생저작물이 된다.

이 관점은 일반화 될 수 있다. 스니펫(snippet), 모듈, 플러그인은 또한 개선되고 지배적인 프로그램 또는 보다 복합적인 라이브러리에 의해 사용될 수 있다. 자선적인 공정한 개발자의 관점으로 최종적으로 형성되는 관점에 근거하여 OSLiC에서는 파생저작물로 판단하기 위한 다음 규칙을 사용한다.

**복사 사례(Copy-Case)** 소스 파일의 코드 부분을 복사하여 대상 파일에 붙이기하면 소스 파일의 파생저작물이 되는 대상 파일이 만들어진다.<sup>295)</sup>

**수정 사례(Modify-Case)** 새 콘텐츠를 삽입하거나 소스 파일의 기존 콘텐츠를 삭제하면 소스 파일의 파생저작물이 되는 대상 파일이 만들어진다.

**호출 사례(Call-Case)** 소스 파일에 의해 정의되고 설정되는 함수의 호출을 대상 파일에 삽입하면 소스 파일에 의존하는 대상 파일이 만들어진다.

다음은 이 규칙들의 적용 사례들이다.

- **기존 소스 파일을 외부 텍스트로 확대하면 파생 저작물이 생성된다!** 왜? 왜냐하면 우리의 삶을 단순화하기 위해 외부 코드를 재사용 할 것이기 때문이다. [‘복사 사례’ 참조]
- **소스 파일을 줄이면 파생 저작물이 만들어진다!** 왜? 더 나은 재사용을 위해 주어진 파일을 준비 할 것이기 때문이다.[‘수정 사례’ 참조]
- **소스 파일에서 무언가를 대체하면 파생 저작물이 생성된다!** 왜? 기존 코드의 일부를 재사용하여 삶을 단순화하기 때문이다. [‘수정 사례’ 참조]

---

292) Open Source Initiative: GPL-2.1 License (OSI), 1999, wp §0. 참조

293) 위 책과 동일 참조

294) 인용한 글과 동일, wp §5. 참조

295) 주의: 이 사례는 컴파일/실행하는 동안 자동 삽입(헤더 파일, 스크립트 라이브러리) 사례와 구분해야 한다. 헤더 파일만으로 파생저작물을 발생시키지 못한다.

- 기존 소스코드에 외부 스니펫을 통합하면 파생 저작물이 생성된다! 왜? 왜냐하면 외부 스니펫과 원본 파일 모두를 재사용함으로써 삶을 단순화 할 것이기 때문이다. ['복사 사례' 및 '수정-사례' 참조]
- 함수/메소드를 독립적으로(자주적으로) 추출하여 주어진 저작물을 리팩토링하면 두 가지 측면에서 파생 저작물이 만들어진다! 왜? 첫째, 모든 수정/생성된 파일은 원본 파일에 의존하기 때문이다. 둘째, 파일의 함수 호출은 함수 자체를 정의해야하기 때문이다. ['수정-사례' 및 '호출-사례' 참조]
- 정의 모듈이 제공하는 함수를 호출하면 호출하는 파일은 제공 모듈의 파생 저작물이 될 수 있다! 왜? 이미 준비된 작업(다른 개발자가 자주 사용하지 않는 작업)을 재사용하여 일상생활을 단순화하기 때문이다.[호출 사례' 참조]
- 정의 라이브러리에 의해 제공되는 요소를 호출함으로써 호출하는 파일은 제공 라이브러리의 파생 저작물이 된다! 왜? 이미 준비된 작업을 재사용하여 일상생활을 단순화하기 때문에 (다른 개발자가 자주 사용). ['호출 사례' 참조]<sup>296)</sup>

그리고 다음은 앞으로 논의할 수 있는 몇 가지 추가 '아이디어'이다.

- 플러그인은 프레임워크에 의존하는가? 아니다. 왜? 모듈과 같기 때문에 함수를 제공하기 때문이다.(일반적으로 프레임워크 자체에서 제공하는 함수를 사용하지 않고)
- 프레임워크가 플러그인에 종속되어 있는가? 대답자하면, 때로는 그렇고 때로는 그렇지 않다. 왜? 프레임워크가 플러그인을 잃어버렸을 때 충돌을 일으킨다면 분명히 플러그인에 의존하는 것이다. 그것은 단순히 자율적이지 않은 것이다. 그러나 충돌이 발생하지 않는다면 완벽하게 설계된 것이다. 플러그인에 의해 채워질 수 있는 장소가 아니라서 그런 것은 아니다. 이러한 종류의 프레임워크는 처리해야할 포트와 다른 응용 프로그램에서 제공하는 데이터를 얻기 위해 포트에 나열하는 응용 프로그램과 같다.
- 프로세스 간 통신을 사용하는 프로그램이 IO-파트너에 의존하는가? 절대 그렇지 않다! 왜? 그렇지 않으면, 우리는 이 모든 경우에 대해 논할 필요가 있기 때문이다. 모든 것은 각각의 실행중인 시스템에 의존하고 있다.

[계속 논의 예정]

### 3.4 부가사항: 리버스 엔지니어링과 오픈소스

의심의 여지없이 LGPL은 리버스 엔지니어링이 어떤 면에서든 LGPL이 적용된 소프트웨어를 사용하고 배포하도록 허용되어야함을 나타내는 리버스 엔지니어링을 문자 그대로<sup>298)</sup> 언급하고 있다:

"[...] 라이브러리의 일부를 포함하는 저작물을 생성하고 그 저작물을 임의로 선택한 규정에 따라 배포.

---

296) 이런 맥락에서 당신은 가끔 (예를 들어 C 코드)와 선언적 파일(예를 들어 C 헤더)을 구분해야한다는 것을 읽을 수도 있다. 그러나 우리가 보기기에, 그러한 차이를 만드는 것은 그리 중요하지 않다. 선언 헤더 파일을 포함하는 사용 파일은 정의 소스 코드 파일('호출 사례')에 의존한다. 그래서 선언하는 헤더 파일('복사-사례')에 대한 공식적인 의존성을 무시할 수 있다.

298) LGPL-v2에 대해서는 다음을 참고 cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp., §6 LGPL-v3에 대해서는 다음을 참고 cf. Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp., §4

단, 배포 규정에서 [...] 코드 역분석(reverse engineering)을 허용할 경우 [...]”<sup>299)</sup>

이러한 조항을 다루기 위한 세 가지 전략이 있는데 하나는 이것이 의미하는 바를 완전히 존중하거나, 그 의미를 완화시키거나 또는 이 요구사항을 모두 논의하는 것을 회피하는 방법이 있다.

잘 알려진 오픈소스 전문가의 첫 번째 그룹은 LGPL 라이선스가 적용된 구성요소를 해당 제품에 포함시키면 전체 소프트웨어 제품의 리버스 엔지니어링을 허용해야 한다는 LGPL-v2의 조항을 엄격하게 받아들였다.<sup>300)</sup>

두 번째로 잘 알려지고 식견이 있는 오픈소스 전문가 그룹은 실제로 LGPL-v2가 문자 그대로 엄격한 규칙을 포함하고 있음을 나타냈다. 그러나 이 규칙을 실제로 의미가 없는데, 예를 들어 이 전문가들 중 두 명은 “라이선스 조합에 대한 이러한 요구사항은 전반적인 작업에서 이러한 수정사항을 디버깅하기 위해 사용자의 수정 및 리버스 엔지니어링을 금지한다”고 설명했다. 그러나 이 제약에 대해 “시행 기록에 따르면 선택한 라이선스 조항이 결합물에 포함된 LGPL 코드의 수정에 대한 디버깅을 위한 수정이나 리버스 엔지니어링을 금지하지 않을 수 있음을 의미한다”라는 제한을 직접적으로 추가하였다.<sup>301)</sup>

---

299) cf. Open Source Initiative: The GPL-2.1 License (OSI), 1999, wp, §6. LGPL-v2는 “조항에 의해 배포된” 라이브러리를 나타내기 위해 “Library”라는 대문자를 사용한다. (cf. id., l.c., wp, §). LGPL 관련 장에서는 이 해석을 따라 기술하였다.

300) 예를 들어, 매우 신뢰할만한 독일인 전문가는 LGPL-2.1이 일반적으로 LGPL-2.1 라이선스가 적용된 라이브러리에 액세스하는 프로그램의 배포자에게 고객에게 액세스하는 프로그램을 수정할 수 있는 권리와 그에 따른 리버스 엔지니어링 수행 권한을 부여해야 한다고 규정한다. 문자 그대로 독일어 텍스트는 다음과 같이 말하고 있다:

“Ziffer 6 GPLv2.1 knüpft die Erlaubnis, das zugreifende Programm unter beliebigen Lizenzbestimmungen verbreiten zu dürfen, an eine Reihe von Verpflichtungen, die in der Praxis oft übersehen werden: Zunächst muss dem Kunden, dem die Software geliefert wird, die Veränderung des zugreifenden Programms gestattet werden und zu diesem Zweck auch ein Reverse Engineering zur Fehlerbehebung. Dies darfte alle Formen des Debugging und das Dekompilieren des zugreifenden Programms umfassen.” (cf. Jaeger a. Metzger : Open Source Software. Rechtliche Rahmenbedingungen der Freien Software, 2011, pp. 81; emphasis KR).

언뜻 보면 “copyleft.org”가 - “[중략] 모든 형태의 카피레프트 라이선스에 관한 유용한 정보, 학습서 자료 및 새로운 정책에 대한 아이디어를 작성하고 보급하기 위한 협업 프로젝트” (cf. copyleft.org: What is copyleft.org; n.l, 2014 <URL:<http://copyleft.org/>> - 참조 다운로드: 2014-12-15, wp.) 이러한 진지한 태도에 대한 또 다른 증거로 간주 될 수 있다. 그 기여자 중 일부는 “컴플라이언스의 특별 주제”를 다루는 장에서 “전체 저작물의 라이선스가 라이브러리에 일련의 수정을 디버깅하기 위한 리버스 엔지니어링을 허용해야 하며” 따라서 “응용프로그램에 적용된 EULA가 이 사용권한고가 상충되지 않도록 주의해야 한다”라고 설명한다. (cf. Kuhn, Bradley M. et.al.: Copyleft and the GNU General Public License: A Comprehensive Tutorial and Guide; n.l, 2014 <URL: <http://copyleft.org/guide/comprehensive-gpl-guide.pdf>> - 참조 다운로드: 2014-12-15, p.86

301) cf. Moglen, Even a. Mish Choudhary: Software Freedom Law Center Guide to GPL Compliance, 2nd Edition: 2014 <URL: [https://www.softwarefreedom.org/resources/2014/SFLC-Guide\\_to\\_GPL\\_Compliance\\_2d\\_ed.html](https://www.softwarefreedom.org/resources/2014/SFLC-Guide_to_GPL_Compliance_2d_ed.html)> - 참고 다운로드: 2014-12-15, wp., GPLv2.1장, 6절. 이와 같은 완화는 copyleft.org의 튜토리얼에서도 찾을 수 있다. LGPL-v2 문장을 엄격한 규칙으로 요약한 후에 다음과 같이 계속하여 기술하고 있다 “(중략) 더 큰 비 LGPL 산출물의 허가된 컴포넌트의 교체를 금지하는 허가된 저작물을 기반으로 하는 이에 대한 라이선스 조항을 억제하거나 LGPL 컴포넌트의 버그를 개선하거나 수정하기 위해 디컴파일이나 리버스 엔지니어링을 금지해야 한다.” (cf. Kuhn et.al.: Copyleft and the GNU General Public License, 2014, p.86). 이 추가사양은 라이브러리의 수정을 용이하게 해야하며 임베디드 라이브러리를 개선 할 수 있는 다른 방법이 있는 한 리버스 엔지니어링을 무시할 수 있음을 나타냅니다.

마지막으로 세 번째 전문가 그룹은 비록 라이선스에 리버스 엔지니어링에 대해 명문화되어 언급되어 있고 GPL이나 LGPL라이선스를 사용하는 방법을 설명하기를 원하지만 이 기술에 대해 논의하지 않으려고 한다.<sup>302)</sup>

이러한 상황은 오픈소스 소프트웨어를 따르며 사용하고자 하는 기업 또는 사람들과 가이드를 찾고자 하는 사람들을 괴롭힐 것이다. 특히 그들 자신의 비즈니스 관련 소프트웨어를 보호하고자 하는 사람들을 괴롭혀 댈 것이다. 결국 그들은 이 문장이 오픈소스 커뮤니티 자체에 의해 지속적으로 이해되지 않는다는 것을 고려할 수도 있다. 그리고 우리가 알다시피 이러한 기업들 중 몇몇은 개발자가 비즈니스 관련 기술을 포함하는 프로그램에 LGPL 라이선스가 적용된 컴포넌트를 내장하는 것을 예방적으로 금지한다. 불행히도, 이 결과는 잘 만들어진 큰 규모의 자유 소프트웨어에 대한 접근을 방해할 뿐만 아니라 이러한 금지령을 지속적으로 지키는 것은 거의 불가능하다. GNU/Linux 시스템의 커널과 통신할 수 있게 해주는 glibc는<sup>303)</sup> LGPL이 적용되어 있다.<sup>304)</sup> 따라서 이 라이브러리는 GNU/Linux 시스템에서 실행되는 프로그램과 간접적으로 링크되거나 결합된다. 따라서 만약 LGPL-v2가 모든 LGPL 라이브러리의 일부를 포함하는 모든 프로그램의 리버스 엔지니어링 허용을 요구했을 경우, 모든 GNU/Linux 사용자들은 GNU/Linux 상에서 실행되는 모든 프로그램을 리버스 엔지니어링으로 시험해볼 수 있어야 하며 이는 단순히 모든 'GNU/Linux 프로그램'이 glibc에 링크 또는 결합되어 있기 때문이다.<sup>305)</sup> 즉, LGPL이 실제로 리버스 엔지니어링 허가를 요구한다면 GNU/Linux에서 실행되는 모든 프로그램이 리버스 엔지니어링 될 수도 있다는 뜻이다.

하지만 LGPL-v2에 대한 철저한 분석 결과에 따르면 LGPL에 대한 또 다른 유효하고 더 유동적인 이해가 있어야만 한다. 서문에서는 GPL 외에 또 다른 약한 라이선스가 제공되는 이유를 설명하고 있다. 서문에서는 다음과 같이 말하고 있다. "(중략) 드문 경우지만 특정 라이브러리의 최대한의 사용을 장려하여 사실상의 표준이 되도록 하는 특별한 요구가 있을 수 있다." 그래서 이러한 프로그램이 자유 소프트웨어가 되도록 강요하지 않고도 "비 자유 프로그램 (중략) 라이브러리를 사용하도록 허용 (중략)" 하는 것이 전략적으로 필요할 수 있다.<sup>306)</sup>

따라서 만약 LGPL이 LGPL 라이브러리와 연결되거나 결합되는 모든 프로그램에 대해 리버스 엔지니어링을 허용할 경우 LGPL은 원래의 의도와는 반대되는 결과를 가져오게 된다. '무명에 의한 보안'과 같은 것이

302) Terry J. llardi의 기고문이 세 번째 전략의 첫 번째 증거로 간주될 수 있다. 그는 LGPL의 본질을 깊이있게 설명하며 특히 §6에 대해 논하며 “프로그램을 독점적으로 유지하기를 원한다면 LGPL (중략) 코드에 정적으로 링크하지 마라”와 같은 적용 가능한 규칙을 전달한다. 하지만 그는 리버스 엔지니어링에 대해서 다루지는 않는다. (cf llardi, Terry K.: Common OSS License Problems: n.l., 2010 <URL: [http://www2.aipla.org/html/spring/2010/papers/llardi\\_Paper.pdf](http://www2.aipla.org/html/spring/2010/papers/llardi_Paper.pdf)> - reference download: 2014-12-16, pp.5f). Rosen의 비슷한 주장 (cf. Rosen: Open Source Licensing, 2005, pp. 121ff). 또한 - 리버스 엔지니어링에 대한 의견에도 불구하고 컴프라이언스에서의 특별 주제 장에서 - copyleft.org의 문서는 이러한 태도의 사례로 간주될 수도 있다. 저자들이 LGPL 2.1 컴플라이언스에 대한 적절한 수행을 위해 10장을 학습하기를 추천하고 있다. (cf. Kuhn et.al.: Copyleft and the GNU General Public License, 2014, p.86). §6에서 GPLv2.1에 대해 깊이있게 설명하고 있지만 “Lesser GPL”的 의미에 초점을 둔 10장에서 리버스 엔지니어링을 다루고 있지는 않다. (cf. id., l.c., pp.56ff, esp.60f).

303) cf. <http://www.gnu.org/software/libc/>

304) cf. [http://en.wikipedia.org/wiki/GNU\\_C\\_Library](http://en.wikipedia.org/wiki/GNU_C_Library)

305) 이 결론은 놀라울 수도 있다. 그러나 GPL에 따라 모든 프로그램이 라이선스 되었을 것이라는 결론과 정확히 같은 주장으로 추론된다. copyleft.org의 문서는 이와 같은 논증을 아주 자세히 설명하고 있다. (cf. Kuhn et.al.: Copyleft and the GNU General Public License, 2014, pp.56f).

306) cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp, §preamble.

도입되었을 것이다. 첫째, LGPL은 라이브러리를 사용하는 비 자유 프로그램의 코드를 비밀로 유지할 수 있게 함으로써 저작물 내부를 조사로부터 보호할 수 있다.<sup>307)</sup> 하지만 결국 LGPL은 사용자가 받은 바이너리의 리버스 엔지니어링 허용함으로써 모든 내부를 확인할 수 있게 할 것이다.<sup>308)</sup> 따라서 LGPL-v2는 결과적으로 '작성자들이 의도한 존재의 이유'를 훼손하게 될 것이다. 그러한 상황에서는 어떠한 LGPL 라이브러리도 사실상 표준이 될 가능성에 희박했을 것이다.

우리는 GNU 라이선스 및 GNU 소프트웨어의 작성자들은 매우 정교한 전문가라는 것을 알고 있다. 그들은 결코 이렇게 일관성 없는 문서를 공개하지 않을 것이다. 따라서 문서상의 이러한 이견은 라이선스를 이해하는 더 좋은 방법이 있어야 한다는 사실에 대한 강력한 지표이다. 따라서 이에 대한 더욱 적절한 해석을 설명해야 할지는 추종자들인 우리에게 달렸다. 물론 그러한 해석은 서면 텍스트에 근거해야 한다. 당연히 우리 학자들은 개인적인 소망을 추가할 수 없으며 라이선스를 매우 엄격하게 읽어야만 한다. 해석은 명시적이고 합리적이도록 라이선스 문구 자체와 일치시킴으로써 '이해'를 추론해야 한다.

LGPL에 대한 더 나은 이해가 존재할 수 있고 다른 전략들과는 반대로, 우리는 리버스 엔지니어링을 허용하지 않고 오픈소스 기반 소프트웨어를 정당히 배포할 수 있는 유효한 방법이 있음을 증명할 것이다. 오픈소스 라이브러리를 사용하는 저작물이 동적으로 링크 가능한 파일 형태로 배포되는 경우 주요 오픈 소스 라이선스<sup>309)</sup> 중 어느 것도 리버스 엔지니어링이 필요 없음을 증명하고자 한다. 특히 LGPL라이브러리를 사용하여 저작물을 동적으로 링크 가능한 것으로 배포하게 되는 경우 리버스 엔지니어링을 허용하지 않아도 된다는 사실을 증명할 것이다. 그리고 다른 경우에 오픈소스 라이브러리를 사용하여 저작물을 배포하는 경우 리버스 엔지니어링을 암묵적으로 허용했음을 보여줄 수 있다. 특히 우리는 허용적 라이선스 하에 허가된 라이브러리를 사용하여 저작물을 배포하는 경우에도 암묵적으로 허용된 것을 두려워해야 한다는 것을 증명하고자 한다.<sup>310)</sup>

전반적으로 라이선스 원문 자체를 기반으로 한 우리의 분석이 기업과 사람들이 오픈소스 소프트웨어를 보다 자주 그리고 거리낌 없이 사용하도록 지원하는 일이기를 희망한다.

---

307) 약한 Copyleft는 라이브러리의 최대한의 사용을 장려하기 위해 도입되었다.

308) 그것은 모호함에 의해 보안이 나타나는 것처럼 단지 조금 더 많은 비용이 들게 될 것이다.

309) OSLiC와 마찬가지로 이 부분은 가장 중요한 오픈소스 라이선스들에만 초점을 맞추고 있다. (cf. <https://www.blackducksoftware.com/resources/date/top-20-open-source-licenses> wp.) the Apache license (cf. Open Source Initiative: APL-2.0, 2004, wp.), the BSD licenses (cf. Open Source Initiative: The BSD 3-Clause License, 2012, wp. and cf. Open Source Initiative: The BSD 2-Clause License, 2012, wp.), the MIT license (cf. Open Source Initiative: The MIT License, 2012, wp.), the MS-PL (cf. Open Source Initiative: MS-PL, 2013, wp.), the PostgreSQL (cf. Open Source Initiative: PostgreSQL License, 2013, wp.), the PHP license (cf. Open Source Initiative: PHP-3.0, 2013, wp.), the EPL (cf. Open Source Initiative: EPL-1.0, 2005, wp.), the EUPL (cf. Open Source Initiative: EUPL-1.1 (OSI), 2007, wp.), the MPL (cf. Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp.), the LGPLs (cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. and cf. Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp.), the GPLs (cf. Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. and cf. Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp.) and the AGPL (cf. Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp.)

310) 우리의 분석은 또한 LGPL이 "범용 소프트웨어 라이선스에 있어서는 안된다"는 "기술의 흡 잡을 데 없는 미로"를 포함하는 '중독' 라이선스와 같은 것이 아니라는 증거를 제공해야 한다. (cf. Rosen: Open Source Licensing, 2005, p. 124). 오늘날 후예들이 직면한 도전은 컴퓨팅 언어 C가 유발할 수 있는 모든 번거로움을 포함하여GNU 라이선스들의 옛 발명가들과 컴퓨팅에 대해 생각하는 그들의 방식을 이해하는 것이다.

그러므로 우리의 입장을 '상향식'으로 증명하자. 먼저 LGPL-v2에 대해 사실임을 보여주자 – 올바른 이해를 추론하기 전에 라이선스 문구를 언어적, 논리적 그리고 실험적으로 설명하라. 그런 다음 LGPL-v3에서도 마찬가지라는 것을 보여주고 결국 모든 다른 라이선스들에 대해서도 그것이 사실임을 보여주자.<sup>311)</sup>

### 3.4.1 LGPL-2의 리버스 엔지니어링

LGPL-2.1 리버스 엔지니어링의 문제에 대해 다루는 한 문장을 포함한다.

“라이브러리를 사용하는 저작물’을 라이브러리와 결합하거나 링크로 연결하여 라이브러리의 일부를 포함하는 저작물을 생성하고 그 저작물을 임의로 선택한 규정에 따라 배포할 수 있다. 단, 이 경우 배포규정에는 수취인들이 필요에 따라 저작물을 수정할 수 있으며 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다는 내용이 포함되어야 한다.”<sup>312)</sup>

이후로 이 문장은 LGPL2-RefEng 문장이라 지칭한다.

#### 3.4.1.1 언어적 해석

우리의 규칙을 충족하고, 문구를 엄격하게 해석하고 적절한 해석을 추론하기 위해 먼저 구문 접속사에 대해 설명하겠다.

“라이브러리를 사용하는 저작물’을 라이브러리와 결합하거나(or) 링크로 연결하여 라이브러리의 일부를 포함하는 저작물을 생성하고(and) 그 저작물을 임의로 선택한 규정에 따라 배포할 수 있다. 단(provided that), 이 경우 배포규정에는 수취인들이 필요에 따라 저작물을 수정할 수 있으며(and) 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다는 내용이 포함되어야 한다.”<sup>313)</sup>

‘단(provided that)’이 문장을 두 개로 나누고 있다는 것이 명백하다. 특정 상황이 만족되어야만 무언가가 허용된다. 또한, 문장의 두 부분 – 접속사인 ‘단(provided that)’이전 부분과 이후 부분 – 은 의미를 함축적으로 내포하고 있다.<sup>314)</sup> 이러한 구문 상호 연결은 해체되어야 한다.

먼저 “단(provided that)”접속사 앞 문장에 내포되는 의미를 해석해보자. 다른 두 접속사 *and*와 *or*을 사용하여 설립되어 있으며 *you may (X or Y and Z)* 문구 이후에 배치되어 있다. 원칙적으로 보면, *you may (X or Y and Z)*라는 문구는 *you may ((X or Y) and Z)*로도, *you may (X or (Y and Z))*로도 해석이 가능하다.

---

311) OSLiC에 의해 분석되었다: -> 60쪽

312) Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp., §6 참조. 이 문장 앞에 “위의 조항들에 대한 예외적경우로”라고 명시되어 있으며, “결합하거나 링크”앞에 “또한”이라는 문구가 포함되어 있다. 이 문구들이 뜻하는 것은, LGPL은 §6-배포방식을 라이브러리의 일반적인 배포방식과는 다른, 예외적 방식이라는 것이다.

313) 위와 동일

314) [http://en.wikipedia.org/wiki/Ellipsis\\_%28linguistics%29](http://en.wikipedia.org/wiki/Ellipsis_%28linguistics%29), wp 참조

그러나 의미적으로 보면, "(무언가를) 결합하거나(or) 링크로 연결하여 라이브러리의 일부를 포함하는 저작물을 생성"한다는 것은 "라이브러리의 일부를 포함하는 저작물을 생성"하기 위하여 컴포넌트를 연결하기 위한 두 가지 다른 방법을 의미하는 것을 알 수 있다. 이에 따라 잠시<sup>315)</sup> "결합하거나 링크로 연결"을 "\*연결"로 대체해보자.<sup>316)</sup> 이렇게 하면 문장의 구조가 *you may (W and Z)*로 축약될 수 있다 (*W*는 (*X or Y*)를 의미함).

이제, *you may (W and Z)*문구 자체에 (*you may W*)와 (*you may Z*)이 암시되어 있다고 말할 수 있다.

마지막으로, 접속사 '*provided that*' 앞에 오는 문구가 숨은 의미를 내포하고 있다. 이에 따르면 컴포넌트를 \*연결하여 "라이브러리의 일부를 포함하는 저작물을 생성하고 그 저작물을 임의로 선택한 규정에 따라 배포할 수 있다."<sup>317)</sup> 영어 문법에 따라, 두 번째 문장에서 나오는 저작물 (*that work*)는 이전 문장에서 나온 라이브러리의 일부를 포함하는 저작물을 뜻한다고 결론지을 수 있다. 전체 문장이 명확하게 정의되었으면, 영어 문법에서는 그 다음에 문장이 나올 때 이를 축약하되 의미는 원래대로 유지하는 것을 허용한다. 따라서 역으로, 축약된 문장을 다시 풀어써서 다음과 같이 전체 문장을 만들 수 있다.

즉, 접속사 '*provided that*' 이전의 문장을 구조 (*you may W*)와 (*you may Z*)를 포함하는 다음 문장으로 이해할 수 있다.

(("라이브러리를 사용하는 저작물을"을 \*연결하여 라이브러리의 일부를 포함하는 저작물을 생성하고) 또한 (라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다.)) 단(**provided that**) [ . . . ]

이론상으로는, 독자는 우리의 LGPL2-RefEng 문장의 해석을 거절할 수 있다. 그러나 우리 해석을 거절하려면 다른 해석을 적절하게 제시해야 한다. 현재로서의 우리 해석이 영어 문법에 따라 가장 적합한 해석인 것으로 보인다.

다음으로 "단(**provided that**)"접속사 뒤에 오는 문장의 의미를 해석해보자. 접속사 '하고(*and*)'와 문구 '허용(*permit*)'은 둘다 수정(*modification*)과 리버스 엔지니어링(*reverse engineering*)을 지칭한다. 접속사 '*and*'를 삽입함으로써 문장을 더욱 세련되게 구성하였다.

즉, 다음 문장에서 말하고자하는 바는 명백하다.

"단(**provided that**), 이 경우 배포규정에는 수취인들이 필요에 따라 저작물을 수정할 수 있으며 (*and*) 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다"

이는 곧 다음을 의미한다고 볼 수 있다.

"단(**provided that**), 이 경우 배포규정에는 (수취인들이 필요에 따라 저작물을 수정할 수 있으며 (*and*) 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다)"

315) 나중에 원문을 다시 삽입할 것이다!

316) LGPL와 GPL이 처음 정의될 때, 소프트웨어 개발의 지배적인 모델은 C 프로그래밍 언어였다. 이를 염두에 두면 두 라이선스를 이해하기 수월해진다. 따라서 \*join을 C 프로그래밍 언어에 대한 경의 정도로 생각해도 된다.

317) [http://en.wikipedia.org/wiki/Ellipsis\\_%28linguistics%29](http://en.wikipedia.org/wiki/Ellipsis_%28linguistics%29), wp 참조

또한 이는 다음 문장과 동일하다.

"단(provided that), 이 경우 배포규정에는 (수취인들이 필요에 따라 저작물을 수정할 수) 있으 며(and) (수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다)".

우리는 LGPL2-RefEng 문장의 이 부분은 다른 의미로 해석될 여지가 없다고 생각한다. 그러나, 독자가 공식적으로 우리 해석과는 다른 주장을 펼칠 수 있다. 이 경우, 독자는 다른 해석을 적절한 다른 해석을 제공하거나 언어 생략 부호에 대한 다른 제안을 제공해야 한다. 하지만 우리는 누군가 합리적인 대안을 주장하기는 힘들다고 생각한다.

마지막으로, 문장 안에 깊은 의미를 내포하고 있어 추가로 해석이 필요한 부분들이 더 있다.

1. 접속사 '단(provided that)' 이전 부분에서 우리는 이미 '그 저작물(that work)이란 '라이브러리의 일부를 포함하는 저작물을 축약하여 나타낸 문구라고 해석했다. 접속사 이후 문장에서도 '저작 물(the work)이라는 문구가 등장한다. 공식적으로, 이 문구는 연결된 컴포넌트 중 하나로써 '라이브러리를 사용하는 저작물 (the work that uses the library)'를 의미하거나 혹은 연결된 결과물인 '라이브러리를 사용하는 저작물 (the work that uses the library)'를 의미할 수 있다. 우리는 이 문구가 의미하는 바를 지속적으로 해석해 나가기로 결정하였다.
2. 접속사 '단(provided that)' 이후에 첫 문장에서는 "저작물을 수정(하는 것을 허용)"하는 목적에 대해 설명한다. 이 문장은 '라이브러리의 일부를 포함하는 저작물의 수정(을 허용)"한다로 풀어 쓸 수 있다. 두 번째 문장에선 "리버스 엔지니어링(을 허용)"하는 목적에 대해 "수정을 디버깅을 위한" 것이라고 설명한다. 여기서 '수정'이란 '라이브러리의 일부를 포함하는 저작물의 수정'을 뜻한다.
3. 마지막으로, 첫 문장에서 "선택한 규정"에서 언급된 "규정"을 풀어써야 한다.

따라서 종합적으로, LGPL2-RefEng 문장을 의미를 바꾸지 않고 다음과 같이 풀어서 해석할 수 있다.<sup>318)</sup>

( ( 라이브러리와 라이브러리를 사용하는 저작물을 \*연결하여 라이브러리의 일부를 포함하는 저작물을 생성하고)

또한(and)

( 라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다.

) )

단(provided that)

( ( 이 경우 배포규정에는 수취인들이 필요에 따라 라이브러리의 일부를 포함하는 저작물을 수정할 수 있으며)

또한(and)

(배포규정에는 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다

---

318) '\*연결'은 아직 '결합하거나 링크를 통해 연결'을 뜻한다는 것을 상기하자.

) )

는 내용이 포함되어야 한다.

이 시점에서 우리는 독자들에게 이 '구조적으로 복잡한 문장'이 LGPL2-RefEng 문장과 동일한 의미를 갖는다는 것을 확인할 것을 추천한다. 이제 문장의 논리적 해석을 논의하겠다. 논의에 들어가기에 앞서, LGPL2-RefEng 문장을 언어적으로 풀어쓴 해석이 원문과 동일한 의미를 갖는다고 인정을 해야한다. 두 문장의 동일성을 평가하기 위해 독자는 어떤 부분의 해석이 잘못 해석된 여지가 있는지, 그렇다면 어떤 올바른 해석으로 대체할 수 있는지 고려해야한다. 다만 우리는 두 문장이 완벽하게 동일하다고 여긴다.

### 3.4.1.2 논리적 해석

논의를 단순화하기 위해 문장들을 분리하여 논리적 기호로 변환한다.

$\Gamma$  :- (라이브러리를 사용하는 저작물을 \*연결하여 라이브러리의 일부를 포함하는 저작물을 생성(할 수 있다))

$\Delta$  :- (라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다)

$\Phi$  :- (배포규정에는 수취인들의 필요에 따라 라이브러리 일부가 포함된 저작물을 수정을 허용한다)

$\Sigma$  :- (배포규정에는 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 (허용한다))

$\theta$  :-  $\Gamma$  와  $\Delta$

$\Omega$  :-  $\Phi$  와  $\Sigma$

이 정의에 따라, 우리는 LGPL2-RefEng 문장을 다음 공식으로 정리할 수 있다. ( $\Gamma$  와  $\Delta$ )  $\sqcap$  ( $\Phi$  와  $\Sigma$ ), 또는 더 짧게 요약하여 ( $\theta$   $\sqcap$   $\Omega$ )로 나타낼 수 있다.

다음으로, 접속사 ' $\sqcap$ (provided that)'을 명확하게 해석해야 한다.

명백하게,  $\sqcap$ (provided that)은 (어떠한) 조건을 만족했을 시에 라는 뜻이다. 따라서 이 접속사를 다른 유사한 의미를 지닌 만약(. . .)그렇다면(. . .)으로 대체해볼 수 있다. 이는 (논리적) 함축이라고도 불린다.<sup>319)</sup> 이에 따라 문장을 논리적 공식으로 대체할 때 순서를 고려해보아야 한다. 만약 우리가 접속사를 이러한 논리적 함축 문구로 대체하면, 문장의 어느 부분이 전제가 되고, 어느 부분이 결론이 되겠는가? 즉, ( $\theta$   $\sqcap$   $\Omega$ )는 (만약  $\theta$  그렇다면  $\Omega$ )을 뜻하는 것인가 혹은 (만약  $\Omega$  그렇다면  $\theta$ )을 뜻하는 것인가?

$\sqcap$ (provided that)이 전제를 세우기 위한 목적을 가지고 있음을 분명하다. 따라서 ( $\theta$   $\sqcap$   $\Omega$ )는 (만약  $\Omega$  그렇다면  $\theta$ )을 뜻한다고 결론지을 수 있다. 이를 논리적 공식으로 다음과 같이 표기할 수 있다 - (( $\Phi$  ^  $\Sigma$ ) → ( $\Gamma$  ^  $\Delta$ )). 이 해석이 적절하다면, 이는 해당하는 LGPL-v2 섹션의 목적인 LGPL 라이브러리의 일부를 포함하는 저작물의 배포를 규제를 충족해야한다.

319) 실제로는 논리적 함축과 만약-그렇다면 문장은 동일하지 않다. 그러나 해당 문맥에서는 이 차이가 근소하여 무시 할 수 있다.

우리의 주장의 이해를 돋기 위해, 먼저 이 논리적 해석이 LGPL의 목적에 부합하는지 확인해보자. 이를 위해 ( $\Sigma \rightarrow \Delta$ )를 다시 글자 형태로 변환한다.

**만약** ([ . . .] 배포규정에는 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 (허용한다면),) 그렇다면 ([ . . .] 라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다)

이로서 문제점이 더욱 명확해진다. 함축 전체는 전제가 사실이고 결론이 거짓일 경우에만 거짓이다. 모든 다른 경우에는 함축이 사실이다. 특히, 전제가 거짓일 경우, 함축은 사실이 된다. 전제가 거짓이면, 결론이 사실인지 아닌지의 가치는 더 이상 의미가 없어지기 때문이다. 따라서 이 함축을 우리의 행동을 결정짓을 규칙이라고 여기면, 리버스 엔지니어링을 허용하기로 한 경우에만 이를 적용할 수 있다. 이런 경우 규칙에 따라 라이브러리의 일부를 포함하는 저작물을 배포할 수 있다. 그러나 역으로 우리가 리버스 엔지니어링을 허용하지 않기로 결정한 경우, 아무 결론도 내릴 수 없다. 앞서 말했듯이, 거짓 가정은 결론의 사실 여부에 영향을 주지 않기 때문이다. 특히, 이 규칙은 라이브러리 일부를 포함하는 저작물을 배포해도 될지 아닐지 알려주지 않는다. 논리적으로 보면, 원문의 접속사인 ‘*단(provided that)*’은, 만약 선택한 라이선스가 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 허용하지 않는다면<sup>320)</sup>, 그렇다면 라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 **배포할 수도 있고 배포하지 않을 수도 있다**.<sup>321)</sup> 즉, 이 해석은 LGPL-V2의 목적인 LGPL 라이브러리의 일부를 포함하는 저작물의 배포의 규제를 충족하지 않는다. 따라서 LGPL-RefEng 문장을 위에서 설명한 순서대로 해석하는 것은 옳지 않다고 결론을 내릴 수 있다.

그러나 우리는 이 결론을 LGPL2-RefEng의 축약문으로부터 도출했다. 따라서 우리는 같은 결론은 완전히 풀어쓴  $((\Phi \wedge \Sigma) \rightarrow (\Gamma \wedge \Delta))$  공식에서도 도출할 수 있어야 할까? 이에 대한 답변은 ‘그렇다’이다.  $((\Phi \wedge \Sigma) \wedge \Gamma)$ 와 같은 전제는 논리적 결합을 의미하고 있다. 결국 전제의 사실여부는 각각의 마지막 문구의 사실여부에 따른다. 특히  $\Sigma$ 에 대해서는, 우리가 리버스 엔지니어링을 허용하지 않을 경우, 우리가 수정을 허용 혹은 금지하는지 여부와 상관없이, 전제가 거짓이 된다. 결론적으로, 전제는 결론의 사실여부를 결정짓지 않는다. 따라서 우리가 리버스 엔지니어링을 허용해야 하는지, 혹은 허용하지 않아도 되는지 결론지를 방법이 없다. 이에 따라 우리는 다음 결론을 도출할 수 있다:  $(\theta \dashv \Omega)$ 을 가정한다는 것은 곧 **만약  $\Omega$  그렇다면  $\theta$** 이 거짓이라는 뜻이다.

다른 또 하나의 경우를 시험해보자.  $(\theta \dashv \Omega)$ 은 **(만약  $\theta$  그렇다면  $\Omega$ )**을 뜻하는 것일까? 즉,  $((\Gamma \wedge \Delta) \rightarrow (\Phi \wedge \Sigma))$ 이 맞는 것일까? 다시 한 번 이 공식의 축약된 버전인  $(\Delta \rightarrow \Sigma)$ 를 고려하고 이를 풀어써본다.

**만약** ([ . . .] 라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다(면),) 그렇다면 ([ . . .] 배포규정(은) 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 (허용한다).)

---

320) 이 전제은 거짓이다.

321) 규칙은 결론의 사실여부를 결정짓지 않는다.

우리는 이 해석이 LGPL 라이브러리의 일부를 포함하는 저작물의 배포를 완벽하게 규제하고 있음을 알 수 있다. 우리가 LGPL 라이브러리의 일부를 포함하는 저작물을 라이선스 규정을 준수하며 배포하는 경우,<sup>322)</sup> 우리는 리버스 엔지니어링을 허용해야한다.<sup>323)</sup> 이는 함축문에 전건긍정식(*Modus Ponens*)을 적용함으로써 얻을 수 있다.<sup>324)</sup> 그리고 만약 리버스 엔지니어링을 허용하지 않는다면,<sup>325)</sup> 우리는 LGPL 라이브러리의 일부를 포함하는 저작물을 배포할 수 없다.<sup>326)</sup> 이는 함축문에 후건부정식(*Modus Tollens*)을 적용함으로써 얻을 수 있다.<sup>327)</sup>

그러나 우리는 다시 한 번 위 결론이 LGPL2-RefEng 문장을 살짝 요약한 문장으로부터 도출되었음을 고려해야한다. 즉 우리는 완전히 풀어쓴  $((\Gamma \wedge \Delta) \rightarrow (\Phi \wedge \Sigma))$ 으로부터도 같은 결론을 도출할 수 있을지 보여주어야 한다. 만약 우리가 라이브러리와 라이브러리를 사용한 저작물을 연결한 결과물인, 라이브러리 일부를 포함하는 저작물을 배포하고자 할 경우,<sup>328)</sup> 우리가 선택한 규정은 배포할 저작물의 수정 그리고 리버스 엔지니어링을 허용해야 한다.<sup>329)</sup> 만약 우리가 수정 또는 리버스 엔지니어링을 허용하지 않을 경우,<sup>330)</sup> 라이브러리와 라이브러리를 사용한 저작물을 연결한 결과물인, 라이브러리 일부를 포함하는 저작물을 배포하는 행위는 라이선스 조항에 어긋난다.<sup>331)</sup> 즉, 우리는  $((\Gamma \wedge \Delta) \rightarrow (\Phi \wedge \Sigma))$  공식이 LGPL 라이브러리의 일부를 포함하는 저작물의 배포를 완벽하게 규제하고 있다고 말할 수 있다.

위 설명에 따라, 우리는 접속사인 ‘단(provided that)’을 더욱 잘 알려진 ‘함축(implication)<sup>332)</sup>으로 대체할 수 있으며, 이를 논리적 함축을 뜻할 때 자주 사용되는 기호인 ‘ $\rightarrow$ ’으로 나타낼 수 있다.

```
# θ 단(provided that) Ω
≡ θ → Ω
≡ (Φ ∧ Σ) → (Γ ∧ Δ)
≡ ( ( [Φ] 라이브러리와 라이브러리를 사용하는 저작물을 *연결하여 라이브러리의 일부를 포함하는
      저작물을 생성하고 )
      ^
      ( [Σ] 라이브러리의 일부를 포함하는 저작물을 임의로 선택한 규정에 따라 배포할 수 있다
      ) )
      →
      ( ( [Γ] 배포규정에는 수취인들이 필요에 따라 라이브러리의 일부를 포함하는 저작물의 수정할 수
          있으며 )
```

322) 이 전제는 사실이다.

323) 결론 또한 사실이 분명하다!

324) 함축/규칙이 사실일 경우, 사실인 전제는 사실인 결론을 도출한다.

325) 이 결론은 거짓이다.

326) 전제 또한 거짓이 분명하다!

327) 함축/규칙이 사실일 경우, 거짓인 결론은 거짓인 전제를 도출한다.

328) 전제는 사실이다.

329) 전건긍정식(*Modus Ponens*)에 따라 결론 또한 사실이 되어야 한다.

330) 결론은 거짓이다.

331) 후건부정식(*Modus Tollens*)에 따라 전제 또한 거짓이 되어야 한다.

332) 여기서 우리는 만약-그렇다면 문장과 논리적 함축의 차이가 우리의 결론에 영향을 주지 않는다는 것을 확인할 수 있다.

^  
( [Δ] 배포규정에는 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지니어링을 할 수 있다  
))  
는 내용이 포함되어야 한다.

### 3.4.1.3 경험적 해석

이제 몇 가지 경험적 사실을 고려하고 몇 가지 기본적인 이해를 설명하여 이 공식을 다시 간단하게 할 수 있다.

첫 번째 문장의 Φ는 라이브러리와 사용된 라이브러리 자체를 함께 사용하는 저작물이 결합되어 라이브러리의 일부를 포함하는 저작물로 변형됨을 설명한다. 공식적으로, 라이브러리의 일부를 포함하는 이 새로 생성된 저작물이 여전히 라이브러리를 사용하는지 여부를 질문할 수 있다.

불행하게도, 두 가지 컴포넌트를 결합하는 과정은 (a) 라이브러리의 모든 원래 부분을 결코 공개되지 않는 프로그램의 '막 다른 부분'과 같은 것으로 복사 할 수 있다는 것과 (b) 라이브러리의 원래 부분을 다른 라이브러리의 기능적으로 동일한 부분으로 대체할 수 있다는 것이 경험적으로 가능하다. 따라서 라이브러리의 일부를 포함하는 결과물은 실제로는 라이브러리의 일부를 포함하지만 더 이상 사용하지 않을 것이다. 그리고 이 가능성으로 인해, 라이브러리의 일부를 포함하는 모든 다른 저작물도 라이브러리를 사용한다고 말할 수는 없다.<sup>333)</sup>

그러나 다행스럽게도 라이브러리를 라이브러리와 함께 사용하여 일부를 포함하는 저작물을 결합하고 링크하는 일반적인 프로세스는 본질적으로 결합된 라이브러리의 활용도를 보존하게 된다. 이는 응용 프로그램에서 실제로 사용되는 기능 그리고/또는 데이터 (구조)에 대한 소프트웨어 라이브러리의 일반적인 목적이다. 또한 소프트웨어 개발자는 이 서비스를 선호하기 때문에 특정 라이브러리를 참조한다. 작업 품질 수준을 유지하면서 자신의 작업을 단순화하기 위해 쉽게 준비된 라이브러리 (또는 클래스 등)을 사용하는 것이다. 프로그래밍의 이러한 실행과 관련하여 소스와 객체 파일을 결합, 컴파일 및 링크하는 **일반적인 개발 프로세스에 의해 구축된** 라이브러리의 부분을 포함하는 저작물은 실제로 의도된 라이브러리를 사용한다고 말할 수 있다.

이제 우리는 첫 번째 문장 Φ와 두 번째 문장 Σ 사이의 경험적 상관을 고려할 수 있다.

다른 말로하면, 우리는 이미 각기 결합되거나 연결되어 합쳐진 Φ를 완성해 왔음이 분명하다. (라이브러리의 일부를 포함하는 저작물을 작성하기 위해 라이브러리와 함께 라이브러리를 사용하는 저작물, 사용자가 선택한 조건을 준수하며 라이브러리의 일부를 포함하는 저작물을 배포하려는 경우) 아니면 간단히 말해서 Σ는 경험적으로 단호히 Φ를 의미한다.<sup>334)</sup>

---

333) ... 비록 이것이 우리가 합류하는 과정을 편성하는 아주 어리석은 방법이라고 생각한다고 할지라도!

334) 그러나 그 반대는 아니다.

하지만 과연 이 결론이 옳은 것일까? 이 결론이 맞을지 반대 의견을 가정하여 문장을 확인해보자. 반대가 사실이라면 어떤 의미에서든 저작물과 라이브러리를 연결하거나 결합하지 않고 얻은 라이브러리의 일부를 포함하는 저작물이 존재해야 했다. 그러나 위의 추론을 통해 우리는 라이브러리를 사용하는 저작물을 결합하고 링크하는 표준 컴퓨팅 과정에서 생성된 라이브러리의 일부를 포함하는 저작물이 라이브러리를 실제로 사용하는 것을 이미 알고 있다. 따라서 표준 결합 및 링크 과정에 의해 만들어지고 어떤 의미에서든 라이브러리와 결합된 것이 아닌 상태와 유사한 상태의 라이브러리의 일부를 포함하는 저작물에 대해 이야기하는 것은 자기모순이 될 수 있다. 그리고 모순에 의한 증거로부터 논리적 반대의 진실을 추론해 보면 표준 결합되거나 링크되는 것의 의미와 관련하여 아래와 같이 말할 수 있다.

- 사실상 라이브러리와 라이브러리를 사용하는 저작물의 기초 위에서 표준으로 작성되어 사실상 라이브러리의 상당 부분을 포함하는 저작물은 실제로 라이브러리를 사용하는 것으로 사실상 라이브러리이다.
- $\Sigma^{335}$ 는 경험에 기인하여  $\Phi^{336}$ 를 함축하고 있는데 (소프트웨어 개발의 표준화 된 세계에서) 이는  $\Sigma$ 가 실행되려고 할 때  $\Phi$ 가 실행되어 있어야하기 때문이다.

따라서 이제 LGPL2-RefEng-Sentence를 실제 핵심 인 LGPL2-RefEng-Rule로 줄일 수 있다.

( [Σ] (a) 라이브러리의 일부를 포함하는 저작물을<sup>337</sup> 임의로 선택한 규정에 따라  
배포할 수 있다)

→

( ( [Γ] 배포규정에는 수취인들이 필요에 따라 라이브러리의 일부를 포함하는 저작물의 수정할 수  
있으며 )

^

( [Δ] 배포규정에는 라이브러리의 일부를 포함하는 저작물의 수정에 따른 디버깅을 위한 리버스 엔지  
니어링을 할 수 있다

) )

는 내용이 포함되어야 한다.

이것은 실제로 LGPL2-RefEng-Sentence의 본질이다. 우리가 리버스 엔지니어링을 허용하고 라이브러리의 일부를 포함하는 저작물을 수정해야만 한다는 점과(긍정논법) 수정이나 리버스 엔지니어링을 허용하지 않는다면(부정논법) 라이브러리의 일부를 포함하는 저작물을 배포 할 수 없다는 점을 논리적으로 설명하고 있다.

따라서 이 규칙을 올바르게 적용하기 위해서는 라이브러리의 일부가 실제로 포함되어 있는지 여부를 알  
아야 한다.

---

335) 라이브러리를 사용하고 라이브러리의 일부를 포함하는 저작물을 배포하는 행위

336) 라이브러리를 사용하는 저작물은 라이브러리의 일부를 포함하는 저작물을 만들기 위해 라이브러리와 결합되었다.

337) 라이브러리를 사용하는 저작물을 실제로 라이브러리를 사용하는 방식으로 라이브러리와 함께 결합 및 링크하여  
배포하기 위해 사전 준비되었다.

#### 3.4.1.4 최종 결론

불행하게도, 질문에 대한 대답을 자세하게 답변하는 것을 고려해야하는 하나 이상의 소프트웨어 개발 시나리오가 있다. 세 가지 유형의 컴퓨터 소프트웨어 개발 형태가 있다.

1. 스크립트 언어를 사용하여 소프트웨어를 만들 수 있다. 스크립트 언어 명령을 포함하는 소스 파일은 사전에 다른 "더 많은" 기계 사양 언어로 변환되지 않고 인터프리터에 의해 배포되고 실행된다.
2. 기계 독립적인 바이트 코드로 컴파일 되도록 설계된 언어를 사용하여 소프트웨어를 개발할 수 있다. 나중에 이 독립 바이트 코드는 시스템 특정적인 가상 머신에 의해 실행된다.
3. 전통적인 소프트웨어 파일을 작성 할 수 있다. 때로는 이러한 파일은 실제 프로세스가 시작되기 전에 전처리기에 의해 재처리된다. 전통적인 소스 자체나 전처리기의 출력 파일은 컴파일되어 기계의 특정 바이너리 파일로 링크된다.

당신은 'php'를 첫 번째 환경에 대한 예제로 사용할 수 있는데 'Java'는 두 번째 예제이고 'C / C ++'는 세 번째 예제이다.

다행히도 이러한 환경의 본질은 라이브러리를 사용하는 작업에 라이브러리의 일부분이 포함되어있는 조건에 대한 질문에 대한 답을 명료화해준다.

**3.4.1.4.1 라이브러리의 일부를 수동으로 복제한 저작물 배포 시 Copyleft 가 적용된다:** 라이브러리의 소스에서 코드를 라이브러리를 사용하는 포괄적인 작업으로 수동으로 복사하는 것은 스크립트 프로그래밍이나 바이트 코드 프로그래밍 또는 특정 시스템 기반 프로그래밍 계에서 두 컴포넌트를 결합하는 표준 방법이 아니다.

일반적으로 라이브러리를 사용하는 저작물은 include 문, 입력문, import 문, 패키지 문 또는 기타 항목에 의해 의도된 라이브러리에 연결된다. 이러한 결합 구문들은 저작물의 코드에 삽입되며 사용된 기능, 메소드, 클래스 또는 데이터를 전달하는 파일을 나타낸다. 그러한 결합 구문을 삽입하는 것이 라이브러리의 일부 코드에 의해 라이브러리를 사용하는 작업을 직접적으로 증가시키는 것은 일반적인 개발 도구의 통합된 기능이다. 그 개발 프로세스들은 실제 개발 과정이 종료된 후에 시작되는 표준 컴파일의 일부로 자동 기능 보강을 위해 설계되었다.

그럼에도 불구하고 개발자는 라이브러리 사용에 대한 이러한 표준 방법을 우회 할 수 있다. 기술적으로 라이브러리의 코드를 직접 자신의 저작물로 복사 할 수 있다. 결과적으로, 이러한 수동으로 복사된 코드는 저작물의 '자체'코드와 함께 컴파일 또는 실행된다. 따라서 이 경우 특히 수행 환경이 실행되기 전의 결합 단계에서 라이브러리를 사용하는 저작물에는 라이브러리의 일부가 이미 포함되어 있다는 점은 분명하다.

따라서 라이브러리 소스 코드의 리터럴 사본을 포함하는 저작물을 배포하려는 경우 이러한 추가 파일을 기반으로 컴파일 되었지만 여전히 링크되지 않은 경우에도 리버스 엔지니어링을 허용해야한다.<sup>338)</sup>

그러나 라이브러리를 사용하여 라이브러리에서 직접 코드를 복사하는 경우 LGPL-v2가 라이브러리를 사용하여 이러한 종류의 코드를 직접 규제한다는 점도 고려해야한다. LGPL-v2는 “귀하의 사본 또는 라이브러리의 일부 또는 사본을 수정할 수 있다 (중략) 저작물 전체에 라이선스가 부여된다 (중략) 이 라이선스의 규정에 따라”라고 기술하고 있다.<sup>339)</sup> 따라서 LGPL은 라이브러리에서 코드를 문자 그대로 라이브러리를 사용하여 저작물에 복제하는 경우 카피레프트 효과를 유발하는 명제에 대한 강력한 주장을 포함하고 있다. 라이브러리를 이용하는 저작물의 코드도 액세스 가능해야한다.

따라서 전반적으로, 표준 방식으로 소프트웨어를 결합하고 배포하는 것의 우회적인 방법인 라이브러리를 이용하여 LGPL-v2 라이브러리로부터 저작물에 수동으로 코드를 복제하는 것은 이 권한이 코드에 직접 액세스해야하기 때문에 받는 사람에게는 그다지 중요하지 않더라도 실제 리버스 엔지니어링을 추가적으로 허용해야한다고 말할 수 있다.

**3.4.1.4.2 스크립트 배포 시 리버스 엔지니어링을 필요로 하지 않는다:** 스크립트 언어로 작성된 컴퓨터 프로그램은 개발된 대로 배포되며 다른 종류의 코드로 변환되지 않는다.<sup>340)</sup> 인터프리터는 스크립트 파일을 그대로 가져와서 직접 실행한다. 따라서 이런 종류의 소프트웨어를 이해하기 위한 리버스 엔지니어링의 특별한 기술은 없으며 스크립트 언어를 알고 있으면 직접 읽을 수 있다.

다시 말해 스크립트 라이브러리를 이용하는 스크립트는 리버스 엔지니어링을 허용하도록 되어있을 수 있으나 직접 그 코드를 읽을 수 있게 되어있기 때문에 받는 사람에게 이 권한은 중요하지 않을 수 있다.

---

338) 이것은 긍정논법의 LGPL2-RefEng-Rule을 직접적으로 따른다. 그럼에도 불구하고 라이브러리에서 코드를 수동으로 라이브러리의 사본을 저작물에 복사 한 결과조차도 LGPL-v2-§5에서 소개된 허용 오차 한계에 포함된다. 형식적으로 이 주장은 적절하게 보이며 실제로 이러한 허용 한계를 고려해야 한다. 그러나 라이브러리의 코드를 직접 저작물에 복제하는 것과 관련하여 자세히 살펴보면 그 영향은 줄어들게 된다. 다음의 세 가지 경우를 구분해야 한다.

1. 개발자는 라이브러리 헤더 파일의 일부 또는 대부분의 요소를 전처리기가 자체 코드로 복제 / 전송하는 저작물의 코드 내에 수동으로 복제 / 전송할 수 있지만 그렇게 하지는 않을 것이다. 일부 간단한 include 문은 동일한 효과를 발생시킨다. 개발자는 리소스, 특히 자신의 근무 시간을 줄이기를 원한다. 그렇다면 표준 개발 프로세스에 위임 할 수 있는 작업을 수동으로 수행해야하는 이유는 무엇일까. 그럼에도 불구하고 라이브러리의 일부를 자신의 저작물에 복사하는 개발자는 전처리기의 서비스를 수동으로 반복하지 않고 단지 이러한 요소 뿐만이 아닌 그 이상을 전송하려고 한다고 가정하는 것이 타당할 것이다. 그러므로 LGPL2-RefEng-Rule의 적용을 받는 것으로 가정하는 것이 합리적이다.
2. 개발자는 라이브러리 헤더 파일의 요소만 라이브러리 소스에서 라이브러리를 사용하는 저작물의 코드로 수동 복제 / 전송할 수 있으며 그럼에도 라이브러리에 링크되도록 할 수 있다. 하지만 이번에도 역시 개발자는 그렇게 하지는 않을 것이다. 왜냐하면 일부 단순한 include 문과 링크 명령어는 동일한 효과를 발생시키기 때문이다. 따라서 복제를 수행하는 개발자는 실제로 이보다 더 많은 것을 한다는 것을 전제로 시작하는 것이 합리적이라고 하겠다. 그래서 그들의 저작물은 LGPL2-RefEng-Rule의 적용을 받는 것으로 가정하는 것이 합리적이다.
3. 개발자는 라이브러리 헤더 파일의 요소만 라이브러리에 링크하지 않고 라이브러리 소스에서 라이브러리를 사용하는 저작물의 코드로 수동 복제 / 전송할 수 있다. 이는 개발자가 자신의 저작물을 라이브러리에 링크 할 수 없도록 하기 때문에 합당한 작업 과정이다. 그러나 정의된 것에 의해 증대된 저작물은 LGPL-v2-§5가 허용하는 라이브러리의 더 많은 요소를 포함한다. 그래서 이번에도 역시 LGPL2-RefEng-Rule의 적용을 받는 것으로 가정하는 것이 합리적이다.

339) cf. Open Source Initiative: The GPL 2.1 License (OSI), 1999, wp., §2, especially §2c.

340) 자바스크립트는 종종 압축된 형태의 코드로 제공된다. 대략적으로 말해서, 이것은 처음에 모든 공백 표시가 공백으로 대체된 다음 모든 공백 행이 하나의 공백으로 줄어들었다는 것을 의미한다. 그럼에도 불구하고 코드 자체는 직접 읽고 이해할 수 있는 형태이며 이는 물론 매우 정교한 전문가들에게만 가능한 이야기이다.

**3.4.1.4.3 정적 결합된 바이트 코드 배포 시 리버스 엔지니어링 허가가 필요하다:** 독립적인 휴대용 바이트코드로 컴파일된 언어의 프로토타입인 Java에서 각 클래스는 별도로 컴파일된다. 이 클래스 파일은 클래스 경로 어딘가에 저장되어야 한다. 그 측면에서 클래스는 또한 '전통적인' 라이브러리들처럼 사용할 수 있는 패키지의 형태로 수집되거나 배포될 수 있다. 이러한 패키지는 클래스 경로 어딘가에 저장되어야 한다. 하나의 클래스는 그것을 사용하려는 작업물에 알려지게 되고 패키지 문을 코드에 통합하여 전체 Java 라이브러리는 사용 가능하게 된다.

임포트 또는 패키지 명령문 다음에 오는 코드는 클래스에 의해 제공된 정의를 이용할 수 있다. Public 또는 Protected 멤버 변수 또는 메서드의 (한정된) 이름으로 클래스의 요소를 나타낸다. 따라서 엄격한 관점에서 볼 때 라이브러리를 사용하는 Java 작업의 코드에는 실제로 해당 라이브러리의 일부가 포함되어 있다고 볼 수 있는데 이는 명칭만을 식별하거나 식별명을 포함하는 데이터구조일 경우에도 마찬가지이다. 바이트 코드를 생성하는 Java 컴파일 프로세스는 이런 이름을 유지한다. 이것이 참조된 메소드의 참조 코드 등으로 참조명을 대체하지는 않는다. 결국, 자바 가상 머신 자체가 라이브러리를 사용하여 작업을 실행하려고 할 때, 모든 결합된 클래스에 필요한 모든 명령들을 수집한다.

따라서 분산된 자바 바이트 코드가 이미 서용된 라이브러리의 일부를 포함하고 있는 여부에 대한 질문에 대답하는 것은 해석하기 나름이며 실제로 라이브러리의 식별자가 일부인지 여부에 달려있다. 이 사례는 C/C++-케이스에서 함께 다룰 것이다.

그러나 Java의 또 다른 측면이 있는데 이 역시 고려해야 한다. 이미 언급했듯이 Java에서는 식별자를 나타내는 저작물 및 표시된 라이브러리를 포함하여 새 패키지를 작성하면 라이브러리를 사용하는 저작물과 사용된 라이브러리 모두를 포함하는 저작물을 결합할 수 있다. 따라서 이 패키지가 거의 정적으로 링크되어 있다고 말할 수 있는데 이러한 통합 패키지를 배포하면 두 구성요소를 함께 배포할 수 있다. 따라서 전체 패키지를 배포하는 경우, 즉, 라이브러리 및 라이브러리의 모든 부분을 사용하는 저작물을 포함하는 거의 정적으로 링크된 저작물인 경우 리버스 엔지니어링을 허용해야 한다.<sup>341)</sup>

따라서 Java 프로그래밍과 관련하여 (a) 라이브러리 및 라이브러리 자체를 (정적으로 링크 된) 통합 패키지로 배포하는 경우 리버스 엔지니어링을 허용해야 하며<sup>342)</sup> (b) 다른 모든 경우에는 리버스 엔지니어링을 허용해야 하는 의무는 라이브러리가 선언한 식별자가 실제로 라이브러리의 일부인지 여부에 달려 있다.

다행스럽게도 사례 (b)의 문제는 곧 합리적으로 결정할 수 있을 것이다.

**3.4.1.4.4 정적 결합된 바이너리 배포 시 리버스 엔지니어링 허가가 필요하다:** Java와 마찬가지로 장비의 존적인 코드로 컴파일된 언어들 중 프로토타입에 해당하는 C/C++에서 C/C++ 라이브러리는 또한 명시적으로 이를 사용하려는 저작물 즉, 몇몇 include 구문으로 이를 알려준다. 이러한 include 문은 라이브러리에 의해 생성되고 라이브러리와 함께 배포되는 헤더 파일을 나타낸다. 여기에는 라이브러리가 게시하고자 하는 요소의 선언부가 포함되어 있다. 간단히 말하자면, 라이브러리는 코드 형식의 정의를 포함

341) 이것은 긍정논법의 LGPL2-RefEng-Rule을 직접적으로 따른다.

342) 이것은 긍정논법의 LGPL2-RefEng-Rule을 따른다.

하고, 헤드 파일을 해당 선언을 포함한다고 할 수 있다.

그러한 include문 다음의 C/C++ 코드는 헤더 파일에 의해 알려진 선언을 사용하여 라이브러리에 의해 제공된 정의를 참조할 수 있다. 그래서 다시 엄격한 관점으로 봤을 때 라이브러리를 사용하는 C/C++ 코드로 된 저작물은 그 일부가 헤더 파일들에 의해 공표된 이름 또는 데이터 구조들만 식별하는 경우라 하더라도 실제로 라이브러리의 일부를 포함하고 있다고 할 수 있다.

이러한 개념적 관계 외에도 C/C++ 개발 프로세스는 최종적으로 특정 장비 의존적인 코드가 포함된 객체 파일로 라이브러리를 사용하여 저작물을 컴파일 한다. Java 컴파일과 마찬가지로 이 프로세스는 참조된 이름을 라이브러리의 참조된 코드로 대체하지 않고 항상 그 이름을 표시한다. 결과 파일은 장비 의존적인 코드로 컴파일 되었지만 항상 '오브젝트 코드 파일'로 불리는 식별자를 나타내는 파일을 포함한다.

C/C++ 컴파일 프로세스는 (대부분) make 명령에 의해 실행되는 make 파일에 의해 관리된다.<sup>343)</sup> 이 개발 도구는 각 소스 파일에 대해 컴파일러를 호출하고 컴파일 된 대상 객체 파일을 포함하는 디렉토리를 알려주고 마지막에 링커를 호출한다. 링커는 컴파일 된 객체 파일을 재귀적으로 스캔하고 라이브러리에 있는 저작물의 일부로 실행되는 라이브러리 명령 집합에 실제 실행 점프 명령으로 각 임베디드 식별자를 대치한다. 따라서 결국 링커는 모든 '결합된' 객체 파일과 라이브러리에 필요한 모든 명령을 수집하고 실제로 실행 가능한 작업을 생성한다.

그러나 위 내용에도 불구하고 링커는 개발 과정 자체의 통합 단계로 불릴 수 있다. 또는 특히 다른 시스템에서 링커를 분리하여 호출 할 수 있는데 첫 번째 경우, 개발 프로세스는 사용된 모든 라이브러리의 모든 필요한 부분을 이미 포함하고 있는 정적 링크된 실행 파일을 생성한다. 두 번째 경우에, 개발 프로세스는 여전히 배포되지 않은 (unlinked) 오브젝트 코드 파일들을 배포 가능한 패키지로 수집함으로써 동적으로 링크 가능한 프로그램을 생성한다. 따라서 정적으로 링크된 실행 파일을 배포하는 경우 라이브러리의 '일부분'을 포함하게 되는 것이다. 동적으로 링크 가능한 프로그램을 배포하는 경우 라이브러리의 임베디드 식별명이 실제로 라이브러리의 일부로 해석되어야하는지 여부를 결정해야한다.

불행히도 우리는 C/C++ 개발 프로세스의 성격에 따라 약간의 복잡성을 고려해야한다. Java 개발 환경과는 달리, C/C++ 개발 프로세스는 본질적으로 전처리 엔진을 사용한다. 이 엔진은 라이브러리에 의해 전달된 헤더 파일을 취하여 구문론적으로 올바른 라이브러리의 사용을 검증하고 라이브러리의 명령 또는 라인에 의해 라이브러리를 사용하는 작업의 토큰을 실제로 대체 할 수 있다. 이 기술은 인라인 함수 또는 매크로라고 한다. 실제 함수 호출로 컴파일 하는 동안 명령 스택을 확장하는 것이 함수의 임베디드 된 명령을 한 번 이상 전체 코드에 작성하는 것보다 비용이 많이 듈다. 따라서 C/C++ 개발 프로세스에서 컴파일 된 객체 파일에는 실제로 라이브러리의 일부를 나타내는 참조 이름만이 포함될 수 있고 식별자를 표시하는 것 외에도 라이브러리의 실제 기능적으로 관련된 부분을 이미 포함 할 수 있다.

---

343) 때로는 그러한 make 파일을 생성하는 완전한 메타 환경이 추가로 존재한다. 예를 들어 GNU 빌드 시스템은 복잡한 설정 스크립트를 제공하고 파일 템플릿을 만든다. (cf. [http://en.wikipedia.org/wiki/GNU\\_build\\_system](http://en.wikipedia.org/wiki/GNU_build_system), wp.).

따라서 Java 컴파일과 유사하게 C/C++ 프로그래밍과 관련하여 (a) 라이브러리를 함께 정적 링크된 프로그램으로 배포하는 경우<sup>344)</sup> 리버스 엔지니어링을 허용해야한다는 결론을 내릴 수 있다. (b) 다른 모든 경우 엔지니어링을 허용할 의무는 라이브러리에 의해 선언되었고 오브젝트 파일에 라이브러리의 일부로써 자동으로 그리고 표준에 일치하도록 임베디드되어 사용된 식별자 또는 해독된 인라인 함수 및 매크로가 자동 및 표준 준수가 객체에 포함되어 있는지 여부에 따라 달라질 것이다.

이제는 분명히 이 중요한 질문에 답할 때이다.

**3.4.1.4.5 동적 결합이 가능한 바이트 코드 및 링킹 가능한 오브젝트 코드는 리버스 엔지니어링 권한을 필요로 하지 않는다:** 물론 라이브러리를 사용하는 저작물에 포함된 개발 표준에 따라 식별 및 해독된 인라인 함수 또는 매크로가 실제로 라이브러리의 일부인지 여부와 같은 질문에 대답할 수 있는 인스턴스는 하나뿐이고 이 인스턴스는 LGPL-v2 자체입니다. 그리고 다행스럽게도 이 라이선스는 리버스 엔지니어링을 다루는 §6이 아니라 §5에서 이 질문에 답하는 매우 분명한 방법으로 우리를 지원하고 있다.

LGPL은 단순히 "'라이브러리를 사용하는 저작물'을 라이브러리와 링크하는 것은 라이브러리의 파생물(라이브러리의 일부가 포함되기 때문에)인 실행 파일이 생성된다 (중략)" 더욱이 "실행 파일은 따라서 본 라이선스의 적용을 받는다."<sup>345)</sup>라고 명시하고 있다. 또한 컴파일 되었지만 여전히 링크되지 않은 목적 파일이기 때문에 실행 파일이 아닌 것들에 대해 다루고 있다. "숫자로 된 매개 변수, 데이터 구조 레이아웃 및 접근자 그리고 작은 매크로와 작은 인라인 함수(10줄 이하)를 사용하는 (중략)" 그러한 실행 불가능한 "목적 파일"은 주어진 제한을 초과하지 않는 한 "목적 파일의 사용은 이것이 법률적으로 파생 저작물인지 여부에 관계없이 금지되지 않은 (중략)" 이유로 "라이브러리를 사용하는 저작물"과 같이 라이브러리의 라이선스에 의해 실질적으로 다루어지지 않아야 한다.

분명히 우리의 질문에 대한 LGPL의 대답은 다음과 같다. (a) 예, 사용된 라이브러리에 의해 이름과 스니펫이 포함된 목적 파일에는 라이브러리의 일부가 포함 될 수 있다. (b) 표준 컴파일 과정이 어떠한 요구 사항도 유발시키지 않는 것에 의해 목적 파일에 크기의 한계까지 이러한 종류의 작은 부분이 임베디드 되기 때문에 세부사항을 명확히 할 필요는 없다. 그들은 특히 리버스 엔지니어링을 허용해야 할 의무를 느끼지 않는다. 바꿔 말하면 표준 개발 프로세스에 의해 내장되고 특정 크기 이상의 코드를 포함하지 않는 라이브러리의 이러한 작은 부분은 실제로 리버스 엔지니어링을 허용해야하는 의무를 제기하는 정상적인 실제 부분과 비교하여 다른 유형의 부분으로 간주 될 수 있다. LGPL의 관점에서 볼 때, 그들은 어떤 점에서든 객체를 제한하지 않기 때문에 라이브러리의 모조 부분으로 볼 수 있다.

따라서 LGPL-RevEng-Rule에서 라이브러리가 제공하는 "수치 매개 변수, 데이터 구조 레이아웃 및 접근자, 작은 매크로 및 작은 인라인 함수 (10줄 이하)"를 포함하는 동적으로 링크나 결합 가능한 바이트 코드나 오브젝트 코드 파일은 리버스 엔지니어링을 허용할 필요가 없다고<sup>347)</sup> 간접적으로 결론을 내릴 수 있다.

---

344) 이것은 긍정논법의 LGPL2-RevEng-Rule을 따른다.

345) cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. §5.

346) cf. id., ibid.

347) 리버스 엔지니어링을 허용하지 않기로 하는 결정에 이어 부정논법에 의해 LGPL2-RevEng-Rule에 라이브러리를 사용하는 저작물의 배포에는 라이브러리의 실제 부분이 포함되어서는 안된다는 것이 적용되었다. LGPL-v2-§5 부

불행히도, 우리의 단순한 결과를 방해하는 것으로 보이는 실질적인 반대가 있을 수 있다. 이 규칙을 올바르게 적용하려면 라이브러리를 사용하지만 아직 결합되지 않은 컴파일 된 저작물이 "작은 매크로 또는 작은 인라인 함수 (10줄 이하의 길이)"로만 확장되었는지를 분명히 확인해야한다. 따라서 리버스 엔지니어링을 허용하지 않고 라이브러리를 사용하여 저작물을 배포하려는 경우 사용된 모든 라이브러리의 헤더들을 자세히 조사해야한다. 이것은 실질적으로 관리 할 수 없는 대량의 작업 일 수 있다.

다행히도, 이 과제에 대해 "업스트림 신뢰" 원칙에 기반한 경험 법칙이라는 해결책이 존재한다.<sup>348)</sup>

물론 라이브러리 개발자들은 헤더 파일이나 클래스의 공용 멤버 및 함수를 이러한 요소들이 사용되기를 원하는 형태로 발표한다. 또한 그들은 자신의 라이브러리를 LGPL 라이브러리로 사용하기를 원하는데, 그렇지 않으면 다른 라이선스를 선택했을 것이다. 그래서 그들은 라이브러리의 개선작업이 용이하도록 만들어지기를 원하지만, 라이브러리를 사용하는 저작물이 반드시 소스 코드의 형태로 배포되는 것은 아니다.<sup>349)</sup> 따라서 그러한 원 저작자가 배포하는 형태 그대로 사용하고 공식 저장소에서 내려 받으며 원래 그대로의 헤더와 클래스 파일에 의해 정의되고 배포된 의도된 인터페이스들을 수정하지 않는 한 저작권 소유자가 사용되기를 원하는 그대로 라이브러리를 사용하고 있다고 정당하게 추측 할 수 있다. 따라서 우리가 업스트림을 신뢰한다면 라이브러리의 헤더와 클래스 파일은 LGPL-v2의 제한을 따른다고 가정 할 수 있다.

**3.4.1.4.6 리버스 엔지니어링 권한이 있을 때와 없을 때의 LGPL-2.0 컴플라이언스:** 이제 우리는 목표하는 바에 도달했다. 우리의 마지막 정의는 C/C++ 오브젝트 코드의 경우와 마찬가지로 Java 바이트 코드를 배포하는 경우 모두에 대해 적용될 수 있다. 현재 LGPL-v2은 라이브러리를 사용하는 저작물의 일부가 리버스 엔지니어링의 허용을 유발하지는 않는다는 사실을 알고 있다. 그리고 우리는 표준 개발 환경에서 '업스트림 승인된' C/C++ 및 Java 라이브러리를 사용한다면 현재 LGPL-v2-§5가 제공한 한계가 리버스 엔지니어링을 허용해야하는 의무를 유발하지는 않는다는 것을 알고 있다. 따라서 실제로 LGPL-RevEng-Sentence "(중략) 귀하는 (중략) 라이브러리의 일부를 포함하는 저작물을 만들고 자신의 저작물을 수정하는 것을 허용하고 그러한 수정을 디버깅하기 위해 리버스 엔지니어링을 사용하는 것과 같이 당신의 선택에 따른 조건 하에 배포하기 위해 라이브러리를 사용하는 저작물을 라이브러리와 결합하거나 링크 할 수 있다."<sup>350)</sup>는 다음 그 이상을 의미하지 않는다고 결론내릴 수 있다.

- LGPL-v2 라이선스 라이브러리와 관련하여 [A] 원저작자가 설계한 것처럼 인터페이스가 포함된 표준 버전의 라이브러리를 기반으로 하여 라이브러리를 사용해 자체 저작물을 개발하는 경우, [B] 본 라이브러리를 사용하여 귀하의 저작물을 동적으로 링크 또는 결합 할 수 있는 파일들의 이산 집합으로 컴파일 하는 경우, [C] 업스트림 승인된 인터페이스를<sup>351)</sup> 보존하는 표준 컴파일 방법만 사용하는 경우, [D] 링크가

---

터는 표준 절차의 한계에 따라 라이브러리를 사용하는 저작물에는 정상적인 실제 부분이 포함되지 않는다. 따라서 이 경우는 LGPL2-RevEng-Rule의 적용을 받지 않기 때문에 리버스 엔지니어링을 허용하지 않고 라이브러리를 사용하여 저작물을 배포 할 수 있다.

348) 2013년 ELLW에서는 이 원칙에 대해 처음으로 들을 수 있었다. Armijn Hemel이 그것을 발명했는지 여부는 알 수 없지만 우리는 그것이 "업스트림을 신뢰하라"는 원칙의 정신과 목적을 설득력있게 설명했다고 정중하게 생각할 수 있다.

349) 약한 카피 레프트의 의미.

350) cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp., §6, emphasis KR..

351) 따라서 LGPL-v2 한도를 초과하지 않아야한다!

풀린 오브젝트 코드 또는 바이트 코드 파일들을 실행 파일로 링크되기 전에 배포하는 경우에는 리버스 엔지니어링을 허용 할 필요는 없다.

- 그러한 라이브러리를 사용하여 저작물을 배포하는 다른 모든 경우에는 이 라이브러리를 사용하여 저작물의 리버스 엔지니어링을 허용해야한다. – 특히,
  - 라이브러리와 라이브러리를 사용하여 저작물을 정적으로 링크된 프로그램 또는 라이브러리와 라이브러리 자체를 사용하는 저작물을 포함하는 통합 패키지로 함께 배포하는 경우.<sup>352)</sup>
  - 수동으로 복사 한 라이브러리의 일부가 포함 된 저작물을 배포하는 경우.

### 3.4.1.5 최종 확보

지금까지 우리는 많은 작업을 해왔다: 처음에는 LGPL2-RevEng-Sentence의 세련된 요약본을 언어적으로 명백한 버전으로 전개하고 풀이했다. 둘째로, 우리는 문장의 논리적 구조를 설명했다. 셋째로는 문장의 진정한 의미를 경험적으로 새겼다. 그리고 마지막으로 규칙의 트리거 부분을 검증 가능한 사실로 맵핑했다.

사실 한 문장만 정확하게 이해하기위한 것에도 많은 노력이 필요하다.<sup>353)</sup> 따라서 파생된 결과가 LGPL-v2의 정신 그리고 목표와 완벽하게 일치하는지 확인하는 것이 좋다.

이러한 목적으로, LGPL 라이선스가 적용된 Java 라이브러리의 호환 사용을 다루는 David Turner가 작성하고 FSF가 발행한<sup>354)</sup> (약간의 반 공식쯤에 해당하는) 문서에 대해 논의해보자. Turner는 "FSF의 입장"을 말하며, "(중략) 그 동안 일정하게 유지되었다."고 했다.

"(중략) LGPL은 Java를 포함한 모든 알려진 프로그래밍 언어로 의도된 대로 작동하며, LGPL 라이브러리에 링크되는 응용 프로그램은 LGPL 하에 배포될 필요가 없다. 응용 프로그램은 LGPL의 섹션 6의 요구 사항만을 준수하면 된다: 라이브러리의 새 버전을 응용 프로그램과 연결할 수 있다. 이를 리버스 엔지니어링으로 디버그 할 수 있게 허용한다."<sup>355)</sup>

---

352) 이는 스크립트 언어 기반 응용 프로그램을 이해하기 위해 리버스 엔지니어링의 허용이 필요없다는 사실에도 불구하고 스크립트 언어 기반 프로그램이나 패키지를 배포하는 경우에도 마찬가지이다.

353) 여기서 일부 독자는 원저자가 왜 그런 정교한 문장으로 명확한 아이디어를 캡슐화 했는지를 묻는다. 여기에 2 가지 대답이 있다. 첫째, 이 질문은 실제적으로 부적합하다. LGPL-v2의 저작자가 한 일이며 또한 많은 개발자들이 LGPL-v2의 조건에 따라 이미 라이선스를 취득했다. 따라서 LGPL-v2의 조건에 따라 게시되는 마지막 소프트웨어가 더 나은 버전으로 다시 라이선스 될 때까지 결과를 유지해야한다. 아마 이것은 우리의 평생 동안 일어나지 않을 것이다. 둘째, LGPL-v2 저작자의 예지에 감사한다. 그들은 20년 이상 성공적으로 역할을 수행해온 라이선스를 작성했다. 그들은 '무결점(uninvented)'기술을 다루어야하는 제형을 선택했다. 그래서, 이는 그리 놀랄만 한 것이 아니며, 오늘날 우리는 원저작자가 이해 하고자하는 모든 세부 사항을 이해하기 위해 많은 노력을 기울여야한다.

354) cf. Turner, David: The LGPL and Java; 2004 <URL: <http://www.gnu.org/licenses/lgpl-java.en.html>> - 참조 다운: 2015-02-09, wp..

355) cf. id., l.c., wp.

그리고 그는 Java 라이브러리가 "(중략) 별도의 JAR (Java Archive) 파일로 배포되고" 이 응용프로그램은 "(중략) Java 라이브러리의 클래스에 액세스하기 위해 Java의 import 기능을 사용 한다"고 설명한다. 또한 컴파일 과정은 컴파일 된 응용 프로그램에 "링크"를 "생성"하고 통합하여 응용 프로그램을 라이브러리의 "파생 저작물"로 만든다. 마지막으로 그는 LGPL이 그러한 파생 저작물을 배포하는 것을 허용할 뿐만 아니라 사용자가 정말 원한다면 "(중략) LGPL 라이브러리를 import 하는 Java 응용프로그램을 배포하려는 경우" "LGPL을 준수하기 쉽다"고 말한다: "응용 프로그램의 라이선스로 인해 사용자가 라이브러리를 수정하고 코드를 리버스 엔지니어링하여 이러한 수정 사항을 디버깅 할 수 있어야한다."<sup>356)</sup>

그래서 우리는 이러한 반 공식의 문서조차도 우리와 매우하게 주장하고 있다고 할 수 있다. 이 텍스트에는 단 하나의 작은 문구가 있다 : "LGPL 섹션 6"을 한 문장으로 요약하면 "(중략) 라이브러리의 새로운 버전이 응용 프로그램과 연결될 수 있게 하고, 리버스 엔지니어링을 통해 이것을 디버그 하도록 허용한다." LGPL 섹션 6의 첫 번째 문장에는 복잡한 조건이 포함되어 있다고 간주하지 않는다.

LGPL2-RefEng-Sentence는 우리가 증명할 수 있는 것처럼, (a) 라이선스의 일부가 디버깅을 위해 리버스 엔지니어링을 허용하는 경우에만 라이브러리의 일부를 포함하는 저작물을 배포 할 수 있음을 의미한다.<sup>357)</sup> 그러나 우리가 볼 수 있듯 응용 프로그램에 실제로 라이브러리의 일부가 포함되어 있는지 여부를 결정하기 위해 LGPL의 섹션 5에서 정의 된 제한을 추가로 고려해야한다.<sup>358)</sup> LGPL-v2의 §5가 제한하는 것보다 많은 라이브러리 요소가 포함된 경우에만 응용 프로그램의 라이선스가 응용 프로그램을 리버스 엔지니어링 할 수 있어야한다.

우리의 분석이 LGPL 정신에 부합한다는 사실은 또한 LGPL을 직접적으로 고려한다는 것에서 드러날 수 있다.

LGPL-v2는 목표를 명확하게 설명한다. LGPL은 커뮤니티가 LGPL 라이브러리를 "사실상의 표준이 되게" 할 수 있게 하고자 한다. 그리고 LGPL은 "이 목표를 달성하기 위해서는 비 자유 프로그램이 라이브러리를 사용할 수 있어야한다"라는 것을 알고 있다. 왜냐하면 "(중략) 비 자유 프로그램에서 특정 라이브러리를 사용할 수 있는 허가로 인해 더 많은 사람들이 많은 자유 소프트웨어를 사용할 수 있기" 때문이다. 하지만 LGPL은 이러한 맥락에서 "약소 일반 공중 사용 허가서(Lesser General Public License)는 사용자의 자유를 덜 보호하지만 라이브러리와 연결된 프로그램의 사용자는 라이브러리의 수정 버전을 사용하여 해당 프로그램을 실행할 자유와 자유가 보장된다."<sup>359)</sup> 고 주장한다.

따라서 파생된 결과가 이 목표를 위반하는지 여부는 파생물을 마지막으로 체크하여 확인할 수 있다. 만약 그렇다면 우리는 엄청난 잘못을 저질렀을 것이고 그렇지 않다면, 분석의 일관성을 신뢰할 수 있을 것이다.

만약 라이브러리를 이용하는 저작물을 이산집합 동적링크 또는 결합 가능한 파일들의 형태로 받았고 만약 그러한 이유로 제공자가 그가 제공한 파일이 링커와 필요한 동적 링크 가능한 라이브러리를 제공해야 하는 타겟 시스템에 링크된다는 것을 가정했다면 업데이트 된 버전으로 동적으로 링크된 라이브러리를

356) cf. id., lc., wp..

357) -> p. 69

358) -> p. 73

359) cf. Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp., preamble, emphasis KR.

자유롭게 체계적으로 대체 할 수 있다.<sup>360)</sup> 그리고 최신 버전의 라이브러리가 정의된 인터페이스와 선언된 인터페이스를 보존하는 한 이를 성공적으로 수행 할 수 있다. 그것이 바로 LGPL-v2가 보장하고자 하는 것이다.

다른 모든 경우에는 리버스 엔지니어링 허가를 받아야하거나 소스코드에 직접 액세스 할 수 있어야한다. 따라서 특히 귀하가 정적으로 링크된 패키지를 받았다면 해당 툴과 기술을 사용하여 라이브러리의 임베디드 버전을 최신 버전으로 대체 할 수 있다. 그러므로 또한 우리 해석의 두 번째 부분은 LGPL-v2의 정신을 존중한다.

그래서 결국 모든 것이 괜찮다고 말할 수 있다. LGPL2-RevEng-Rule은 라이브러리의 일부라는 의미와 함께 LGPL2-RevEng-Sentence의 의미를 입증할 뿐만 아니라 전문에 있는 그대로 LGPL-v2의 정신과 목적을 설명하고 있다.

### 3.4.2 LGPL-3.0의 리버스 엔지니어링

라이선스 텍스트의 의미를 성공적으로 파헤쳐 낼 수 있었던 경험을 토대로 리버스 엔지니어링을 언급하는 LGPL3-RevEng-Sentence를 이해하는데 걸리는 시간을 단축 할 수 있다.

"귀하가 정하는 조건하에 결합된 저작물을 전달할 수 있으며 결합된 저작물에 포함된 라이브러리의 일부분에 대한 수정과 그러한 수정에 대한 디버깅을 위한 리버스 엔지니어링을 제한하지 않으면서 효과적으로 다음을 수행할 수 있다. (중략)"<sup>361)</sup>

모호성 제거 방법을 다시 사용하여 다음 텍스트를 이용해 LGPL3-RevEng-Sentence의 의미를 예를 들 수 있다.

( [Θ :]  
( You compliantly distribute a Combined Work  
under terms of your choice  
( (that together effectively, do not restrict modification of  
the portions of the Library contained in the Combined Work)  
AND  
(that together effectively, do not restrict reverse  
engineering for debugging modifications of the portions  
of the Library contained in the Combined Work)  
)  
IF

360) 예를 들어, GNU/Linux에서는 동적으로 링크 가능한 새 버전의 라이브러리를 lib 디렉토리에 복사하고 기존 링크를 최신 버전을 가리키는 버전으로 바꿔야한다. 때로는 ld.so.conf 파일을 확인하고 ldconfig 도구를 호출해야한다.

361) cf. Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp., §4. 문장의 끝 부분에 있는 줄임표는 여기서 다루지는 않지만 이 문장의 통합된 부분으로 간주되어야 하는 일련의 저작물입니다.

[ $\Omega$  :]

( you also do each of the following [...])

)

하지만 이제는 단순히 실행된 논리적 직렬화로 인해 문제가 발생한다.

만약 직렬화된 ( $\Theta$  IF  $\Omega$ )를 ( $\Omega \rightarrow \Theta$ )로 부정논법에 따라  $\Theta$ 를 존중하지 않는 것으로부터  $\Omega$ 를 인식하도록 허용되지 않는다. 다른 말로하면, 우리는 생략 부호에 의해 다루어진 단 하나의 작업도하지 않게 되는데 그것은 어리석은 결과이다.

만약 직렬화된 ( $\Theta$  IF  $\Omega$ )를 ( $\Omega \rightarrow \Theta$ )로 부정논법에 따라  $\Theta$ 를 한다면  $\Omega$  또한 수행해야만 한다. 그리고 부정논법에 따라  $\Omega$ 를 존중하지 않는 것으로부터  $\Theta$ 를 절대 수행해서는 안된다. 그러나 안타깝게도 우리는 수정 작업을 제한하는 조항과 이러한 기술을 제한하지 않고 리버스 엔지니어링하는 조건에서 결합된 저작물을 배포함으로써 이 두 번째 저지를 존중 할 수 있지만 이는 또다시 어리석은 결과라고 할 수 있다.

분명히 직관적으로 명확한 읽기를 기반으로 하는 간단한 직렬화에 실패한다. 사실, LGPL3-RevEng-Sentence는 보다 정교한 기본 구조를 가져야한다. 논리적으로 직렬화해야만 하는 양식으로, 수정사항을 제한하지 않고 요구사항을 통합하여 실제 상황을 트리거 할 수 있다. 따라서 문장의 의미는 LGPL3-RevEng-Rule을 통해 논리적으로 설명 될 수 있다.

( [ $\Sigma$  :] )

( You complaingly distribute a Combined Work  
under the terms of your choice

)

->

( [ $\Gamma$  :] )

( the terms of your choice together effectively do  
not restrict modification of the portions of the  
Library contained in the Combined Work)

^ [ $\Delta$  :] )

( the terms of your choice together effectively, do  
not restrict reverse engineering for debugging  
modifications of the portions of the Library  
contained in the Combined Work)

^ [ $\Omega$  :] )

( you also do each of the following [...])

) )

이 LGPL3-RevEng-Rule은 실제로 결합된 저작물을 어떻게 준수하는지를 성공적으로 규정한다.

- 결합저작물을 준수하여 배포하고자 한다면  $\Gamma$ ,  $\Delta$  그리고  $\Omega$ 의 모든 개별 부분들을 존중해야 한다.<sup>362)</sup>
- 요구 사항  $\Gamma$ ,  $\Delta$  중 하나 또는  $\Omega$ 의 부분 중 하나를 존중하지 않는다면 결합 저작물을 배포하지 않는다는 것이다.<sup>363)</sup>

이제 GPLv3이 우리를 강요하지 않으며, 모든 면에서 리버스 엔지니어링을 방해하지 않는다는 것을 직접 볼 수 있다. 필요한 리버스 엔지니어링은 수정에 대한 디버깅을 지원하는 목적으로 제한되며 라이브러리의 일부를 포함하는 결합된 작업에 초점을 맞춘다. 즉, 이 조항으로 다른 리버스 엔지니어링 목적을 방해하거나, 결합된 저작물로 규정 될 수 없거나 라이브러리의 일부를 포함하지 않는 다른 형태의 리버스 엔지니어링을 제한 할 수 있다. 따라서 가장 중요한 질문은 GPLv3이 "결합된 저작물"에 대해 말하면 무엇을 의미하는지에 있다. 두 번째 질문은 GPLv3이 라이브러리의 일부로 규정한 것은 무엇인가이다.

다행히도 GPLv3는 다음과 같이 분명하게 기술하고 있다. "결합된 저작물"은 응용프로그램을 라이브러리와 결합하거나 링크하여 생성된 저작물이다.<sup>364)</sup> GPLv2 분석을 통해 우리는 라이브러리를 사용하는 저작물이 기술적으로 라이브러리와 링크되거나 결합되는 방법을 알고 있다.

- 라이브러리를 사용하여 라이브러리에서 저작물로 코드를 복사하면<sup>365)</sup> 라이브러리를 사용하는 저작물에는 실제로 라이브러리의 일부가 포함된다.<sup>366)</sup>
- 스크립트 언어 기반 응용프로그램과 라이브러리를 결합하면 결과를 나타내는 응용프로그램에 라이브러리의 일부가 포함될 수 있다. 그러나 스크립트 코드는 개발 과정에서 배포되므로 직접 이해 할 수 있기 때문에 세부적인 내용은 리버스 엔지니어링과 관련하여 무시 될 수 있다.<sup>367)</sup>
- 자바 클래스와 라이브러리를 통합된 준 정적 링크 된 패키지로 결합하면 결과 패키지에 이미 기능적으로 필요한 라이브러리 코드가 모두 포함된다.<sup>368)</sup>
- Java 클래스를 참조된 라이브러리 클래스와 결합하지 않고 컴파일하면 컴파일 된 클래스는 최소한 라이브러리에 의해 선언된 식별자를 포함한다.<sup>369)</sup>
- C/C ++ 파일 또는 클래스를 컴파일하고 참조된 라이브러리와 정적으로 링크하면 결과로 생성된 실행 파일에는 실제로 사용된 모든 라이브러리의 기능 관련 코드가 모두 포함된다.<sup>370)</sup>
- C/C ++ 파일 또는 클래스를 참조된 라이브러리에 연결하지 않고 컴파일하면 결과 오브젝트 파일이 다른 시스템에서 동적으로 링크 될 수 있고 라이브러리에 의해 식별자가 포함되거나 일부 인라인 함수 또는 매크로를 제거하여 주입되는 기능적인 코드를 포함한다.<sup>371)</sup>

---

362) 긍정논법에 의해, 이 경우 특히 조항에서 "결합된 저작물에 포함된 라이브러리 부분의 수정을 디버깅하기 위해 리버스 엔지니어링을 제한해서는 안된다."

363) 부정논법에 의해, "결합된 저작물에 포함된 라이브러리 부분의 수정을 디버깅하기 위해 리버스 엔지니어링을 제한하는 경우" 결합된 저작물을 배포를 준수하지 않는다.

364) cf. Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp., §0.

365) GPLv3는 라이브러리를 "애플리케이션"으로 사용하는 저작물을 지정하며 "라이브러리가 제공하는 인터페이스를 사용함"을 정의한다. (cf. id., ibid.).

366) -> p. 71

367) -> p. 73

368) -> p. 73

369) -> p. 75

370) -> p. 73

371) -> p. 75

전반적으로 상황은 다음과 같다. LGPL3-RevEng-Rule은 결합된 저작물에 포함된 라이브러리의 일부분을 리버스 엔지니어링 할 수 있어야한다고 말한다. LGPL3은 또한 결합된 저작물이 단순히 라이브러리(응용 프로그램)와 라이브러리를 사용하는 저작물을 기술적으로 결합한 결과임을 추가로 명시하고 있다. 마지막으로, (a) 두 구성 요소를 정적으로 결합하면 결과로 결합된 저작물에 라이브러리의 일부가 포함된다는 사실을 설명하고,<sup>372)</sup> (b) 두 부분을 동적으로 결합 가능한 구성 요소로 준비하는 경우 결과물에 이미 라이브러리의 일부가 포함되어 있는지 여부를 명확히 해야한다.

LGPL-v2와 마찬가지로 LGPL-v3은 우리가 철저히 고려해야할 언어 연계성에 대해 §3에서 이 질문에 답하고 있다.

응용프로그램의 오브젝트 코드 형식은 라이브러리의 일부분인 헤더 파일의 자료를 통합 할 수 있다. 귀하는 귀하가 선택한 조건에 따라 해당 오브젝트 코드를 다음과 같은 경우 전달할 수 있다. [만약 포함된 자료가 수치 매개변수, 데이터 구조 레이아웃이나 접근자 또는 (열 줄 이하의 작은) 매크로, 인라인 함수나 템플릿에 국한되지 않는다면] 다음 두 개의 사항을 준수했을 경우이다. a) 오브젝트 코드의 각 사본에 라이브러리가 사용되고 그 라이브러리와 라이브러리의 사용이 본 라이선스의 적용을 받는다는 사실을 눈에 띄게 알려주어라. b) 객체 코드를 GNU GPL 및 이 라이선스 문서의 사본과 함께 제공해라.<sup>373)</sup>

이 단락의 첫 번째 문장은 컴파일 되고 사용된 라이브러리에 연결되지 않지만 라이브러리의 일부를 포함할 수 있는 객체 파일 만을 위해 만들어졌다는 것을 알려준다. 두 번째 문장은 이러한 객체 파일의 배포를 규제하며 논리적으로 나열할 수 있다.

( [Λ :] )

( You compliantly distribute object code [incorporating material from the Library] under terms of your choice )

->

[Ξ :] )

( [ω :] )

( the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates [ten or fewer lines in length] )

->

( [α :] ( you do [a] ... )  
^ [β :] ( you do [b] ... )

) ) )

---

372) 그래서 LGPL3-RevEng-Rule을 트리거한다.

373) 그래서 이것이 LGPL3-RevEng-Rule을 트리거한다.

이 LGPL3 문장은 주된 규칙 ( $(\Lambda \rightarrow \Xi)$ )을 포함하고, 결론  $\Xi$  자체는 임베디드 서브 룰 ( $\omega \rightarrow (\alpha \wedge \beta)$ )의 형태를 갖는다.

첫째, 주 규칙은 객체 규칙을 준수하도록 배포하려는 경우 하위 규칙을 따르게 한다.<sup>374)</sup> 둘째, 주 규칙은 하위 규칙을 따르지 않는다면 객체 코드를 배포하지 않는다고 말한다.<sup>375)</sup>

하위 규칙을 따르는 두 가지 방법과 이를 준수하지 않는 한 가지 방법이 있다.

- 오브젝트 코드가 제한보다 라이브러리의 더 많은 요소 또는 더 큰 요소를 포함한다면  $\alpha$ 와  $\beta$ 를 수행하면 하위 규칙을 따른다.<sup>376)</sup>
- 오브젝트 코드에 라이브러리의 요소가 최대한도까지 포함되어있는 경우 추가 작업을 수행할 필요 없이 하위 규칙을 따른다.<sup>377)</sup>
- 그러나 오브젝트 코드가 제한 사항보다 라이브러리의 더 많은 요소 또는 더 큰 요소를 포함하고  $\alpha$  또는  $\beta$  하지 않거나 한다면 하위 규칙을 따르지 않는다.<sup>378)</sup>

따라서 결국 소프트웨어 프로그래밍에 관한 추가적인 오브젝트 코드 지정 및 알려진 경험적 배경 지식을 기반으로 LGPL3-RevEng-Rule은 LGPL2-RevEng-Rule과 동일한 결과를 제공한다.<sup>379)</sup>

- 만약 [A] 원작자가 설계한 그대로의 인터페이스를 포함하는 표준 버전의 라이브러리를 기반으로 하여 라이브러리를 이용하는 저작물을 개발한다면, [B] 이 라이브러리를 사용하여 사용자의 저작물을 동적으로 링크된 또는 결합된 파일들의 분리된 세트로서 컴파일 하는 경우, [C] 업스트림 승인된 인터페이스로 보존하는 표준 컴파일 방법만 사용한다면,<sup>380)</sup> [D] 생성된 링크 해제된 오브젝트 코드 또는 바이트 코드 파일이 실행파일로 링크되기 전에 배포하는 경우. LGPL-v3 라이선스가 적용된 라이브러리를 따라서 사용자가 리버스 엔지니어링을 허용해야만 할 필요는 없다.
- 그러한 라이브러리를 사용하여 저작물을 배포하는 다른 모든 경우에는 이 라이브러리를 사용하는 저작물의 리버스 엔지니어링을 허용해야한다 - 특히 ...
  - 라이브러리와 라이브러리를 사용하는 저작물을 정적으로 링크된 프로그램 또는 라이브러리와 라이브러리 자체를 사용하는 저작물을 포함하는 통합 패키지로 함께 배포하는 경우.<sup>381)</sup>
  - 수동으로 복제된 라이브러리의 일부를 포함하는 저작물을 배포했을 경우.

---

374) 긍정 논법에 의해 ( $\Lambda \rightarrow \Xi$ )

375) 긍정 논법에 의해 ( $\Lambda \rightarrow \Xi$ )

376) 긍정 논법에 의해  $\omega \rightarrow (\alpha \wedge \beta)$

377) 함의의 정의를 따름. 이 하부 규칙의 전제가 거짓이라면, 하부 규칙은 전체적으로 진실이므로 존중된다.

378) 함의의 정의를 따름. 전제가 사실이고 결론이 거짓이라면 전체로써의 함의는 거짓이다..

379) -> p. 76

380) 따라서 LGPL-v3 제한을 초과하지 않아야한다.

381) 이는 스크립트 언어 기반 응용 프로그램을 이해하기 위해 리버스 엔지니어링 허가가 필요 없다는 사실에도 불구하고 스크립트 언어 기반 프로그램이나 패키지를 배포하는 경우에도 마찬가지이다.

### 3.4.3 기타 오픈소스 라이선스의 리버스 엔지니어링

나머지는 간단하다. 첫째, 우리가 고려하는 다른 오픈소스 라이선스 중 '리버스 엔지니어링'이라는 문구가 포함되어 있지 않다는 것을<sup>382)</sup> 확인 할 수 있을 것이다. 더욱이, 심지어 단일 단어 중 하나를 포함하지도 않는다.<sup>383)</sup> 따라서 우리는 가장 중요한 다른 오픈소스 라이선스가 간접적으로 리버스 엔지니어링 허가를 요구 할 수 있다고 추측 할 수 있을 것이다. 두 번째로, 우리는 이미 스크립트 코드를 배포하는 것이 리버스 엔지니어링을 허용하는 것이라는 것을 알고 있으며 이로 인해 관련성이 없어진다. 스크립트 언어를 알고 있다면 스크립트 코드를 직접 읽고 이해 할 수 있다.<sup>384)</sup> 셋째, 강력한 카피레프트의 정의에서 강력한 카피 레프트 라이선스 하에 라이선스 된 소프트웨어의 배포는 리버스 엔지니어링 허가가 중요하지 않은데 이는 카피 레프트 라이선스 하에 허가된 라이브러리를 사용하는 저작물의 소스코드도 액세스 가능해야하기 때문이다.<sup>385)</sup>

따라서 소프트웨어의 일부가 바이너리 또는 바이트 코드의 형태로 배포되는 경우, 허가된 오픈소스 라이선스 또는 약한 카피 레프트 라이선스 하에 허가된 라이브러리를 사용하는 경우를 고려해야한다고 결론 내릴 수 있다.

허가된 라이선스 또는 약한 카피 레프트 라이선스의 정의에서 우리는 이미 오픈소스 컴포넌트의 라이선스가 오픈소스 소프트웨어 컴포넌트를 사용하는 중요한 저작물을 사용하기 위한 허가 또는 금지에 직접적으로 영향을 미치지 않는다는 것을 알고 있다.<sup>386)</sup>

그래서 아직 바이너리나 바이트 코드 파일로 링크되지 않았지만 만약 그러한 저작물을 동적 링크가 가능한 형태로 배포한다면 컴포넌트를 사용하는 저작물이 리버스 엔지니어링 될 수 있음을 합리적으로 파생할 수 있는 방법이 없다. 허용적 또는 약한 카피레프트 오픈소스 라이선스는 주로 오픈소스 컴포넌트들에 관한 것이지 컴포넌트를 사용하는 저작물에 대한 것이 아니다. 한 가지 면에서 이 라이선스는 실제로 우리가 사용하고자하는 모든 오픈소스 컴포넌트의 라이선스 텍스트와 저작권 문구를 우리 저작물을 포함하는 분산 패키지에 추가해야한다. 그리고 라이선스들은 우리의 저작물에서 사용하고자하는 오픈소스 컴포넌트에 통합된 라이선스 주장을 수정하는 것을 금지한다.<sup>387)</sup> 그러나 다른 한 편 그리고 관대하거나 약한 카피레프트인 라이선스들에 따라 이러한 오픈소스 라이선스에 의해 확립된 소프트웨어의 사용, 연구, 수정 또는 배포의 자유는 오픈소스 컴포넌트를 사용하는 저작물이 아닌 오픈소스 컴포넌트 그 자체에만 해당된다. 따라서 이러한 컴포넌트나 표준 컴파일 및 계산 방법에 따라 사용 작업과 연결되거나 결합되지 않는 한 실제로 이 컴포넌트를 사용하는 작업을 연구하거나 수정할 필요 없이 연구하거나 수정할 수 있다.<sup>388)</sup>

382) -> p. 60

383) 이 부정적인 진술은 (a) OSI 홈페이지에서 모든 라이선스들을 다운로드 받고 (<http://opensource.org/licenses/alphabetical>) (b) grep -i "engineering" \* 과 grep -i "reverse" \* 각 명령어를 라이선스 파일이 저장된 곳에서 수행한다: grep 명령은 LGPL 에서만 reverse 와 engineering 이라는 단어를 발견할 것이다.

384) -> p. 73

385) cf. Stallman: What is Copyleft?, 1996, wp.

386) cf. Reincke, Karsten, Greg sharpe, a. contributors: Open Source License Compendium. How to Archive Open Source License Compliance; 2015 <URL: <http://www.oslic.org/releases/oslic.pdf>> - 참조 다운: 2015-01-20, pp.20ff..

387) 이러한 요구사항들은 우리가 여기에서 고려하는 모든 오픈소스 라이선스의 일부이다. 자세한 내용은 다음을 참조. cf. id, l.c. pp. 6장

388) 구성 요소의 사용권이 사용 저작물에서도 사용된다는 것을 추론하는 유일한 방법은 사용하는 저작물이 최소한 라

다른 면에서 정적으로 링크 된 바이너리 또는 바이트 코드 파일을 사용하여 저작물을 컴포넌트를 사용하여 배포하면 (이미 필요한 모든 컴포넌트를 포함하고<sup>390)</sup> 직접 실행할 수 있음), 모든 오픈소스 라이선스 텍스트와 모든 저작권 라인을 패키지에 추가해야하며 우리는 원본 오픈소스에 존재하는 라이선스 조항 중 하나를 수정할 수 없다.<sup>391)</sup> 따라서 오픈소스 컴포넌트에 정적으로 링크되거나 결합된 소프트웨어를 배포하는 경우 오픈소스 컴포넌트 자체를 사용하고 수정하는 자유는 존속 할 수 있다. 따라서 바이너리 또는 바이트 코드 파일을 편집해야하더라도 정적으로 링크된 작업의 수신자는 내장된 오픈소스 컴포넌트를 수정할 수 있다. 반대로 바이너리 파일을 역으로 개발하는 방법은 리버스 엔지니어링이라고 한다. 따라서 오픈소스 라이선스 컴포넌트를 사용하여 정적으로 링크된 저작물을 배포하는 경우라면 받는 사람이 간접적으로 전체 제품을 리버스 엔지니어링 할 수 있는 권한을 갖게 될지도 모른다. 그리고 정적으로 링크된 라이브러리가 관대하거나 약한 카피 레프트 라이선스로 허가 된 경우에도 두려움을 가져야만 한다.

결과를 다시 요약하면 다음과 같다.

- 만약 [A] 원작자가 설계한 그대로의 인터페이스를 포함하는 표준 버전의 라이브러리를 기반으로 하여 라이브러리를 이용하는 저작물을 개발한다면, [B] 이 라이브러리를 사용하여 사용자의 저작물을 동적으로 링크된 또는 결합된 파일들의 분리된 세트로서 컴파일 하는 경우, [C] 업스트림 승인된 인터페이스로 보존하는 표준 컴파일 방법만 사용한다면, [D] 생성된 링크 해제된 오브젝트 코드 또는 바이트 코드 파일이 실행파일로 링크되기 전에 배포하는 경우. 어떠한 허용적 또는 약한 카피레프트 라이선스 조건을 가지고 있는 라이브러리라도 사용자가 리버스 엔지니어링을 허용해야 할 필요가 없다.
- 그러한 라이브러리를 사용하여 저작물을 배포하는 다른 경우에는 이 라이브러리를 사용하는 저작물의 리버스 엔지니어링을 간접적으로 허용하게 될 수 있다는 두려움을 갖고 있어야 한다. - 특히 ...
  - 라이브러리와 라이브러리를 사용하는 저작물을 정적으로 링크된 프로그램 또는 라이브러리와 라이브러리 자체를 사용하는 저작물을 포함하는 통합 패키지로 함께 배포하는 경우.<sup>392)</sup>
  - 수동으로 복제된 라이브러리의 일부를 포함하는 저작물을 배포했을 경우.

---

이브러리에 의해 선언되지만 (정의되지는 않은) 인터페이스의 요소 (식별자 등)를 포함해야한다고 주장하는 것이다. 그러므로 적어도 이들 요소는 조사되거나 변경 될 수 있다. 이 과제는 명시적으로 LGPL에 의해 해결된다.<sup>389)</sup> 다행스럽게도, 소프트웨어 라이브러리의 일반적인 특징은 인터페이스에 따라 사용해야하며, 라이브러리 개발자는 라이브러리를 실제로 사용할 수 있도록 설계했다. 따라서 LGPL와 달리 링크되지 않은 바이너리 또는 바이트 코드 파일과 같이 라이선스들이 암암리에 라이브러리의 부분을 포함하는 문제를 명시적으로 해결하지 않은 경우 이들이 표준 방법에 따라 컴파일 되고 표준 인터페이스를 통해 오픈소스 소프트웨어를 사용하게 되면 개발자가 인터페이스를 설계했기 때문에 선언된 인터페이스 요소가 아닌 선언된 인터페이스 요소를 통합하는 것과 같은 요구사항 없이 개발자가 허용한 계산의 본질을 추구해야 한다. 요구사항 없이 이러한 요소를 사용하지 않으려는 경우 다른 인터페이스를 설계한다. 그리고 카피 레프트 효과 또는 리버스 엔지니어링의 허가를 통합하고자한다면 그들은 다른 라이선스를 선택했을 것이다. 하지만 결론은 소프트웨어 라이브러리를 사용하는 표준 방법에 대해서만 적용된다.

390) 선언 된 인터페이스 요소 대신에!

391) cf. Reincke, Sharpe, a. other contributors: OSLiC, 2015, pp. 6장..

392) 이는 스크립트 언어 기반 응용 프로그램을 이해하기위해 리버스 엔지니어링 허가가 필요없다는 사실에도 불구하고 스크립트 언어 기반 프로그램이나 패키지를 배포하는 경우에도 마찬가지이다.

### 3.4.4 오픈소스 라이선스의 리버스 엔지니어링: 개요

따라서 우리는 모든 결과를 하나의 단일 결과로 편집 할 수 있다.

- 만약 [A] 원작자가 설계한 그대로의 인터페이스를 포함하는 표준 버전의 라이브러리를 기반으로 하여 라이브러리를 이용하는 저작물을 개발한다면, [B] 이 라이브러리를 사용하여 사용자의 저작물을 컴파일 하는 경우, [C] 업스트림 승인된 인터페이스를 보존하는 표준 컴파일 방법만 사용한다면,<sup>393)</sup> [D] 생성된 링크 해제된 오브젝트 코드 또는 바이트 코드 파일이 실행파일로 링크되기 전에 배포하는 경우. 오픈소스 라이선스가 적용된 어떠한 라이브러리도<sup>394)</sup> 리버스 엔지니어링을 허용해야만 할 필요는 없다.
- 그러한 라이브러리를 사용하여 저작물을 배포하는 다른 모든 경우에는 아마도 당신의 저작물에 대해 리버스 엔지니어링을 허용해야 할 수 있고, 적어도 이 라이브러리를 사용하는 저작물의 리버스 엔지니어링을 간접적으로 허용하게 될 수 있다는 두려움을 갖고 있어야 한다. - 특히 ...
  - 라이브러리와 라이브러리를 사용하는 저작물을 정적으로 링크된 프로그램 또는 라이브러리와 라이브러리 자체를 사용하는 저작물을 포함하는 통합 패키지로 함께 배포하는 경우.<sup>395)</sup>
  - 수동으로 복제된 라이브러리의 일부를 포함하는 저작물을 배포했을 경우.

그리하여 우리는 원래 다른 뷰포트에서 자신의 작업을 보호하는 문제를 분석한 오픈소스 전문가가 조금 수정된 "엄지손가락 규칙"으로 결과를 재구성 할 수 있다.

- 프로그램을 [독점적으로 유지하고 리버스 엔지니어링으로부터 보호하려는 경우] [오픈소스] 코드를 정적으로 링크하거나 [결합]하지마라.<sup>396)</sup>
- LGPL [뿐만 아니라 모든 오픈소스 코드에] 동적으로 링크하라.<sup>397)</sup>

**q.e.d**

---

393) 라이브러리의 저작권자가 규정한 한도를 초과하지 않아야 한다.

394) -> p. 60

395) 이는 스크립트 언어 기반 응용 프로그램을 이해하기위해 리버스 엔지니어링 허가가 필요없다는 사실에도 불구하고 스크립트 언어 기반 프로그램이나 패키지를 배포하는 경우에도 마찬가지이다.

396) cf. Ilardi: Common OSS License Problems, 2010, pp. 6: bracketed text KR..

397) cf. id., ibid.

### **3.5 부가사항: 라이선스 호환성 문제 [추후 보완]**

여기서는 종종 소홀해지거나 단지 피상적으로 다루어진 다른 종류의 라이선스가 부여된 소프트웨어를 결합하는 것과 같은 문제에 대해 논의합니다. 우리는 자유 소프트웨어 재단의 배제 목록을 시사할 것이다. 우리는 Eclipse / GPL 플러그인 문제를 시사할 것이고 커널이 완전한 Android를 GPL로 라이선스해야 하는지에 대한 최근 논의를 언급 할 것이다. 그리고 마침내 최근 Jaeger와 Metzger가 결합 매트릭스를 제시하여 발표한 내용을 짧고 명료하게 분석할 것이다. 또한 다음 질문이 단순하게 답변될 수 있다고 주장할 것이다. 온탑 개발물의 라이선스 하에 라이선스된 두 개의 라이브러리를 임베드하고 이 두 개의 라이선스가 파생저작물에 대해 다른 라이선스를 부여하기를 요구하는 경우에만 문제가 발생한다. 우리가 설명할 다른 경우에는 문제가 없다.

### **3.6 부가사항: 오픈소스 소프트웨어와 수익 [추후 보완]**

여기서는 수익과 오픈소스가 아무런 문제를 발생시키지 않을 방법을 간략히 논의 할 것이다.

## 4 오픈소스 사용사례: 개념과 분류체계

이 장에서는 오픈소스 사용 사례에 대한 개념을 과업 목록(*to-do list*)의 분류체계로서 확립시킬 것이다. 오픈소스 사용 사례에서 처한 특별한 환경에서 요구되는 라이선스의 특정 조건은 해당 과업 목록을 준수함으로써 충족된다. 또한 이 장에서는 오픈소스 사용 사례의 분류체계(*taxonomy*)를 소개하고 나중에 이 분류체계를 통해 오픈소스 사용 사례 파인더(*Open Source Use Case Finder*)를 구성할 것이다.

다음의 소개 글을 제시하면서 우리가 생각한 개념을 간략히 설명할 수 있다. 오픈소스 소프트웨어 사용 권한은 오픈소스 라이선스에서 요구하고 있는 과업에 의해 좌우된다. 상업용 라이선스와 반대로 금전을 지불하여 오픈소스 소프트웨어 일부분을 사용할 권한을 얻을 수는 없다. 소프트웨어 사용 권한은 판매될 수 없다는 점은 오픈소스 정의(Open Source Definition)에 포함되어 있는 내용이다. 오픈소스 소프트웨어 정의에서 처음에는 오픈소스 라이선스는 “소프트웨어 배포물 집합의 컴포넌트로서 소프트웨어를 판매하거나 무료로 제공하지 않도록 제한” 할 수 있다고 명시되어 있으며, 두 번째에서는 오픈소스 라이선스에 의해 “그런 판매에 대해 로열티 또는 다른 수수료가 요구되지 않는다”라는 동일한 맥락으로 부연 설명하고 있다.<sup>398)</sup>

그러나 오픈소스에 대한 보답으로써 아무런 서비스를 제공하지 않아도 오픈소스 소프트웨어 사용이 저절로 허용된다는 결론을 내리는 것은 올바르지 않다. 일반적으로 소프트웨어 사용 권한을 얻으려면 무언가를 해야 한다. 즉, 오픈소스 소프트웨어는 ‘행위로써 지불’하는 개념이 적용된다. 이에 따라, 오픈소스 라이선스에서 사용자는 라이선스를 준수하기 위해서 몇 가지 작업을 이행해야 하는 특정 상황이 기술된다. 따라서 라이선스 조건을 충족하기 위한 과업 목록이 제공된다면 이 작업 및 상황을 고려해야 한다.

실제로 그런 상황은 단선적이거나 간단한 것이 아니다. 여기에는 토큰의 클래스로 그룹화 할 수 있는 (때로는 콘텍스트에 대해 민감한) 조건들의 조합이 포함된다. 토큰의 클래스는 어플리케이션 또는 라이브러리 같은 소프트웨어의 특성을 나타낼 수 있다. 또는 ‘단독 소프트웨어 사용’ 또는 ‘제3자에게 소프트웨어 배포’와 같은 소프트웨어 사용 상황을 나타낼 수 있다.

말미에는 특정 오픈소스 사용 사례(OSUC, Open Source Use Case)를 결정하고자 한다. 그리고 OSUC와 고려된 오픈소스 라이선스에 대해 이 상황에서 라이선스를 충족하는 활동 리스트를 제공하고자 한다.<sup>399)</sup>

오픈소스 사용 사례는 특정 사용 상황을 설명하는 일련의 토큰으로서 간주된다. 이런 이유로, 이 토큰의 타당한 조합(오픈소스 사용 사례)을 결정하기 전에 먼저 적합한 토큰 클래스를 지정해야 한다. 마지막으로, 이 토큰에 근거하여 트리 형태로 분류체계를 생성시킬 것이다. 이 트리는 다음 장에서 제공되는 오픈소스 사용 사례 파인더의 기초가 되고 단지 몇 가지 질문과 답을 평가하여 이 트리는 구체적 OSUC로 여러분을 이끌 것이다.

오픈소스 소프트웨어 라이선스 상황에 적합한 몇 가지 토큰이 있는데 다음과 같다.

398) cf. Open Source Initiative: The Open Source Definition, 2012, wp §1.

399) 다행히 때때로 하나의 작업 목록은 하나 이상의 적용사례 조건을 만족시킨다.

- **오픈소스 소프트웨어 유형(type):** 한편으로는 스니펫, 모듈, 라이브러리 및 플러그인이 다른 한편으로는 독립적 어플리케이션, 프로그램 및 서버가 고려된다. 첫 번째 세트에 대해서는 '스니몰리(snimoli(s))', 두 번째 세트에 대해서는 '프로앱스(proapse(s))'라는 단어를 고려할 것이다. 보통의 의미뿐만이 아니라 광범위한 의미로도 라이브러리와 어플리케이션에 대해서 설명하므로 이는 필요하다.<sup>400)</sup> 좀 더 구체적으로, 여러분이 사용하고 싶은 오픈소스 소프트웨어가 포함 가능한 스니펫 코드, 링크 가능한 모듈 또는 라이브러리, 로드 가능한 플러그인인지, 또는 실행이나 처리가 가능한 독립적 어플리케이션이거나 서버인지 질문하게 될 것이다. 첫 번째 경우에는, '스니몰리(snimoli)'라는 대답을 해야 하고 두 번째 경우에는, '프로앱스(proapse)'라는 대답이 되어야 한다.
- **오픈소스 소프트웨어 상태(state):** 오픈소스 소프트웨어를 받은 것과 동일하게 사용될 수 있거나 사용 전에 수정될 수도 있다. 좀 더 구체적으로 얘기하자면 받은 대로 오픈소스 소프트웨어를 그냥 두고 싶은지 아니면 제3자에게 사용 또는 배포하기 전에 수정하고 싶은지 질문할 것이다. 첫 번째 경우에는 '비수정(unmodified)' 그리고 두 번째 경우에는 '수정(modified)'이라는 대답이 되어야 한다.
- **오픈소스 소프트웨어 사용 상황(context):** 준비된 어플리케이션으로써 받은 오픈소스 소프트웨어를 사용할 수 있다. 반면에, 받은 오픈소스를 구성 요소 중 하나로서 더 큰 어플리케이션에 내장시킬 수도 있다. 보다 구체적으로 여러분이 오픈소스 소프트웨어를 독립적 소프트웨어로 사용 중인지 또는 더 크고 더 복잡한 소프트웨어의 삽입된 부분으로서 사용 중인지 질문할 것이다. 첫 번째 경우에는 "독립(independent)" 그리고 두 번째 경우에는 '임베디드(embedded)'라는 대답이 되어야 한다.
- **오픈소스 소프트웨어를 받는 사람(recipient):** 단독(for yourself)으로 받은 오픈소스 소프트웨어를 사용하고자 할 수 있다. 기타의 경우, 여러분은 다른 사람들에게 소프트웨어를 인계할 의도를 가질 수 있다. 보다 구체적으로, 여러분이 단독으로 오픈소스 소프트웨어를 사용하려 하는지 또는 제3자에게도 배포/재배포할 계획인지를 질문할 것이다. 첫 번째 경우에는 '단독사용(4yourself)', 두 번째 경우에는 '타인배포(2others)'라는 대답이 되어야 한다.
- **배포된 파일 형태(form):** 대부분의 라이선스에서는 소스로서 소프트웨어를 배포하는 것과 바이너리로서 파일을 배포하는 것과 구분을 짓는다. 이 경우 여러분이 바이너리 형태로 또는 소스코드로서 소프트웨어를 배포하려 하는지 질문할 것이다. 첫 번째 경우에는 '바이너리(binary)', 두 번째에는 '소스코드(source)'라는 대답이 되어야 한다.
- **실행된 프로그램의 io액세스(ioAccess):** 적어도 하나의 라이선스는 해당 데이터에 대한 로컬 액세스만하는 오픈소스 기반 작업과 해당 데이터를 인터넷을 통해 배포하는 오픈소스 기반 작업을 구별한다. 첫번째 경우 대답은 '로컬전용(onlyLocally)'이고 두 번째의 경우 '인터넷이용(viaInternet)'라고 할 수 있다.

400) 새로 소개한 'snimoli'와 'proapse'의 개념은 절대로 가장 품격있는 단어들은 아니다. 따라서 여기에서 원래 이 상으로 이 단어들이 나타내야 하지만 처음에 '어플리케이션'과 '라이브러리'에 대해 설명하였다. 그러나 여기 질문자들의 짜증을 최소화하지 못하였다. 엄격한 의미에서 어플리케이션과 라이브러리를 설명하였다는 점을 매우 자주 상기시켜야만 했다. 최종적으로 자체 표현을 찾고 더 나은 제안을 위해 오픈해 두기로 결정하였다 ;-)

보다 실리적 관점에서 다음과 같이 토큰을 간략히 나타낸다.

- |                              |    |                              |
|------------------------------|----|------------------------------|
| • 유형: : 스니몰리(snimoli)        | 또는 | 타입: : 프로앱스(proapse)          |
| • 상태: : 비수정(unmodified)      | 또는 | 상태: : 수정(modified)           |
| • 사용상황: : 독립(independent)    | 또는 | 사용상황: : 임베디드(embedded)       |
| • 받는사람: : 단독사용(4yourself)    | 또는 | 받는사람: : 타인배포(2others)        |
| • 배포형태: : 바이너리(binaries)     | 또는 | 배포형태: : 소스코드(sources)        |
| • io액세스: : 로컬전용(onlyLocally) | 또는 | io액세스: : 인터넷이용(vialInternet) |

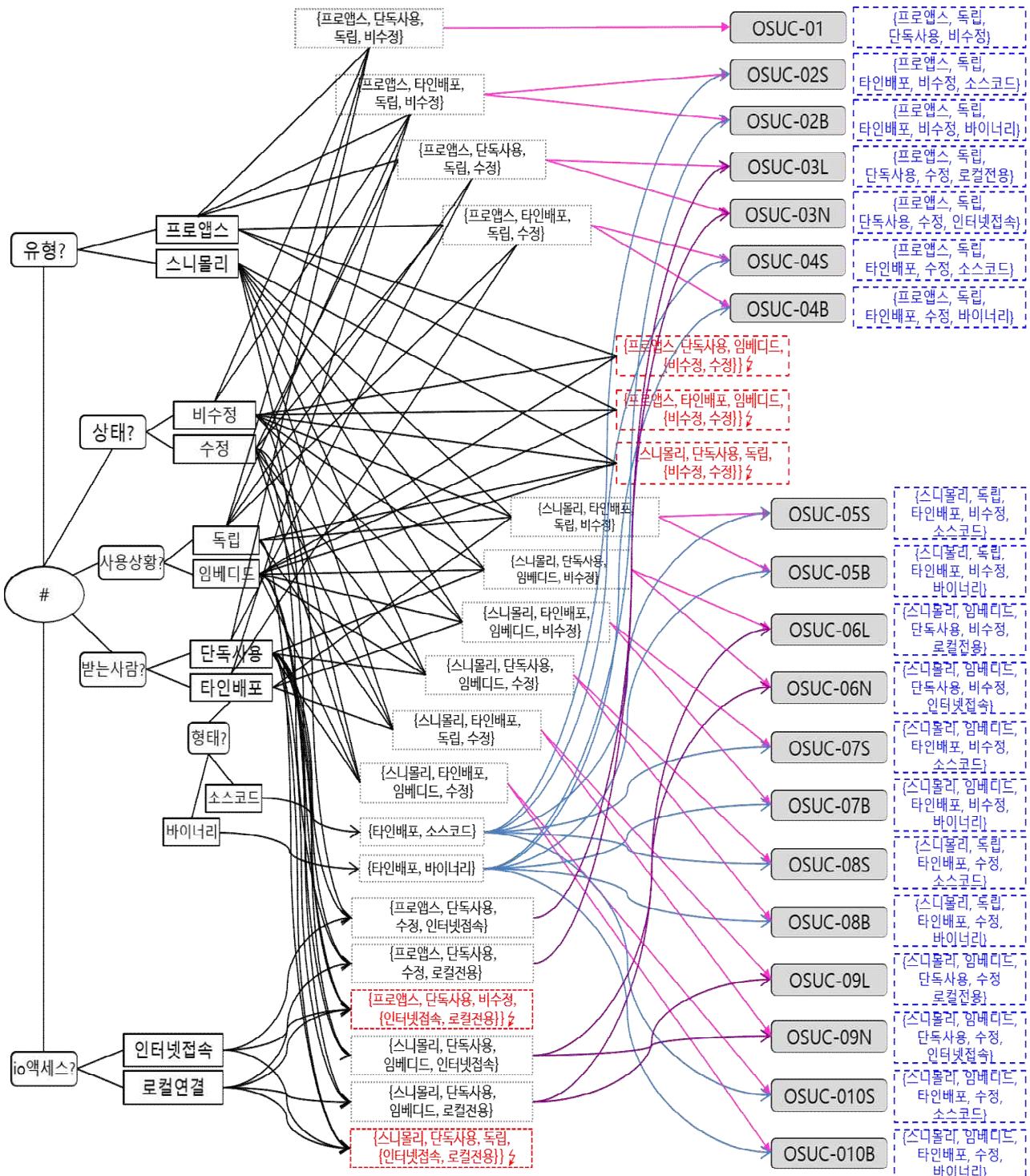
여기에서는 이미 오픈소스 사용 사례를 이 토큰의 조합으로 정의를 내렸었다. 이 모든 클래스의 토큰과 다른 클래스의 토큰을 단지 결합하면<sup>401)</sup>  $2^*2^*2^*2^*2^*2 = 62$  토큰세트 또는 62가지 오픈소스 사용 사례를 얻게 된다. 다행히, 생성된 세트 일부는 경험이나 논리적 관점에서 볼 때 무효하며, 일부 세트는 실제 사용상황에 민감한 유형이다.

1. 사용한 오픈소스 소프트웨어가 프로앱스(독립 프로그램, 어플리케이션 또는 서버)라고 구체화하였다면 여러분의 대답은 소프트웨어가 독립적으로 사용되고 더 큰 유닛으로 다른 컴포넌트와 함께 삽입되지 않는다는 것을 뜻하는 것이다. 그러나 사용한 오픈소스 소프트웨어가 스니몰리(스니펫 코드, 모듈, 플러그인 또는 라이브러리)라고 밝혔다면, 구성된 더 큰 어플리케이션 또는 서버의 삽입된 컴포넌트로서 사용할 수 있거나 제3자에게 배포/재배포하는 경우에만 독립적으로 사용할 수 있다.
2. 여러분이 이미 사용한 오픈소스 소프트웨어가 스니몰리(스니펫 코드, 모듈, 플러그인 또는 라이브러리)이고 스니몰리가 제3자에게 배포하지 않고 단독으로만 사용되어야 한다고 밝혔다면 그 대답은 이 스니몰리는 더 큰 유닛의 삽입된 부분으로 조합되어 사용한다는 것을 뜻하는 것이다. 라이브러리는 다른 어플리케이션 컴포넌트로서 사용하지 않고는 자율적으로 사용할 수 없다. 이 경우 단지 디스크에 자리하여 공간을 차지할 뿐이다.
3. 배포된 파일의 형식을 알아보려면 소프트웨어를 다른 수신자에게 '타인배포'하기로 결정한 경우에만 관련이 있다.
4. ioAccess 유형을 판별자로 사용하는 하나의 라이선스와 관련하여 임베디드 오픈소스 구성 요소를 수정여부에 관계없이 단독사용을 위해 수정된 오픈소스 프로그램을 로컬로 실행하거나 프로그램을 로컬로 실행하는 경우에만 ioAccess의 유형을 알아보는 것과 관련이 있다.

여기서도 복잡하게 들릴 수 있을 있겠지만 이런 제약사항을 우리 스스로 명확히 하기 위해 많은 시간을 소비하였으며, 모든 조합들과 규칙들을 트리로 이동시킨 후 명료해졌다. 다음 다이어그램은 우리의 설명을 개략적으로 나타낸 것이다.<sup>402)</sup>

401) cross product TYPE × STATE × CONTEXT × RECIPIENT × FORM × IOACCESS(유형×상태×상황×받는사람×형태×IO액세스)의 의미에서. OSLiC 일부 초기 버전에서 정적 또는 동적으로 링크하거나 더 큰 생성물에 오픈소스 소프트웨어 (일부)를 문맥상 포함시켜 오픈소스 소프트웨어를 다른 소프트웨어 컴포넌트와 결합 또는 삽입하려 하는지를 질문하였다. 그러나 이 질문 결과로 인해 복잡성을 가중시킬 필요가 없다는 점을 확실히 알게 되었다. 상세 설명 → OSLiC p.xx

402) 각 무효(invalid) 사용 사례(= 토큰 세트) (자세한 내용은 xx쪽)는 ↗로 표시되었고 공집합(empty set) (= Ø)이



된다. 우리는 '무효'라는 말을 약간 모호하게 사용하고 있습니다. 경험적으로 불가능한 경우 값 조합이 잘못되었거나 기능을 결합하는 경우 또는 추가된 기능으로 개념을 하위분류하는 것과 관련이 없는 경우. 특별히:

- Proapse는 주 기능을 포함하는 다른 소프트웨어 장치에 내장 될 수 없다.
- 소프트웨어 라이브러리를 직접 사용하거나 독립적으로 (더 큰 소프트웨어 장치와 함께 사용하지 않고) 사용하는 것은 디스크에 사용되지 않은 바이트 힙이 있는 것과 같다.
- 소스와 바이너리를 구별하는 것은 소프트웨어를 배포하는 경우에만 유효하다.
- 로컬 기반의 io액세스로 실행된 프로그램과 인터넷 기반의 io액세스로 실행된 프로그램을 구별하는 것은 해당 소프트웨어를 실행하는 것을 의미하는 소프트웨어를 사용하는 경우에만 관련이 있다.

## 5 오픈소스 사용사례: 라이선스를 충족시키는 과업 목록(To-Do List) 찾기

이 장에서는 오픈소스 사용 사례 파인더를 제시한다. 이를 통해 '형태'에 의해 수집된 정보를 근거로 과업 목록을 최종적으로 나타내는 오픈소스 사용 사례에 연결되는 잎을 갖는 트리를 검토할 수 있다.

### 5.1 관련 정보 수집을 위한 표준 포맷

	어떤 오픈소스 소프트웨어를 사용하고자 하는가?	
	어떤 오픈소스 라이선스에 의해 배포되고 있는가?	
클래스	질문	응답
유형	(스니펫(snippet) 코드, 링크 가능한 모듈(module), 또는 라이브러리(library), 로드 가능한 플러그인을 포함할 수 있는) 가장 광범위한 의미의 라이브러리를 사용하는 오픈소스 소프트웨어인가. 아니면 실행 가능한 독립 프로그램(program), 어플리케이션(application) 또는 서버(server)인가?	<input type="checkbox"/> 스니몰리(snimoli) <input type="checkbox"/> 프로앱스(proapse)
상태	오픈소스 소프트웨어를 받은 그대로의 비수정 상태로 둘 것인가 아니면 수정판을 작성할 것인가?	<input type="checkbox"/> 비수정(unmodified) <input type="checkbox"/> 수정(modified)
사용상황	오픈소스 소프트웨어를 독립 유닛으로 사용 / 배포할 계획인가 아니면 더 복잡한 소프트웨어의 일부 컴포넌트로 이를 임베디드 할 계획인가?	<input type="checkbox"/> 독립(independent) <input type="checkbox"/> 내장(embedded)
받는 사람	단독으로 오픈소스 소프트웨어를 사용하려고 하는가 아니면 제3자에게 배포/재배포할 계획인가?	<input type="checkbox"/> 단독사용(4yourself) <input type="checkbox"/> 타인배포(2others)
배포형태	오픈소스 기본 저작물을 타인에게 배포/재배포하기를 원할 경우 바이너리 또는 소스코드 중 어디에 중점을 두는가?	<input type="checkbox"/> 바이너리(binary) <input type="checkbox"/> 소스코드(source)
IO액세스	수정된 OS 프로그램을 실행하거나 OS 라이브러리를 사용하여 프로그램을 생성하고 실행하여 오픈소스 소프트웨어를 단독으로 사용하고 이 프로그램의 IO 데이터가 로컬에만 배포되는지 혹은 인터넷을 통해 배포되는가?	<input type="checkbox"/> 로컬전용 <input type="checkbox"/> 인터넷접속

앞서 담론한 바와 같이 일부 무효한 조합들이 존재한다.<sup>403)</sup> 각 클래스에 관한 몇 가지 추가 설명을 제시한다. 요점은 다음과 같다.

**유형(Type):** 일반적인 프로그램 런처로 바이너리 형태를 실행할 수 있다면 그 오픈소스 소프트웨어의 일부분은 프로그램, 어플리케이션 또는 서버이다. 또는 (php, perl, bash 등의 인터프리터로 해석해야만 하는 텍스트파일의 경우에) 파일을 인수(argument) 중 하나로 가져가서 명령어를 실행하는 인터프리터로 스타트 할 수 있는 경우도 마찬가지다.

403) 유형::proapse는 상태::embedded를 제외함; 받는사람::단독사용은 상태::독립과 유형::snimoli의 결합을 제외함; 'mode' 클래스의 값은 상태::embedded를 의미함; 배포형태는 받는사람::타인배포인 경우에만 관련있음; io액세스는 받는사람::단독사용일 경우에만 관련 있음 [세부 사항은 88페이지 참조]. 만약 잘못된 결합에 직면했을 경우, 이에 따른 부가 설명을 확인.

**상태(State):** 소프트웨어 파일 중 하나를 확장, 축소 또는 수정하고 (바이너리 오브젝트 코드를 다루는 경우) 수정된 소프트웨어를 (재)컴파일하고 새 바이너리 파일에 (재)링크한다면, 오픈소스 소프트웨어 일부분을 수정하고 있는 것이다. 그러나 환경 설정 파일의 일부만 수정하는 경우라면 오픈소스 소프트웨어 자체를 수정하고 있는 것은 아니다.

**사용상황(Context):** 다음 사항에 해당된다면, 오픈소스 소프트웨어의 일부를 더 큰 유닛의 내장 컴포넌트로 사용하고 있는 것으로 본다.

- 더 큰 유닛의 파일 중 하나가 수정본(스니펫 코드 등)이나 원본을 포함하고 있는 경우
- 사용하는 오픈소스 소프트웨어의 기능적 정의 파일을 나타내는 include 문을 더 큰 유닛에 포함하는 경우
- 사용하는 오픈소스 소프트웨어에 정의된 함수를 더 큰 유닛이 포함하는 경우
- 개발환경이 사용하는 오픈소스 소프트웨어(또는 바이너리)를 나타내는 링커 지시(linker directive)나 컴파일러를 포함하고 있거나 더 큰 유닛이 링커 지시를 분석하지 않고 실행될 수 없는 경우

**받는 사람(Recipient):** 개인으로서 사람, 조직, 기업 등 다른 실체에 전달하지 않는다면, 또는 특정 개발 그룹의 일원으로서 자신이 속한 개발 그룹의 다른 일원에게만 전달한다면 받은 오픈소스 소프트웨어를 단독으로 사용하고 있는 것으로 본다. 그러나 휴대폰, USB 스틱 등의 장치에 오픈소스 소프트웨어를 저장하거나 이메일과 같은 전송 매체에 첨부하고 이 장치를 판매하거나 무료로 제공하거나 단순히 매체를 자신의 개발 그룹의 직계 일원이 아닌 누군가에게 전송한다면 제3자에게 오픈소스 소프트웨어를 확실히 양도하는 것으로 본다. <sup>404)</sup>

**배포형태(Form):** 오픈소스 소프트웨어는 바이너리 그리고 소스코드의 형태 두 가지로 배포하는 방법이 있다. 대부분 바이너리만을 배포 할 것인지 또는 소스를 함께 배포 할 것인지 결정하는 일은 사용자의 몫이다. 언뜻 보기에 '소스'와 '바이너리'라는 개념이 명확하게 구별되는 것처럼 보이지만 한편으로는 컴파일된 소스는 바이너리로 간주되는 것이 마땅하다. 반면 편집 가능한 소프트웨어는 '소스'의 개념으로 인식된다. 하지만 때로는 그 차이가 명확하지 않을 수 있다. 예를 들어 HEX 편집기를 사용하여 컴파일된 객체를 수정할 수도 있다. 또는 자바 스크립트 파일이 실제 텍스트 파일인 경우에도 축약된 버전일 경우 이를 수정하는 것은 매우 어렵다. 따라서 OSLIC는 유명한 경험 법칙을 '재사용'한다 : "저작물의 소스코드라는 것은 수정 작업을 하는데 일반적으로 선호되는 형식을 의미한다." <sup>405)</sup> 모든 다른 배포형태는 '바이너리'의 개념으로 표시된다. 이 스펙을 기반으로 소스 또는 바이너리를 배포하려는 경우 몇 가지 특수 조건을 중시해야 한다.

**IO액세스(ioAccess):** 오픈소스 프로그램이나 오픈소스 라이브러리를 이용하여 자신의 프로그램을 실행할 때 사용자는 일반적으로 그 소프트웨어를 배포해서는 안 된다. 이러한 상황에서 대부분의 오픈소스 라이선스는 세계적으로 널리 사용되는 인터넷 서비스의 기반이라고 할지라도 프로그램을 실행하는데 아무것도 요구하지 않는다. 이러한 간극을 좁히기 위해서 AGPL이 발표되었다. AGPL은 GPL과 마찬가지로 소프트

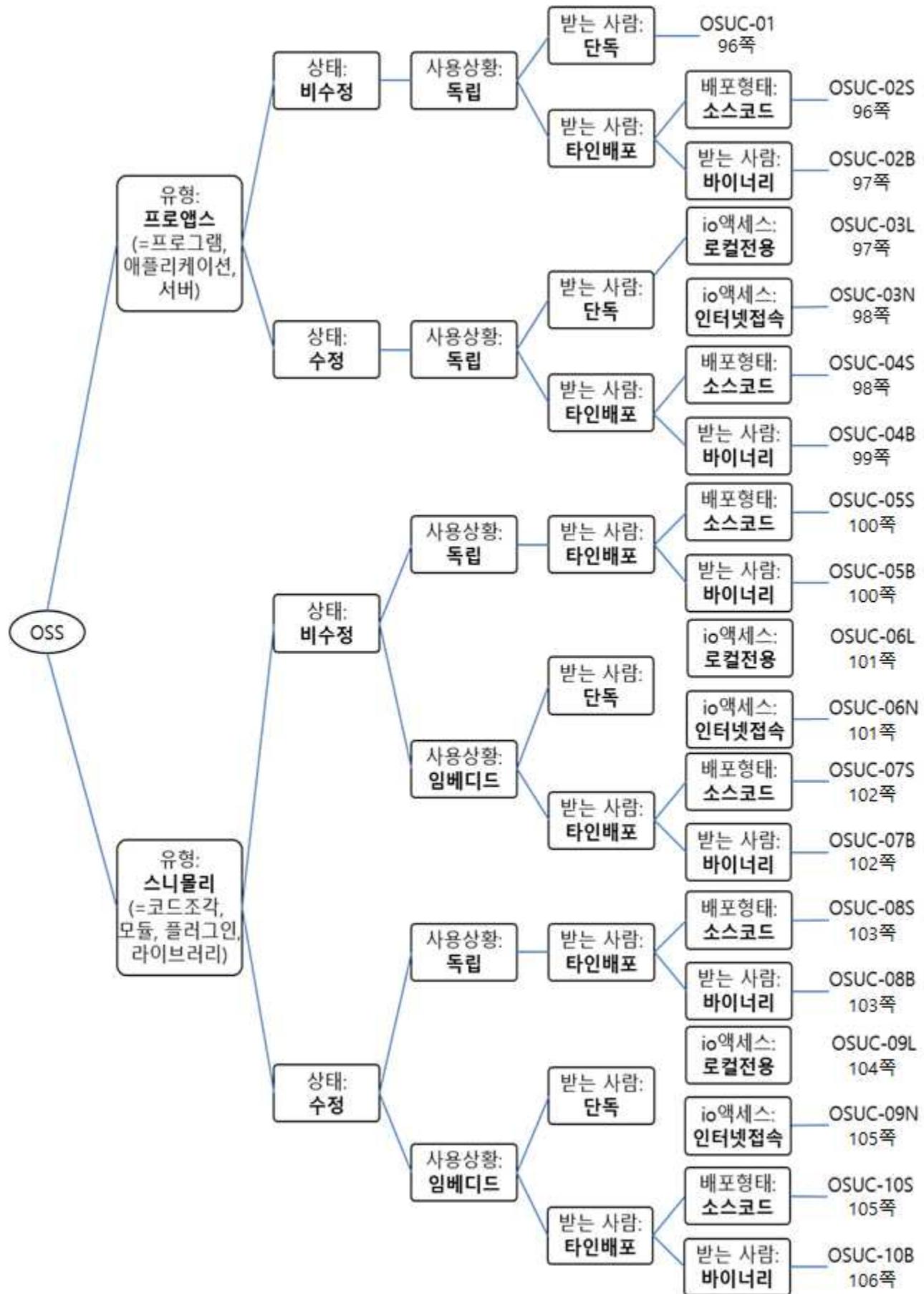
404) 적어도 독일에서는 같은 회사 내의 다른 법적 주체나 부서에 소프트웨어를 다루는 것 역시 배포의 한 형태로 보는 의견이 있음을 명심해야 한다. 배포 또는 양도를 가능한 가장 넓은 의미로 받아들이는 것이 안전하다.

405) cf. Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp §3.

웨어를 배포함으로써 촉발되는 의무사항들을 이행해야 한다. 하지만 이러한 작업들은 원격 네트워크 상호 작용에 의해서도 유발될 수 있다. 컴퓨터 네트워크를 통해 로컬에서 실행되는 프로그램과 원격으로 상호 작용하는 사용자는 누구나 일반적으로 배포판을 받는 사람이 얻을 수 있는 모든 권리를 갖는다. 그럼에도 불구하고 AGPL은 작업 오버 헤드 발생을 꺼리는 측면에서 로컬로 실행된 오픈소스 프로그램만 수정되거나 로컬로 실행된 프로그램만이 실제 의무사항을 발생시킨다. 그리하여 로컬전용 IO액세스 및 인터넷 연결 IO액세스 기능을 발표하였다. 이는 오직 단독으로 프로그램을 사용하거나 [(해당 AGPL 프로그램이 수정{프로앱스와 수정된}되었거나) 혹은 (해당 프로그램이 임베디드 GPL 라이브러리{스니몰리와 임베디드}를 사용한)] 경우 관련된 기능이다.

## 5.2 분류적 오픈소스 사용 사례 파인더

필요한 정보를 수집한 후에 다음과 같은 트리와 해당 분기를 검토하여 특정 오픈소스 사용 사례를 결정 한다.



### 5.3 오픈소스 사용 사례 및 그 과업 목록 참고자료

다음 쪽에서는 각 오픈소스 사용 사례를 한 번 더 구체화하고 페이지 번호 목록을 덧붙인다. 각 쪽에서는 설명된 오픈소스 사용 사례의 정황 하에 라이선스에 따라 처리 가능 방식을 제공하는 항목이 들어있는 라이선스 특정 과업 목록에 대해 다룰 것이다.

**OSUC-01:** 단독으로 수정하지 않고 오픈소스 프로그램, 어플리케이션 또는 서버를 받은 그대로 사용할 의도를 갖고 있다. 그러나 소프트웨어 개발이라는 취지로 다른 컴포넌트와 이를 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-02S:** 받은 그대로의 소스코드 형태로 제3자에게 수정되지 않은 오픈소스 프로그램, 어플리케이션 또는 서버를 배포할 의도를 갖고 있다. 이와 같은 배포 행위에서는 소프트웨어 개발이라는 취지로 이 프로그램, 어플리케이션 또는 서버를 다른 소프트웨어 컴포넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽

- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-02B:** 받은 그대로의 바이너리 형태로 제3자에게 수정되지 않은 오픈소스 프로그램, 어플리케이션 또는 서버를 배포할 의도를 갖고 있다. 이와 같은 배포 행위에서는 소프트웨어 개발이라는 취지로 이 프로그램, 어플리케이션 또는 서버를 다른 소프트웨어 컴포넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-03L:** 수정한 오픈소스 프로그램, 응용 프로그램 또는 서버를 실행하고 있으며 (소프트웨어 개발의 관점에서 다른 컴포넌트와 결합되어 있지는 않음) 오직 로컬을 통해서만 사용자에게 입출력을 배포하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽

- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-03N:** 수정한 오픈소스 프로그램, 응용 프로그램 또는 서버를 실행하고 있으며 (소프트웨어 개발의 관점에서 다른 컴포넌트와 결합되어 있지는 않음) 인터넷을 통해 사용자 또는 다른 사용자에게 입출력을 배포하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-04S:** 제3자에게 소스코드 형태로 배포하기 전에 받은 오픈소스 프로그램, 어플리케이션 또는 서버를 수정할 의도를 갖고 있다. 그러나 소프트웨어 개발이라는 취지로 다른 컴포넌트와 이 수정된 프로그램, 어플리케이션 또는 서버를 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴

보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-04B:** 제3자에게 바이너리 형태로 배포하기 전에 받은 오픈소스 프로그램, 어플리케이션 또는 서버를 수정할 의도를 갖고 있다. 그러나 소프트웨어 개발이라는 취지로 다른 컴포넌트와 이 수정된 프로그램, 어플리케이션 또는 서버를 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽

- **PHP-3.0** License – 243쪽

**OSUC-05S:** 받은 그대로 소스코드 형태로 제3자에게 수정되지 않은 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 배포할 의도를 갖고 있다. 이와 같은 배포 행위에서는 소프트웨어 개발이라는 취지로 이 스니펫 코드, 모듈 또는 플러그인을 다른 소프트웨어 컴포넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-05B:** 받은 그대로 바이너리 형태로 제3자에게 수정되지 않은 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 배포할 의도를 갖고 있다. 이와 같은 배포 행위에서는 소프트웨어 개발이라는 취지로 이 스니펫 코드, 모듈 또는 플러그인을 다른 소프트웨어 컴포넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽

- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-06L:** 사용자에게 오직 로컬을 통해서만 그것의 입력/출력을 배포하고 수정되지 않은 내장된 상태의 오픈소스 라이브러리, 코드 스니펫, 모듈 또는 플러그인을 사용하는 응용프로그램을 실행하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **GPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-06N:** 인터넷을 통해 사용자 또는 다른 사용자에게 그것의 입력/출력을 배포하고 수정되지 않은 내장된 오픈소스 라이브러리, 코드 스니펫, 모듈 또는 플러그인을 사용하는 응용프로그램을 실행하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽

- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-07S:** 받은 그대로 제3자에게 배포하기 전에 더 큰 소프트웨어 유닛과 함께 소스코드 형태로 소프트웨어 개발이라는 취지로 비수정 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 더 큰 소프트웨어 유닛에 결합하고 삽입할 의도를 갖고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-07B:** 받은 그대로 제3자에게 배포하기 전에 더 큰 소프트웨어 유닛과 함께 바이너리 형태로 소프트웨어 개발이라는 취지로 비수정 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 더 큰 소프트웨어 유닛에 결합하고 삽입할 의도를 갖고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽

- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-08S:** 제3자에게 배포하기 전에 소스코드 형태로 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 수정할 의도를 갖고 있다. 그러나 소프트웨어 개발이라는 취지로 이를 다른 소프트웨어 컴포넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-08B:** 제3자에게 배포하기 전에 바이너리 형태로 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 수정할 의도를 갖고 있다. 그러나 소프트웨어 개발이라는 취지로 이를 다른 소프트웨어 컴포

넌트와 결합하지 않는다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-09L:** 사용자에게 오직 로컬을 통해서만 그것의 입력/출력을 배포하고 내장된 수정된 상태의 오픈 소스 라이브러리, 코드 스니펫, 모듈 또는 플러그인을 사용하는 응용프로그램을 실행하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽

- **PHP-3.0** License – 243쪽

**OSUC-09N:** 인터넷을 통해 사용자 또는 다른 사용자에게 그것의 입력/출력을 배포하고 내장된 수정된 상태의 오픈소스 라이브러리, 코드 스니펫, 모듈 또는 플러그인을 사용하는 응용프로그램을 실행하고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-10S:** 소프트웨어 개발이라는 취지로 제3자에게 소스코드 형태로 배포하기 전에 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 수정하여 다른 소프트웨어 컴포넌트와 결합할 의도를 갖고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause** License (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause** License (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT** License (= Massachusetts Institute of Technology) - 219쪽

- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

**OSUC-10B:** 소프트웨어 개발이라는 취지로 제3자에게 바이너리 형태로 배포하기 전에 오픈소스 라이브러리, 스니펫 코드, 모듈 또는 플러그인을 수정하여 다른 소프트웨어 컴포넌트와 결합할 의도를 갖고 있다. 구체적으로 라이선스를 충족시키는 과업 목록을 살펴보려면 아래의 안내하는 쪽으로 이동한다.

- **AGPL-3.0** (= GNU Affero General Public License) - 108쪽
- **Apache-2.0** (= Apache License) - 125쪽
- **BSD 2-Clause License** (= Berkeley Software Distribution) - 139쪽
- **BSD 3-Clause License** (= Berkeley Software Distribution) - 133쪽
- **CDDL-1.0** (= Common Develop and Distribution License) - 146쪽
- **EPL-1.0** (= Eclipse Public License) - 150쪽
- **EUPL-1.1** (= European Union Public License) - 161쪽
- **GPL-2.0** (= GNU General Public License Version 2) - 172쪽
- **GPL-3.0** (= GNU General Public License Version 3) - 183쪽
- **LGPL-2.1** (= GNU Lesser General Public License Version 2.1) - 196쪽
- **LGPL-3.0** (= GNU Lesser General Public License Version 3.0) - 206쪽
- **MIT License** (= Massachusetts Institute of Technology) - 219쪽
- **MPL** (= Mozilla Public License) - 223쪽
- **MS-PL** (= Microsoft Public License) - 234쪽
- **PostgreSQL** (= Postgres License) - 240쪽
- **PHP-3.0** License – 243쪽

## 6 오픈소스 라이선스 준수 : 과업 목록

정의된 오픈소스 사용사례에 관련하여, 이 장에서는 특정 오픈소스 라이선스에 따라 해야 할 일을 열거하여 설명한다.

### 6.1 누군가에게 파일을 '제공'함에 있어서의 일반적 설명

이 장에서는 대부분의 과업 목록과 관련 있는 일반적인 항목부터 짚고 넘어간다. 동일 항목이 너무 자주 반복되지 않기 위해 여기에서는 일반적인 설명으로 시작하고 본 장을 통해서 이를 언급할 것이다.

- 오픈소스 소프트웨어를 포함한 바이너리 패키지를 배포할 때, 바이너리 패키지에서 패키지 내에 포함된 모든 파일에 엑세스를 허용하지 않을 때가 있다. 예를 들면, 상당수의 모바일 장치는 사용자가 파일 시스템에 엑세스하지 못하게 한다. 그러나 오픈소스 라이선스는 종종 라이선스 원문, 저작권 정보, 특정 고지 파일 등과 같은 텍스트 파일 사본을 타인에게 제공할 것을 요구한다. '타인에게 텍스트를 제공하라'는 의미는 수령자가 이를 반드시 읽을 수 있는 상태여야 함을 말한다.<sup>406)</sup> 따라서 파일 브라우저와 적합한 리더(reader)를 제공하는 시스템에서는 이 파일을 파일 시스템에 넣는 것으로도 충분하다. 다른 시스템에서는 애플리케이션 UI(예를 들어, 특정 저작권을 고지하는 화면 같은<sup>407)</sup>)를 통해 파일의 내용을 표시해야 한다. OSLC에서는 운영시스템 레벨까지 분류체계를 상세화하길 꺼리고 있는데, 이는 사용자가 과업 목록을 읽을 때, 이 점을 명심해야 하며, 이것은 전적으로 사용자에게 달려 있다.
- 오픈소스 소프트웨어를 사용하고 배포하는 제품은 인터넷상에서 엑세스 가능한 라이선스 파일의 일반 버전에 링크를 표시하여 "수령자에게 라이선스 등을 제공"해야 한다는 요구사항을 충족시켜야 할 때가 있다. 특히 완벽한 투명성을 위해, 프로젝트 홈페이지에 링크를 거는 전통적인 방식은 여태껏 사용하던 권장 방식이었지만, 링크만 제공하는 것으로는 충분하지 않다. 오픈소스 라이선스 의무사항에 의해 다른 사용자들에게 무언가를 양도해야 한다면, 사용자는 반드시 양도행위를 해야 한다. 제품이 배포될 때마다 다른 누군가가 항상 파일까지 제공할 것이라고 기대하면서 이 업무를 다른 누군가에게 위임하는 것은 안전하지 않다.<sup>408)</sup> 링크가 앞으로 계속 유효하다는 가정이 틀림없더라도, 다른 사람이 아닌 사용자 스스로가 라이선스 의무사항을 준수해야 한다는 것이다.

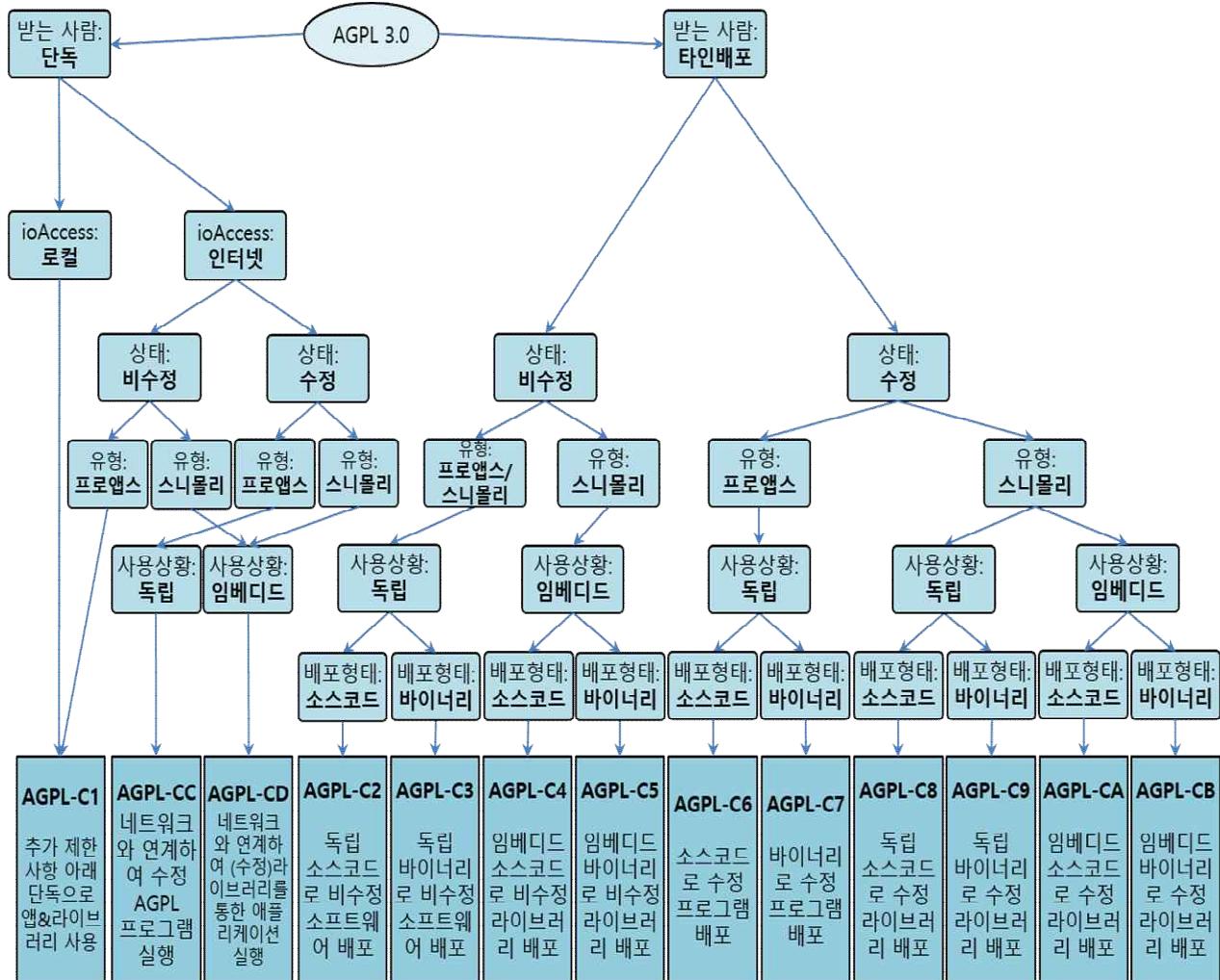
---

406) 타인에게 만지고, 느끼거나, 볼 수 없는 것을 제공하는 것은 물체를 제공하지 않는 것과 같다.

407) 또한 오픈소스 커뮤니티에서는 참조 데이터를 자발적으로 표시하는 것이 이제껏 통상적으로 사용되어 오던 전통적인 방식이었다.

408) 또한 이러한 작업의 이점은 연결되지 않은 타겟 링크의 내포된 수정물에 고심할 필요가 없다는 점이다.

## 6.2 AGPL 라이선스 소프트웨어



### 6.2.1 AGPL-3.0-C1: 추가 제한사항 아래 단독으로 소프트웨어 사용

**의미:** 사용자는 AGPL-3.0 라이선스의 소프트웨어를 획득하였으며, 본인만을 위해서 사용하고 어떤 의미에서든 제 3자에게 양도하지 않는 것을 의미한다. 또한, 컴퓨터 네트워크를 통해 원격으로 실행되는 소프트웨어와 상호작용하는 것을 보증하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-06L, OSUC-09L<sup>409)</sup>

**요구사항:** GNU Affero General Public License Version 3의 조건을 충족하기 위한 의무사항은 없다.

- 사용자가 제 3자에게 프로그램을 배포하지 않는 한, 별 다른 의무사항 없이 수정되지 않은 AGPL 프로그램을 실행 가능하다. 또한, 사용자가 제 3자에게 AGPL의 라이브러리 또는 프로그램을 배포하거나, 컴퓨터 네트워크를 통한 원격의 상호작용이 이루어지지 않는 한, 별다른 의무사항 없이 AGPL라이선스의 라이브러리, 코드일부, 또는 모듈을 사용자 프로그램에 탑재하고 실행이 가능하다.

409) 자세한 사항→ OSLiC, pp.97-107

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

**6.2.2 AGPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포**

**의미:** 사용자는 독립적인 단위로서, 수정하지 않은 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인 등 각기 다른 형태라도 배포하는 것에는 차이가 없다.

**적용범위:** OSUC-02S, OSUC-05S<sup>410)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득 한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함 해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가 해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>411)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

**6.2.3 AGPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포**

**의미:** 사용자는 독립적인 단위로서, 수정하지 않은 바이너리 패키지 또는 바이너리 파일 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인을 배포하는 것에 문제가 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>412)</sup>

410) 자세한 사항 → OSLiC, pp.97-107

411) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>413)</sup>
- [필수적:] 라이브러리를 탑재한 전체 프로그램 소스코드(라이브러리의 소스코드도 포함)를 공개적으로 제공할 수 있도록 만들어야 한다.
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지 배포를 중단한 후에도, 적어도 3년 이상은 저장소를 온라인에서 유지해야 한다.
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-C2 사용 사례의 과업 목록을 수행해야 한다.<sup>414)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

#### 6.2.4 AGPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 큰 단위에 탑재된 컴포넌트로서, 수정하지 않은 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 또는 라이브러리를 획득하였다.

**적용범위:** OSUC-07S<sup>415)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.

412) 자세한 사항 → OSLiC, pp.97-107

413) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

414) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

415) 자세한 사항 → OSLiC, pp.103

- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>416)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 온탑 개발의 카피라이트 다이얼로그에서 해당 소프트웨어는 AGPL-3.0 라이선스의 라이브러리를 사용하고, AGPL-3.0 라이선스로 선언되어 있음을 확실히 명시해야 한다. 그리고 해당 다이얼로그는 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다
- [필수적:] AGPL-3.0 라이선스 전문을 따르는 온탑 개발 소스코드를 준비해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

#### **금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### **6.2.5 AGPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포**

**의미:** 사용자는 큰 단위에 임베디드 컴포넌트로서, 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 또는 라이브러리를 획득하였다.

**적용범위:** OSUC-07B<sup>417)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>418)</sup>
- [필수적:] 라이브러리를 탑재한 전체 프로그램 소스코드(라이브러리의 소스코드도 포함)를 공개적으로 제공할 수 있도록 만들어야 한다.

416) 파일에 대한 양도를 올바르게 수행→ OSLiC, p.108

417) 자세한 사항 → OSLiC, pp.103

418) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

- 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지 배포를 중단한 후, 적어도 3년 이상은 저장소를 온라인에서 유지해야 한다.
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 온탑 개발의 카피라이트 다이얼로그에서 해당 소프트웨어는 AGPL-3.0 라이선스의 라이브러리를 사용하고, AGPL-3.0 라이선스로 선언되어 있음을 확실히 명시해야 한다. 그리고 해당 다이얼로그는 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다
- [필수적:] AGPL-3.0 라이선스 전문을 따르는 온탑 개발의 바이너리를 준비해야 한다.
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-C4 사용 사례의 과업 목록을 수행해야 한다.<sup>419)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.6 AGPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04S<sup>420)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>421)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 해당 프로그램의 카피라이트 다이얼로그에서는 해당 소프트웨어는 AGPL-3.0 라이선스의

419) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

420) 자세한 사항 → OSLiC, pp.99

421) 파일에 대한 양도를 올바르게 수행→ OSLiC, p.108

프로그램임을 확실히 명시해야 한다. 또한, 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다. 이와 같은 조건들이 포함되어 있지 않을 경우, 누락된 요소들을 추가해야 한다.

- [필수적:] 프로그램(proaspse)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 프로그램의 수정파일은 모두 제공해야 한다. 사용자가 새로운 소스코드 파일을 프로그램에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>422)</sup>
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.7 AGPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04B<sup>423)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>424)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 프로그램(proaspse)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다
- [필수적:] 해당 프로그램의 카피라이트 다이얼로그에서는 해당 소프트웨어는 AGPL-3.0 라이선스의

422) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

423) 자세한 사항→ OSLiC, pp.100

424) 파일에 대한 양도를 올바르게 수행→ OSLiC, p.108

프로그램임을 확실히 명시해야 한다. 또한, 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다. 이와 같은 조건들이 포함되어 있지 않을 경우, 누락된 요소들을 추가해야 한다.

- [필수적:] AGPL-3.0 라이선스에 적용받는 프로그램의 수정파일은 모두 제공해야 한다. 사용자가 새로운 소스코드 파일을 프로그램에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>425)</sup>
- [필수적:] 배포한 소스코드를 공개적으로 제공할 수 있도록 만들어야 한다.
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지를 배포를 중단한 후, 적어도 3년이상은 저장소를 온라인에서 유지해야 한다.
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-C6 사용 사례의 과업 목록을 수행해야 한다.<sup>426)</sup>
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항, 소프트웨어명의 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.8 AGPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snakemake)을 획득하였다. 하지만, 해당 파일들이 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>427)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.

425) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

426) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

427) 자세한 사항→ OSLiC, pp.104

- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>428)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 라이브러리(snimoli)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 라이브러리의 수정파일은 모두 제공해야 한다. 사용자가 새로운 소스코드 파일을 라이브러리에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>429)</sup>
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.2.9 AGPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>430)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>431)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 배포한 소스코드를 공개적으로 제공할 수 있도록 만들어야 한다.
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가

428) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

429) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

430) 자세한 사항 → OSLiC, pp.104

431) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

가능하도록 해야 한다. 소프트웨어 패키지를 배포를 중단한 후에도, 적어도 3년 이상은 저장소를 온라인에서 유지해야 한다.

- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-C8 사용 사례의 과업 목록을 수행해야 한다.<sup>432)</sup>
- [필수적:] 라이브러리(snimoli)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 라이브러리의 수정파일은 모두 제공해야 한다. 사용자가 새로운 소스코드 파일을 라이브러리에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>433)</sup>
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항, 소프트웨어명의 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.10 AGPL-3.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>434)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 소스코드 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>435)</sup>

432) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

433) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

434) 자세한 사항→ OSLiC, pp.106

- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 온탑 개발의 카피라이트 다이얼로그에서 해당 소프트웨어는 AGPL-3.0 라이선스의 라이브러리를 사용하고, AGPL-3.0 라이선스로 선언되어 있음을 확실히 명시해야 한다. 그리고 해당 다이얼로그는 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.
- [필수적:] 임베디드 라이브러리(snimoli)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 임베디드 라이브러리의 수정파일은 모두 제공해야 한다. 사용자가 새로운 파일을 임베디드 라이브러리에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>436)</sup>
- [필수적:] AGPL-3.0 라이선스 전문을 따르는 온탑 개발 소스코드를 준비해야 한다.
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

#### **금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### **6.2.11 AGPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포**

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, AGPL-3.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>437)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서

435) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

436) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

437) 자세한 사항 → OSLiC, pp.107

빠져 있다면 이를 추가해야 한다.<sup>438)</sup>

- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 라이브러리를 탑재한 전체 프로그램 소스코드(라이브러리의 소스코드도 포함)를 공개적으로 제공할 수 있도록 만들어야 한다
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지 배포를 중단한 후에도, 적어도 3년이상은 저장소를 온라인에서 유지해야 한다.
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-CA 사용 사례의 과업 목록을 수행해야 한다.<sup>439)</sup>
- [필수적:] 온탑 개발의 카피라이트 다이얼로그에서 해당 소프트웨어는 AGPL-3.0 라이선스의 라이브러리를 사용하고, AGPL-3.0 라이선스로 선언되어 있음을 확실히 명시해야 한다. 그리고 해당 다이얼로그는 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다
- [필수적:] 임베디드 라이브러리(snimoli)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다
- [필수적:] AGPL-3.0 라이선스에 적용받는 임베디드 라이브러리의 수정파일은 모두 제공해야 한다. 사용자가 새로운 소스코드 파일을 임베디드 라이브러리에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>440)</sup>
- [필수적:] AGPL-3.0 라이선스 전문을 따르는 온탑 개발 바이너리를 준비해야 한다.
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항, 소프트웨어명의 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.12 AGPL-3.0-CC: 수정한 프로그램을 네트워크와 연계하여 실행

**의미:** 사용자는 AGPL-3.0 라이선스의 프로그램, 애플리케이션, 또는 서버를 획득하였다. 이는 수정이 가능하고, 또한 사용자가 아닌 제 3자가 컴퓨터 네트워크를 통해 원격으로 상호작용하여 해당 프로그램, 애플리케이션, 또는 서버를 실행한다.

438) 파일에 대한 양도를 올바르게 수행 → OSLIC, p.108

439) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

440) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>442)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 프로그램(proaspse)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] 해당 프로그램의 카피라이트 다이얼로그에서는 해당 소프트웨어는 AGPL-3.0 라이선스의 프로그램임을 확실히 명시해야 한다. 또한, 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다. 이와 같은 조건들이 포함되어 있지 않을 경우, 누락된 요소들을 추가해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 프로그램의 수정파일은 모두 제공해야 한다. 사용자가 새로운 파일을 프로그램에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>443)</sup>
- [필수적:] 수정된 실행 프로그램을 공개적으로 제공할 수 있도록 만들어야 한다.
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지를 배포를 중단한 후에도, 적어도 3년이상은 저장소를 온라인에서 유지해야 한다.
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-C6 사용 사례의 과업 목록을 수행해야 한다.<sup>444)</sup>
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0 링크를 포함해야 한다.

**금지사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

441) 자세한 사항→ OSLiC, pp.99

442) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

443) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

444) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

### 6.2.13 AGPL-3.0-CD: 수정한 임베디드 라이브러리를 네트워크와 연계하여 실행

**의미:** 사용자는 AGPL-3.0 라이선스의 라이브러리, 스니펫, 또는 모듈을 획득하였다. 수정 또는 수정하지 않은 해당 라이브러리, 스니펫, 또는 모듈을 사용자의 중요한 프로그램, 애플리케이션, 또는 서버에 탑재할 목적으로 사용한다. 또한 사용자가 아닌 제 3자가 컴퓨터 네트워크를 통해 원격으로 상호작용하여 해당 프로그램, 애플리케이션, 또는 서버를 실행한다.

**적용범위:** OSUC-06N,OSUC-09N<sup>445)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 AGPL-3.0을 언급 또는 보증의 부인이 포함된 모든 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 배포된 바이너리 패키지는 눈에 잘 뛰고 찾기 쉬운 형태의 카피라이트 고지사항을 포함해야 한다. 해당 요소가 누락되었을 경우, 주요 카피라이트 고지사항을 포함하는 새 파일을 추가해야 한다.
- [필수적:] 수령자에게 AGPL-3.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>446)</sup>
- [필수적:] 현존하는 모든 카피라이트 고지사항을 유지해야 한다.
- [필수적:] 수정된 라이브러리를 탑재한 실행 프로그램의 소스코드(라이브러리의 소스코드도 포함)를 공개적으로 제공할 수 있도록 만들어야 한다.
  - 소스코드 패키지를 사용자가 제어할 수 있는 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 소프트웨어 패키지를 배포를 중단한 후에도, 적어도 3년이상은 저장소를 온라인에서 유지해야 한다.)
- [필수적:] 배포 패키지 안에 소스코드의 찾는 방법 및 위치에 대한 설명이 포함된 쉬운 상세설명서를 삽입해야 한다.
- [필수적:] 소스코드 배포를 위해 AGPL-3.0-CA 사용 사례의 과업 목록을 수행해야 한다.<sup>447)</sup>
- [필수적:] 온탑 개발의 카피라이트 디아일로그에서 해당 소프트웨어는 AGPL-3.0 라이선스의 라이브러리를 사용하고, AGPL-3.0 라이선스로 선언되어 있음을 확실히 명시해야 한다. 그리고 해당 디아일로그는 현존하는 카피라이트 고지사항, 소프트웨어명, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다.
- [필수적:] 임베디드 라이브러리(snimoli)의 소스코드 수정사항 및 수정날짜는 소스코드 내부에 모두 표시해야 한다.
- [필수적:] AGPL-3.0 라이선스에 적용받는 임베디드 라이브러리의 수정파일은 모두 제공해야 한다. 사용자가 새로운 파일을 임베디드 라이브러리에 추가했을 경우, 해당 파일에 AGPL-3.0으로 권고하는 형식의 사용자 카피라이트 라인과 라이선스 전문을 포함한 헤더를 삽입해야 한다.<sup>448)</sup>

445) 자세한 사항 → OSLiC, pp.102, 106

446) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.108

447) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

448) 자세한 사항은 AGPL-3.0 라이선스에서 “사용자의 신규 프로그램에 해당 조건 적용방법” 섹션 참조

- [필수적:] AGPL-3.0 라이선스 전문을 따르는 온탑 개발의 바이너리를 준비해야 한다.
- [자발적:] 수정 텍스트 파일을 생성하여, 기능 레벨에 대한 사용자 수정파일들의 상세설명을 해당 텍스트 파일에 삽입한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 현존하는 카피라이트 고지 사항, 소프트웨어명의 대한 힌트, 홈페이지 링크, 보증에 대한 각 면책조항, 그리고 AGPL-3.0링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.2.14 논의 및 설명

AGPL 해석의 정당화를 단순화하기 위해, AGPL-3.0과 GPL-3.0은 매우 유사하다고 할 수 있다. 다른 이름과 지나가는 발언에 발생되는 몇 가지 차이점을 제외하고<sup>449)</sup>, 두 라이선스의 대부분의 단락은 정확하게 동일한 텍스트를 제공한다.<sup>450)</sup> APGL 섹션 13만 GPL 섹션 13과 매치하지 않는다. GPL-3.0 섹션 13에서는 "GNU Affero General Public License-3.0으로 라이선스 된 저작물과의 링킹 및 통합"<sup>451)</sup>을 허용하지만, AGPL-3.0 섹션 13은 "원격 네트워크 인터액션"<sup>452)</sup>에 대해 다루고 있다. 그러므로, GPL-3.0 라이선스의 분석은 AGPL-3.0에도 유효하다.<sup>453)</sup>

우리는 이 두 라이선스의 차이점에 중심을 맞추어보자. AGPL-3.0은 GPL-3.0과의 격차를 좁히려고 한다. 모든 GNU 라이선스의 목적은 GNU 프로그램과 라이브러리에 대한 사용, 연구, 공유, 수정의 자유를 보존하는 것이다.<sup>454)</sup> 이 두 라이선스는 이러한 자유를 만들고 유지할 수 있는 사용자 의무사항에 대해 회피하지 못하게 막는 역할을 한다. 사용자가 단독으로 GNU 프로그램/라이브러리를 사용했을 경우, 라이선스에 대한 의무사항이 없다. 하지만, 제 3자가 관여하여 GNU 소프트웨어를 사용한다면, 이 제 3자는 다른 사용자들과 마찬가지로 해당 소프트웨어를 사용하기 위한 권한과 가능성을 부여 받는다.

예전에는 프로그램을 사용한다는 것은 소프트웨어를 자신의 컴퓨터에서 실행한다는 것을 의미하였다 (즉 프로그램을 바이너리 형태로 전달받거나 하는 절차가 필요했다). 이때는 소스코드 또는 라이선스 양도의 의무사항이 소프트웨어 배포 시부터 효력을 갖는 것으로 충분하였다. 그러나 클라우드 소프트웨어 시대인 요즘에는 사용자가 소프트웨어의 직접적인 전달 없이도 다른 사용자들에게 자유 소프트웨어의 이득을 취할 수 있게 한다. 이러한 경우, 사용자들은 자유 소프트웨어를 그들의 머신에서 실행하지만, 그럼에도 불구하고, 더 이상 본인만을 위해 단독으로만 사용하지는 않는다. 그래서 클라우드 서비

449) 머리말과 §0가 매우 유사하다. Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp.와 Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp를 비교.

450) §1 - 12, §14 - §17가 동일하다. Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp.와 Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp를 비교.

451) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §13 참조

452) Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp. §13 참조

453) → p.236

454) Free Software Foundation : What is free software? The Free Software Definition: 2015 [n.y.] (URL: <https://www.gnu.org/philosophy/free-sw.en.html>) - reference download: 2015-02-20, wp. 참조

스 테크놀러지 시대에서 의무사항을 준수하는 라이선스 시행의 트리거는 제 3자가 소프트웨어 사용에 관여 한다는 것을 나타내는 표준에 의해 반드시 보완되어야 한다. 그리고 이 표준은 더 이상 제 3자가 소프트웨어를 받았다고 전제를 세워서는 안 된다.

이러한 목적을 위해, AGPL-3.0에서는 다음과 같이 명시되어 있다. “실행되는 AGPL프로그램은 반드시 모든 유저에게 컴퓨터 네트워크를 통해 원격으로 상호작용할 수 있도록 제공하고, 소프트웨어를 용이하게 복사할 수 있는 어떠한 기준 또는 관례적인 수단을 통해서, 네트워크 서버로부터 무료로 이에 상응하는 소스코드에 액세스 권한을 제공하여, 해당 소스코드 버전을 받을 수 있는 기회를 제공해야 한다.”<sup>455)</sup> AGPL 소프트웨어 배포의 트리거는 AGPL 소프트웨어를 컴퓨터 네트워크를 통하여 원격으로 상호작용 할 수 있는 기능으로 확장되어 왔다.

이러한 분석의 첫 번째 결과로서, 우리는 다른 사용자에게 GPL-3-C2<sup>456)</sup>에서 GPL-3-CB<sup>457)</sup>까지의 의무 사항 목록을 준수하는 모든 상용 라이선스와 소프트웨어를 배포하는 GPL 사용 사례를 이어나갈 수 있다. 이 내용들은 GPL 챕터에서 이미 정의하였다.

두 번째 분석 결과는, 우리는 사용자 단독의 오픈소스 사용 사례에 대한 하위분류를 정해야 한다. 우리는 인터넷 인풋/아웃풋 액세스의 사용과 로컬 인풋/아웃풋 액세스의 사용을 구분해야 한다.

추가적으로, AGPL은 요구사항을 사용된 프로그램이 수정된 경우의 조건으로 제한한다. 해당 라이선스는 명시적으로 다음과 같이 말하고 있다. “사용자가 프로그램을 수정하였다면, 사용자의 수정된 버전은 다른 모든 사용자에게 컴퓨터 네트워크를 통하여 원격으로 상호작용할 수 있도록 제공하고, 이에 상응하는 소스코드 버전을 받을 수 있는 기회를 제공해야 한다.”<sup>458)</sup> 그러므로, 세 번째 분석결과에서 우리는 오픈소스 사용 사례(수령자 단독 사용)에 관련하여, 인터넷 액세스를 통한 I/O 액세스, 로컬환경을 통한 I/O 액세스, 수정된 상태, 그리고 수정하지 않은 상태로 하위분류를 정해야 한다.

마지막으로, 한사람만 프로그램을 실행할 수 있다면, 라이브러리는 직접적으로 실행될 수 없을 것이다. 이 때, 우리가 제기 할 수 있는 질문은, 수정되지 않은 AGPL 라이선스의 라이브러리 또는 모듈을 사용하는 사용자의 프로그램을 실행할 경우, 사용자는 어떻게 대처해야 하는가?

AGPL 라이선스의 섹션 13에서는 수정된 프로그램을 실행하는 경우에만 소스코드를 공개해야 한다 라고 명시하고 있다. 하지만, 우리는 AGPL의 다른 단락에 있는 내용을 주목해봐야 한다. 사용자가 AGPL 라이선스의 라이브러리, 코드일부, 또는 모듈을 사용자 프로그램에 탑재할 경우, AGPL의 카피레프트 효과에 의해 라이브러리, 코드일부 또는 모듈을 사용한 해당 프로그램은 AGPL로 라이선스가 적용되어야 한다. 그리고 마지막으로 모든 새 프로그램은 첫번째 빈 파일의 수정으로 간주되어 진다. 즉, 라이브러리와 중요 프로그램의 순수코드로 구성된 완벽한 소프트웨어 복합체를 위해, 사용자가 섹션 13을 존중한다면, 사용자는 수정하지 않은 AGPL 라이브러리를 사용한 프로그램만 실행할 수 있다.

---

455) Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp. §13 참조.

456) → OSLiC, p.185

457) → OSLiC, p.193

458) 위와 동일

이러한 분석결과를 기반으로 우리는 두 개의 새로운 AGPL 상세 오픈소스 사용 사례를 소개해야 하고, GPL 상세 오픈소스 사용 사례의 전체 세트를 이용할 수 있다.

- 다른 사용자에게 소프트웨어 배포를 야기하는 모든 GPL-3.0 사용 사례는 GPL 파인더와 GPL-3.0 의무사항 목록 챕터에서 정의됨으로 인해, AGPL-3.0 파인더와 AGPL-3.0 의무사항 목록 챕터로 이전된다.
- 수령자 단독사용과 로컬에서의 I/O 엑세스는 사용자가 소프트웨어를 본인만을 위해 사용할 경우, 의무사항이 존재하지 않는다는 예전 GPL 단독 사용 사례에 의해 적용된다. 그러나 AGPL의 문 맥에서는 이러한 사용사례에서 다음과 같은 추가적인 조건이 포함되어 있다. 사용자가 소프트웨어를 제 3자에게 배포하지 않거나, 다른 사람이 컴퓨터 네트워크를 통하여 원격으로 상호교환하지 못하는 환경의 사용자 머신에서 소프트웨어를 실행할 경우, 라이선스에 대한 의무사항이 존재하지 않는다.
- 수정되지 않은 AGPL 프로그램을 사용자가 실행했을 경우, 라이선스에 대한 의무사항이 존재하지 않는다.
- 사용자가 중요한 실행 프로그램의 임베디드 컴포넌트로서, 수정하지 않은 라이브러리를 실행할 경우, 사용자는 해당 프로그램을 AGPL로서 라이선스를 부여해야 한다. 그리고 섹션 13의 의무사항을 준수해야 한다.
- 사용자가 획득한 수정된 AGPL 프로그램을 실행할 경우, 섹션 13의 의무사항을 준수해야 한다.
- 사용자가 중요한 실행 프로그램의 임베디드 컴포넌트로서, 수정한 라이브러리를 실행할 경우, 사용자는 해당 프로그램을 AGPL로서 라이선스를 부여해야 한다. 그리고 봇 파트(특정 작업을 반복 수행하는 프로그램)과 관련하여 섹션 13의 의무사항들을 프로그램과 라이브러리에 충족시켜야 한다.

마지막으로, 논의되어야 하는 관점은 세밀성이다. AGPL-3.0은 다음과 같이 명시하고 있다. “실행되는 수정된 버전의 AGPL 프로그램은 반드시 모든 유저에게 컴퓨터 네트워크를 통해 원격으로 상호작용할 수 있도록 제공하고, 네트워크 서버로부터 무료로 이에 상응하는 소스코드에 액세스 권한 제공하여, 해당 소스코드 버전을 받을 수 있는 기회를 제공해야 한다.”<sup>459)</sup> 이러한 의무사항의 관점에서, 사용자는 상응하는 소스코드가 어떤 의미인지 알아야 한다. 프로그램의 임베디드 컴포넌트의 많은 부분은 반드시 전체 프로그램과 함께 배포된다.

다행이도, AGPL-3.0과 GPL-3.0는 다음과 같은 조항을 정의하고 있다. 오브젝트 형태의 저작물에 대한 “상응하는 소스코드는” 생성, 설치, 그리고 실행되는 저작물에 관하여 오브젝트 코드를 실행하고, 저작물 수정이 가능하며, 이 모든 활동들을 제어하기 위한 스크립트를 제공하는 모든 소스코드를 의미한다.<sup>460)</sup> 사용자가 이 라이선스 전문을 중요하게 생각한다면, 실행되는 AGPL 프로그램의 전체 소프트웨어 스택에 액세스할 수 있는 권한 또한 제공할 것이다.

그러나 AGPL은 그렇게 욕심이 많지 않다. 따라서 AGPL은 “저작물의 시스템 라이브러리, 일반적인 도구들, 일반적으로 수정하지 않은 채로 사용가능한 자유 프로그램 등”<sup>461)</sup>은 상응하는 소스코드의 범위에 포함

459) Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp. §13 참조.

460) Open Source Initiative: The AGPL-3.0 License (OSI), 2007, wp. §1 참조.

시키지 않는다. 위 의무사항의 이해를 돋기 위해, 사용자는 시스템 라이브러리에 대한 의미를 알아야 한다. AGPL에서는 실행가능 한 저작물의 시스템 라이브러리는 저작물 전체 이외의 모든 것이 포함된다. 상세조건은 다음과 같다.

- (A) 주요 컴포넌트 패킹 하는 일반적인 형태에 포함되나, 주요 컴포넌트의 일부는 아니다.
- (B) 주요 컴포넌트와 함께 저작물을 활성화하는 용도, 또는 대중이 사용가능한 소스코드 형태의 구현을 위해 표준 인터페이스를 구현하는 용도로 사용된다.<sup>462)</sup>

불행히도, 사용자는 현재 AGPL이 정의하는 주요 컴포넌트의 의미를 분석해야 한다. AGPL의 문맥에서 주요 컴포넌트는 특정 운영체제의 주요 필수 컴포넌트(커널, 윈도우 시스템 등)로서, 이 운영체제에서는 실행 가능한 작업 실행, 컴파일러의 저작물 생성 또는, 저작물을 실행하는 오브젝트 코드 인터프리터가 동작할 수 있다.<sup>463)</sup>

이러한 상세사항들을 기반으로 사용자는 실행되는 AGPL 프로그램의 상응하는 소스코드 엑세스 권한을 제공하는 레벨에 대해 몇가지 경험의 법칙을 제시할 수 있다.

- 사용자가 수정된 AGPL의 바이너리 프로그램을 실행할 경우, 다음과 같은 코드의 엑세스 권한을 제공해야 한다.
  - 실행되는 프로그램
    - 해당 프로그램의 모든 수정된 임베디드 컴포넌트
    - 해당 프로그램의 자유롭게 엑세스 할 수 없는 모든 임베디드 컴포넌트
    - 자유롭게 접근가능한 컴파일 또는 개발환경의 프로그램 소스코드를 컴파일하는데 필요한 모든 도구, 스크립트, 데이터.(이들은 원칙적으로 엑세스 권한을 제공하지 않아도 포함됨)
- 하지만, 사용자의 표준 저장소로부터 언제든지 다운로드가 가능한, 수정되지 않은 표준 라이브러리, 컴파일러, 또는 도구에 대한 엑세스를 제공할 필요는 없다.
- 사용자가 수정한 AGPL 라이선스 스크립트를 실행할 경우, 다음과 같은 코드의 엑세스 권한을 제공해야 한다.
  - 실행되는 스크립트
    - 메인 스크립트에 의해 포함되는 모든 수정된 임베디드 스크립트 컴포넌트
    - 메인 스크립트에 의해 포함되는, 자유롭게 엑세스 할 수 없는 모든 임베디드 스크립트 컴포넌트
    - 자유롭게 엑세스 가능한 인터프리터에 의해 실행되는 메인 스크립트를 실행하는데 필요한 필요한 모든 도구, 스크립트, 데이터.(이들은 원칙적으로 엑세스 권한을 제공하지 않아도 포함됨)
    - 접근권한이 없는 인터프리터
- 하지만, 사용자의 표준 저장소로부터 언제든지 다운로드가 가능한, 수정되지 않은 표준 스크립트 라이브러리, 인터프리터, 또는 도구에 대한 엑세스를 제공할 필요는 없다.

---

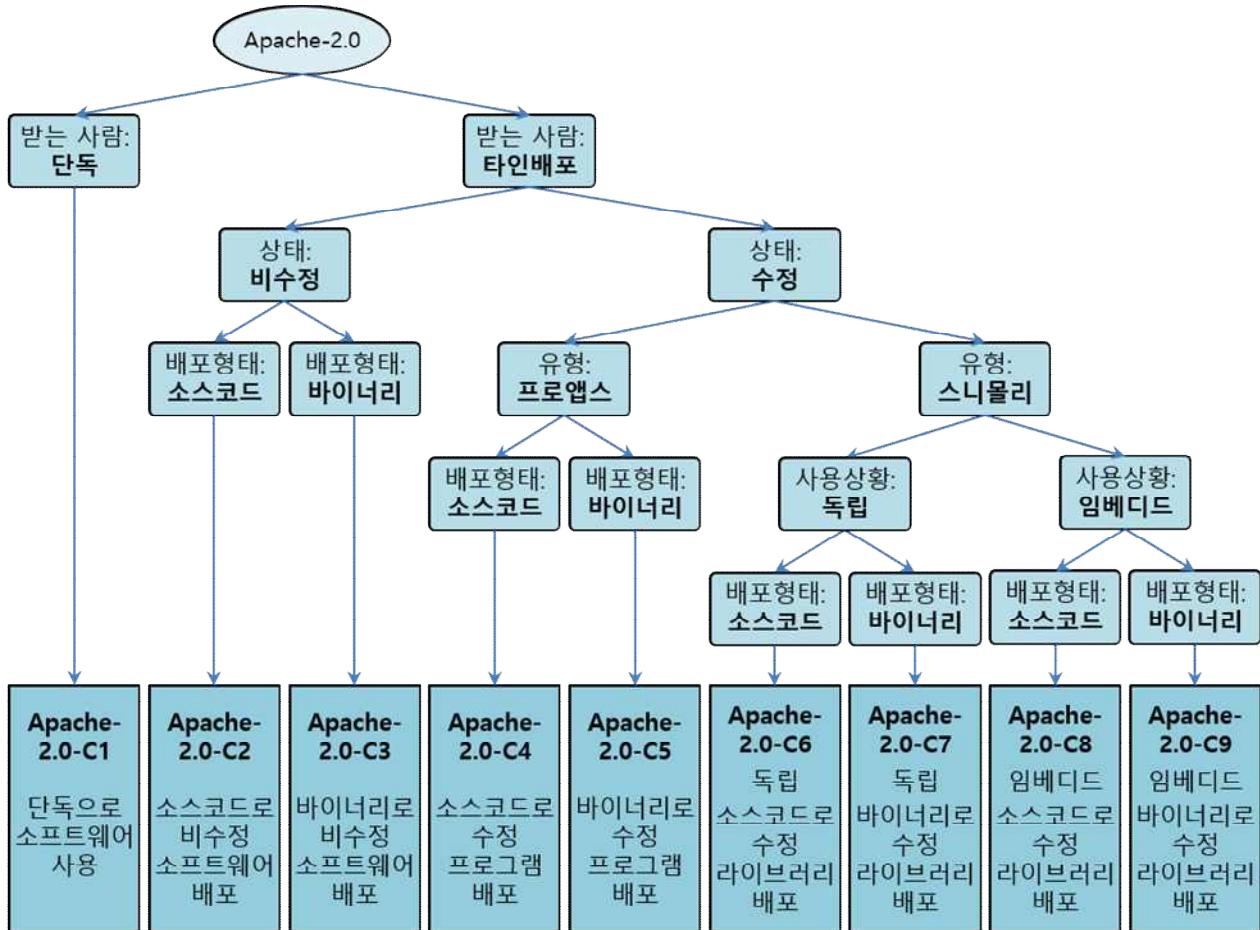
461) 위와 동일

462) 위와 동일

463) 위와 동일

### 6.3 Apache-2.0 라이선스 소프트웨어

오늘날, 아파치 오픈소스 라이선스의 현재 릴리즈 버전은 2.0이며, 구버전은 더 이상 사용되지 않는다.<sup>464)</sup> 이러한 이유는 아파치 라이선스가 “재배포”에 주로 초점을 맞추고 있기 때문이다.<sup>465)</sup> 아래의 간략하게 나타낸 아파치 관련 오픈소스 사용 사례 파인더<sup>466)</sup>를 사용할 수 있다.



#### 6.3.1 Apache-2.0-C1: 단독으로 소프트웨어 사용

**의미:** 사용자는 제 3자에게 양도하지 않고, 단독으로 사용할 목적으로, Apache-2.0 라이선스 소프트웨어를 획득하였다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>467)</sup>

**요구사항:** 다음 사용사례와 관련하여 Apache 라이선스 2.0의 조건을 충족하기 위한 의무사항은 없다.

- 사용자가 제 3자에게 소프트웨어를 배포하지 않는 한, 별 다른 의무사항 없이 어떠한 사용 상황에서

464) 자세한 사항 → OSLiC, pp.28

465) Open Source Initiative: APL-2.0, 2004, wp. §4 참조

466) 일반적인 OSUC 파인더의 자세한 사항 → OSLiC, pp.88

467) 자세한 사항 → OSLiC, pp.96-106

도 아파치 소프트웨어를 사용할 수 있다.

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.2 Apache-2.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포

**의미:** Apache 2.0 라이선스의 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>468)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>469)</sup>
- [필수적:] 라이선싱하는 요소(특히 원 저작권자의 특정 카피라이트 고지문)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 고지사항 텍스트 파일은 최초에 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.<sup>470)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.3 Apache-2.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포

**의미:** 사용자는 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드

468) 자세한 사항 → OSLiC, pp.96-106

469) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

470) 아파치 라이선스는 의도적으로 다소 모호한 것으로 보인다. “고지사항 텍스트 파일”이라는 용어를 사용한다. 제한된 의미에서 이 용어는 “NOTICE.[txt|pdf|. . .]”를 말하고 있다. 보다 약한 의미에서는 라이선싱 고지사항을 포함하는 모든 (텍스트) 파일을 의미할 수 있다. “NOTICE”라는 텍스트 파일 명칭이 없을 경우, 넓은 의미로서 이 요구사항에 따라, 사용자는 이 용어를 이해하고 해당 의무사항을 이행해야 한다.

단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드 일부, 모듈, 라이브러리 또는 플러그인을 배포하는 것에 문제가 없다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>471)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 바이너리 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>472)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대로 사용자의 패키지에 보유 또는 포함해야 한다.
- [필수적:] 제 3자의 고지사항이 일반적으로 표시되는 경우, 이러한 고지사항 텍스트 파일을 포함해야 한다. (특히, 사용자 소유의 카피라이트 고지사항을 나타내는 사용자 저작물의 임베디드 컴포넌트로서, 수정하지 않은 Apache-2.0 라이선스의 라이브러리를 배포하는 경우)
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다. 특히 사용자 소유의 카피라이트 고지사항 하위섹션으로서 포함한다.

**금지사항:**

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.4 Apache-2.0-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04S<sup>473)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>474)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.

471) 자세한 사항 → OSLiC, pp.96-106

472) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

473) 자세한 사항 → OSLiC, pp.98

474) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대로 모든 정보를 사용자의 패키지에 포함해야 한다.
- [필수적:] 제 3자의 고지사항이 일반적으로 표시되는 경우, 이러한 고지사항 텍스트 파일을 포함해야 한다. 프로그램에 이미 카피라이트 다이얼로그가 있을 경우, 적절한 방식으로 이를 업데이트한다.
- [필수적:] 소스코드 내에 사용자의 모든 수정사항을 완전히 표시한다. 고지사항 텍스트 파일이 없다면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다. 특히 사용자 소유의 카피라이트 고지사항 하위섹션으로서 포함한다.

#### **금지사항:**

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### **6.3.5 Apache-2.0-C5: 수정한 프로그램을 바이너리 형태로 배포**

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04B<sup>475)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 바이너리 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>476)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대로의 모든 정보를 사용자의 패키지에 포함해야 한다. 고지사항 텍스트 파일이 없다면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가한다.
- [필수적:] 제 3자의 고지사항이 일반적으로 표시되는 경우, 이러한 고지사항 텍스트 파일을 포함해야 한다. 프로그램에 이미 카피라이트 다이얼로그가 있을 경우, 적절한 방식으로 이를 업데이트한다.
- [자발적:] 사용자가 소스코드 내 본인의 수정사항을 공개하고 싶지 않아도, 수정사항은 모두 표시해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다. 특히 사용자 소유의 카피라이트 고지사항 하위섹션으로서 포함한다.

---

475) 자세한 사항 → OSLiC, pp.99

476) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.6 Apache-2.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 해당 파일들이 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>477)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>478)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대의 모든 정보를 사용자의 패키지에 포함해야 한다.
- [필수적:] 소스코드 내에 사용자의 모든 수정사항을 완전히 표시한다. 고지사항 텍스트 파일이 없으면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다.

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.7 Apache-2.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 해당 파일들이 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>479)</sup>

477) 자세한 사항 → OSLiC, pp.103

478) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

479) 자세한 사항 → OSLiC, pp.103

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 바이너리 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>480)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대의 모든 정보를 사용자의 패키지에 포함해야 한다. 고지사항 텍스트 파일이 없다면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가한다.
- [자발적:] 사용자가 소스코드 내 본인의 수정사항을 공개하고 싶지 않아도, 수정사항은 모두 표시해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다. 특히 사용자 소유의 카피라이트 고지사항 하위섹션으로서 포함한다.

**금지사항:**

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.3.8 Apache-2.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snakemake)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>481)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>482)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대의 모든 정보를 사용자의 패키지에 포함해야 한다.
- [필수적:] 제 3자의 고지사항이 일반적으로 표시되는 경우, 이러한 고지사항 텍스트 파일을 포함해야 한다. Apache 2.0 라이선스의 라이브러리를 포함한 소프트웨어에 카피라이트 다이얼로그가 있을

480) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

481) 자세한 사항 → OSLiC, pp.105

482) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

경우, 해당 정보를 추가해야 한다.

- [필수적:] 라이브러리의 소스코드 내에 사용자의 모든 수정사항을 완전히 표시한다. 고지사항 텍스트 파일이 없다면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가한다.<sup>483)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다.
- [자발적:] 포함된 Apache 라이선스와 고지사항 텍스트 파일이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 소스코드 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 디렉토리에 저장하는 것을 권장한다.

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.9 Apache-2.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, Apache-2.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snakemake)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>484)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 수령자에게 Apache-2.0 라이선스의 사본을 제공해야 한다. 바이너리 패키지의 일부에서 빠져 있다면 이를 추가해야 한다.<sup>485)</sup>
- [필수적:] 라이선스 요소(특히 원 저작권자의 특정 카피라이트 고지사항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 고지사항 텍스트 파일은 획득한 형태 그대로 모든 정보를 사용자의 패키지에 포함해야 한다. 고지사항 텍스트 파일이 없다면, 이를 생성하고, 사용자의 수정사항에 대한 상세설명을 추가 한다.
- [필수적:] 제 3자의 고지사항이 일반적으로 표시되는 경우, 이러한 고지사항 텍스트 파일을 포함해야 한다. Apache 2.0 라이선스의 라이브러리를 포함한 소프트웨어에 카피라이트 다이얼로그가 있을 경우, 해당 정보를 추가해야 한다.
- [필수적:] 사용자가 소스코드 내 본인의 수정사항을 공개하고 싶지 않아도, 임베디드 라이브러리의

483) 라이브러리라는 용어는 코드일부, 모듈, 그리고 플러그인도 포함한다.

484) 자세한 사항 → OSLiC, pp.106

485) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

수정사항은 모두 표시해야 한다.<sup>486)</sup>

- [필수적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 고지사항 내용, 소프트웨어 명에 대한 힌트, 홈페이지 링크, Apache-2.0 라이선스 링크를 포함해야 한다. 특히 사용자 소유의 카피라이트 고지사항 하위섹션으로서 포함한다.
- [자발적:] 포함된 Apache 라이선스와 고지사항 텍스트 파일이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 바이너리 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인을 모든 라이선싱 요소들을 포함하는 디렉토리에 저장하는 것을 권장한다.

#### 금지사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.3.10 논의 및 설명

- Apache-2.0 라이선스는 “고지사항 파일 내용의 복제와 원 저작물을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고 라이선서의 트레이드명, 트레이드 마크, 서비스 마크, 또는 제품 명의 사용을 허용하지 않는다.”<sup>487)</sup> 반면에 이 라이선스는 특히 소송을 제기한 자에게 허가된 모든 특히 라이선스를 자동으로 해지한다.<sup>488)</sup> 이러한 이유로, OSLiC에서는(Apache-2.0-C1 - Apache-2.0-C9) 일반적으로 이러한 요소에 의해 제품 또는 서비스를 홍보 및 Apache-2.0을 포함한 소프트웨어와 연결된 특히 관련 법적 투쟁을 금지하고 있다.
- Apache-2.0은 또한 저작물 또는 2차 저작물의 모든 수령자에게 이 라이선스의 사본을 제공할 것을 요구하고 있다.<sup>489)</sup> 그러므로, 타인에게 배포하는 모든 사용사례는 각각의 의무적인 조항을 포함한다. (Apache-2.0-C2 - Apache-2.0-C9).
- 추가적으로, Apache-2.0은 수정사항은 반드시 표시하도록 요구한다.<sup>490)</sup> 그러므로, 소스코드 형태로 수정된 소프트웨어를 배포하는 모든 사례에서 모든 수정사항을 표시하고 고지사항 파일 안에 힌트를 포함해야 한다. 이와 반면에 바이너리 형태로 수정된 소프트웨어를 배포하는 모든 사례에서는(Apache-2.0-C4- Apache-2.0-C9) 임의적인 조항만 삽입한다.
- 또한, Apache-2.0은 사용자가 배포하는 소스코드 형태의 모든 저작물에 모든 카피라이트, 특히, 트레이드마크, 그리고 소스코드 형태의 원저작물의 귀속 고지사항(attribution notice)를 포함해야 한다. 따라서 OSLiC에서는 모든 문맥 (Apache-2.0-C1 - Apache-2.0-C9)에서 라이선싱 요소들은 획득한 형태 그대로 보유할 것을 요구한다.<sup>491)</sup>

486) 라이브러리라는 용어는 코드일부, 모듈, 그리고 플러그인도 포함한다.

487) Open Source Initiative: APL-2.0, 2004, wp. §6 참조

488) 위와 동일, .c., wp. §3. 참조

489) Open Source Initiative: APL-2.0, 2004, wp. §4.1 참조

490) 위와 동일, wp. §4.2 참조

491) 이는 일부 독자들에게 혼란을 야기할지 모른다. 사용자가 바이너리 형태의 수정된 버전을 배포하는 경우에도, 반드시 이 의무사항을 준수해야 한다. 또한 사용자는 반드시 수령자에게 라이선스를 양도해야 한다. 하지만 수정된 소스코드를 배포할 필요는 없다. (→ OSLiC, p.28)

- 마지막으로 Apache-2.0은 획득한 고지사항 텍스트 파일을 반드시 읽을 수 있는 형태의 사본으로 소스코드 형태로 배포될 각 패키지에 포함해야 한다. 또는 바이너리 배포인 경우, 반드시 제 3자의 고지사항이 일반적으로 표시되는 경우 고지사항 텍스트 파일을 반드시 포함해야 한다.<sup>492)</sup> 이에 따라, OSLiC는 의무적으로 소스코드 배포시, 반드시 고지사항 텍스트 파일을 포함(Apache-2.0-C2, Apache-2.0-C4, Apache-2.0-C6, Apache- 2.0-C8)하고, 일반적으로 카피라이트 화면을 보여주는 바이너리 애플리케이션의 모든 배포시, 고지사항 파일 내용 (Apache-2.0-C5, Apache-2.0-C9)을 해당 스크린에 포함해야 할 것을 요구한다. 바이너리 형태의 배포된 라이브러리일 경우, 일반적으로 이와 같은 카피라이트 다이얼로그를 포함하지 않는다. (Apache-2.0-C7)

#### 6.4 BSD 라이선스 소프트웨어

승인된 오픈소스로서, BSD 라이선스는 두 가지 버전이 존재한다.<sup>493)</sup> 최신 버전은 BSD-2 Clause 라이선스,<sup>494)</sup> 구 버전은 BSD-3 Clause 라이선스이다.<sup>495)</sup> 이 두 버전은 약간의 차이가 있지만, 이 차이점은 매우 엄격히 준수해야 한다.

모든 BSD 오픈소스 라이선스는 타인에게 배포라고 그동안 지정한 재배포 오픈소스 사용 사례에 집중하고 있다. 다른 사용 사례의 조건들은 단독으로 사용하는 사례에서부터 파생될 수 있다.<sup>496)</sup> 추가적으로 BSD 라이선스는 저작물이 소스코드 파일(세트) 또는 바이너리 파일(세트)로 배포 여부에 따라, 배포 형태를 구분한다. 다음 트리를 통해서 BSD 라이선스의 과업 목록을 알아본다.

---

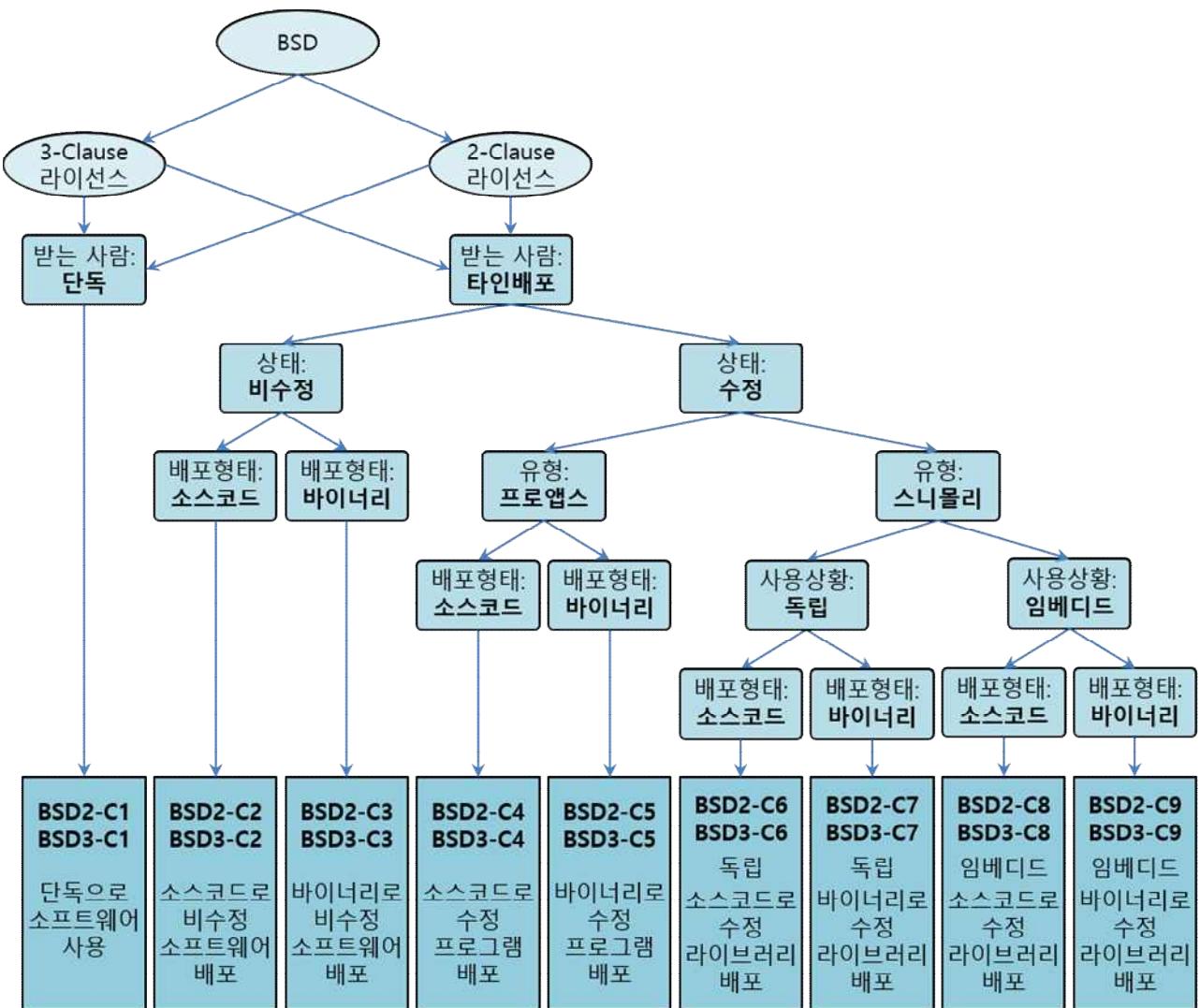
492) Open Source Initiative: APL-2.0, 2004, wp. §4.4 참조

493) OSI에 따르면, 또 다른 BSD 라이선스가 존재한다. 이 라이선스의 광고조항으로 알려진 4번째 조항은 캘리포니아 대학교의 Director of the Office of Technology Licensing 에 의해 1999년 7월 22일에 공식적으로 폐지되었다. 이러한 이유로 인해, 사용자는 가장 오래된 BSD 라이선스의 의무사항을 준수해야 한다면, Open Source Initiative의 BSD-3 Clause 라이선스를 기반으로 의무사항을 준수하면 된다. cf. Open Source Initiative: The BSD 3-Clause License, 2012, wp.

494) Open Source Initiative: The BSD 2-Clause License, 2012, wp. 참조

495) Open Source Initiative: The BSD 3-Clause License, 2012, wp. 참조

496) 오픈소스 사용 사례 토큰에 대한 자세한 사항은 88 페이지를 참조하기 바란다.



#### 6.4.1 BSD-3-Clause-C1: 단독으로 소프트웨어 사용

**의미:** 사용자는 BSD 라이선스의 소프트웨어를 획득하였으며, 본인만을 위해서 사용하고 어떤 의미에서든 제 3자에게 양도하지 않는 것을 의미한다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>497)</sup>

**요구사항:** New BSD(3 Clauses)의 조건을 충족하기 위한 의무사항은 없다.

- 사용자가 제 3자에게 소프트웨어를 배포하지 않는 한, 별 다른 의무사항 없이 어떠한 상황에서도 BSD 소프트웨어를 사용할 수 있다.

497) 자세한 사항 → OSLiC, pp.96-106

#### **금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.<sup>498)</sup>

#### **6.4.2 BSD-3-Clause-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포**

**의미:** BSD 라이선스의 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>499)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

#### **금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직명 사용.

#### **6.4.3 BSD-3-Clause-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포**

**의미:** 사용자는 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드 단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인 등 각기 다른 형태라도 배포하는 것에는 차이가 없다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>500)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>501)</sup>
- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용,

498) 예를 들면, 사용자 소유의 데이터 센터에서 사용된 BSD 소프트웨어를 기반으로 한 인터넷 서비스

499) 자세한 사항 → OSLiC, pp.96-102

500) 자세한 사항 → OSLiC, pp.97-102

501) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

**금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### 6.4.4 BSD-3-Clause-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 프로그램, 애플리케이션, 또는 서버(proapce)를 획득하였다.

**적용범위:** OSUC-04S<sup>502)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하기 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.

**금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### 6.4.5 BSD-3-Clause-C5: 수정한 프로그램을 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 프로그램, 애플리케이션, 또는 서버(proapce)를 획득하였다.

**적용범위:** OSUC-04B<sup>503)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>504)</sup>

---

502) 자세한 사항 → OSLiC, pp.98

503) 자세한 사항 → OSLiC, pp.99

- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하기 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.

#### **금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### **6.4.6 BSD-3-Clause-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포**

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>505)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

#### **금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### **6.4.7 BSD-3-Clause-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포**

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>506)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및

---

504) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

505) 자세한 사항 → OSLiC, pp.103

506) 자세한 사항 → OSLiC, pp.103

BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>507)</sup>

- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

**금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### 6.4.8 BSD-3-Clause-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>508)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하길 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.
- [자발적:] 라이선싱 요소들(특히, BSD 라이선스 전문, 원 저작자의 카피라이트 고지사항, 그리고 BSD 면책조항)이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 소스코드 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.

**금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

507) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

508) 자세한 사항 → OSLiC, pp.105

#### 6.4.9 BSD-3-Clause-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>509)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>510)</sup>
- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하길 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.
- [자발적:] 라이선싱 요소들(특히, BSD 라이선스 전문, 원 저작자의 카피라이트 고지사항, 그리고 BSD 면책조항)이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 바이너리 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.

**금지사항:**

- 사용자 소유의 저작물의 홍보를 위한 라이선싱 기여자명 또는 라이선싱 조직 명 사용.

#### 6.4.10 BSD-2-Clause-C1: 단독으로 소프트웨어 사용

**의미:** 사용자는 BSD 라이선스의 소프트웨어를 획득하였으며, 본인만을 위해서 사용하고 어떤 의미에서든 제 3자에게 양도하지 않는 것을 의미한다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>511)</sup>

**요구사항:** Simplified BSD(2 Clauses)의 조건을 충족하기 위한 의무사항은 없다.

509) 자세한 사항 → OSLiC, pp.106

510) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

511) 자세한 사항 → OSLiC, pp.96-106

- 사용자가 제 3자에게 소프트웨어를 배포하지 않는 한, 어떠한 상황에서도 어떠한 의무사항 없이 BSD 소프트웨어를 사용할 수 있다.

**금지사항:**

- 명시적인 금지사항 없음.

#### 6.4.11 BSD-2-Clause-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포

**의미:** 사용자는 수정하지 않은 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드 단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인 등 각기 다른 형태라도 배포하는 것에는 차이가 없다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>512)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

**금지사항:**

- 명시적인 금지사항 없음.

#### 6.4.12 BSD-2-Clause-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포

**의미:** 사용자는 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드 단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인 등 각기 다른 형태라도 배포하는 것에는 차이가 없다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>513)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않

---

512) 자세한 사항 → OSLiC, pp.96-102

513) 자세한 사항 → OSLiC, pp.97-102

을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>514)</sup>

- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

#### 금지사항:

- 명시적인 금지사항 없음.

### 6.4.13 BSD-2-Clause-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04S<sup>515)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하기 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.

#### 금지사항:

- 명시적인 금지사항 없음.

### 6.4.14 BSD-2-Clause-C5: 수정한 프로그램을 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04B<sup>516)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및

514) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

515) 자세한 사항 → OSLiC, pp.98

516) 자세한 사항 → OSLiC, pp.99

BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>517)</sup>

- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하기 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.

#### 금지사항:

- 명시적인 금지사항 없음.

### 6.4.15 BSD-2-Clause-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>518)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

#### 금지사항:

- 명시적인 금지사항 없음.

### 6.4.16 BSD-2-Clause-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>519)</sup>

517) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

518) 자세한 사항 → OSLiC, pp.103

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.<sup>520)</sup>
- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.

**금지사항:**

- 명시적인 금지사항 없음.

#### 6.4.17 BSD-2-Clause-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>521)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [자발적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하길 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없다면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.
- [자발적:] 라이선싱 요소들(특히, BSD 라이선스 전문, 원 저작자의 카피라이트 고지사항, 그리고 BSD 면책조항)이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 소스코드 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.

**금지사항:**

- 명시적인 금지사항 없음.

519) 자세한 사항 → OSLiC, pp.105

520) 파일에 대한 양도를 올바르게 수행 → OSLiC, pp.107

521) 자세한 사항 → OSLiC, pp.105

#### 6.4.18 BSD-2-Clause-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, BSD 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>522)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다. <sup>523)</sup>
- [필수적:] 사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.
- [자발적:] 오픈소스 커뮤니티에서는 해당 프로그램이 BSD에 의해 라이선스가 선언되었다는 내용을 실행프로그램에서 표시하기 권장한다. 왜냐하면, 표시된 원 저작권 고지사항에 이러한 전문이 없으면, 프로그램을 수정하면서 이러한 힌트를 추가할 수 있기 때문이다.
- [자발적:] 라이선싱 요소들(특히, BSD 라이선스 전문, 원 저작자의 카피라이트 고지사항, 그리고 BSD 면책조항)이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 바이너리 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.

**금지사항:**

- 명시적인 금지사항 없음.

#### 6.4.19 논의 및 설명

BSD 2-Clause 라이선스는 단순한 구조로 구성되어 있다. 먼저, BSD 2-Clasue 라이선스는 사용자가 이 라이선스의 두 가지 규칙만 준수한다면, 수정 여부에 상관없이 재배포와 소스코드 및 바이너리 형태의 사용을 허용한다.<sup>524)</sup> 첫 번째 규칙은 소스코드 형태의 배포/재배포와 관련이 있고, 두 번째로, 바이너리 패키지의 배포/재배포와 관련이 있다. 우리는 이러한 규칙을 다양한 실행 가능한 작업세트로 변환하는 이유에 대해 다음과 같이 설명한다.

- 소스코드의 재배포의 경우, 라이선스에서는 “패키지는 반드시 카피라이트 고지사항, 면책조항 및

522) 자세한 사항 → OSLiC, pp.106

523) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

524) Open Source Initiative: The BSD 2-Clause License, 2012, wp. 참조

의무사항 목록을 포함해야 한다<sup>525)</sup>고 요구하고 있다. 그러므로 사용자는 소스코드 파일에 이미 탑재된 카피라이트 고지사항을 수정할 수 없다. 그리고 논리적인 관점에서 해당 소프트웨어는 BSD-2 Clause 라이선스로 선언되어 있다는 명시적 또는 암묵적인 주장이 존재한다.<sup>526)</sup> 이것은 라이선스 사본을 패키지에 추가하여 이행할 수 있다. 따라서 사용자는 해당 파일 또는 이에 상응하는 텍스트 일부를 수정할 수 없다. 이를 위해 여기서는 금지사항을 실행 가능한 작업으로 전환하였다.

*라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트 고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.*

- 바이너리 파일 형태로 재배포하는 경우, 라이선스는 라이선싱 요소는 반드시 배포물과 함께 제공되는 문서 및 기타 자료에 복제해야 한다.<sup>527)</sup> 따라서 이는 소스코드 패키지로 재배포하기 위한 조건으로 필요하지 않다. 그러나 소스코드 형태로 배포하는 경우라도, 이 의무사항을 역시 이행 할 수 있다.(예를 들어, 소스코드 패키지를 위한 다운로드 사이트를 제공하는 경우). 이러한 경우 라이선싱 언급이 패키지 내부뿐만 아니라, 사이트 텍스트에서도 중요하다는 의미가 된다. 이로 인해, 여기에서는 소스코드 형태로 재배포하는 것을 다루는 BSD 오픈소스 사용 사례에 대한 다음의 임의적인 작업을 추가하였다.

*사용자의 배포 또는 사용자의 추가 자료의 문서 또한, 원 카피라이트 고지사항 내용, BSD 조항, 그리고 BSD 면책조항을 포함해야 한다.*

- 자연스럽게 배포물과 함께 제공되는 문서 또는 기타 자료에서의 라이선싱 요소 복제는 바이너리 형태로 재배포시 명백히 필요하므로,<sup>528)</sup> 우리는 바이너리 형태로 재배포하는 내용을 다루는 BSD 오픈소스 사용 사례의 의무적인 작업으로서, 소스코드 형태로 배포하는 조건적 작업에 대해 다시 써야 했다.
- 바이너리 형태로 프로그램을 배포/재배포하는 경우, 라이선싱 요소를 획득한 형태 그대로 배포하는 것이 때로는 충분하지 않다. 소스코드에서 바이너리 패키지를 컴파일 한다면, 라이선싱 요소 또한 자동적으로 생성된 후 반드시 바이너리 패키지로 삽입되는 것은 아니다. 그래서, BSD 라이선스에 따라, 사용자가 카피라이트 고지사항, 의무사항 및 면책조항을 해당 패키지에 추가해야 한다. 따라서 여기서는 바이너리 배포 의무사항을 준수하기 위한 라이선스, 실행파일의 다음과 같은 형태를 선택하였다.

*라이선스 요소(특히 BSD 라이선스 전문, 원 저작권자의 특정 카피라이트*

---

525) 위와 동일

526) BSD 라이선스에서는 재배포된 소프트웨어 패키지는 반드시 (패키지 별) 카피라이트 고지사항, (라이선스 별) 라이선스 의무사항 및 BSD 면책조항을 포함해야 한다. cf. id., l.c., wp 사용자의 패키지에 이러한 요소들이 누락되어 있다면, 어떻게 대처해야 하는지 문의를 받을 수 있다. 만약 그렇다면, 획득한 그대로의 패키지에는 적절한 라이선스 선언이 부여되지 않았다. 그러므로 사용자가 획득한 소프트웨어가 BSD 라이선스의 소프트웨어인지 확실히 알 수 없다. 다시 말하면, 사용자가 BSD 라이선스의 소프트웨어 패키지를 획득하였다면, 반드시 충분한 라이선스 의무사항 요소들이 포함하거나, BSD 라이선스의 소프트웨어가 아니라고 선언할 수 있다.

527) 위와 동일

528) Open Source Initiative: The BSD 2-Clause License, 2012, wp. 참조

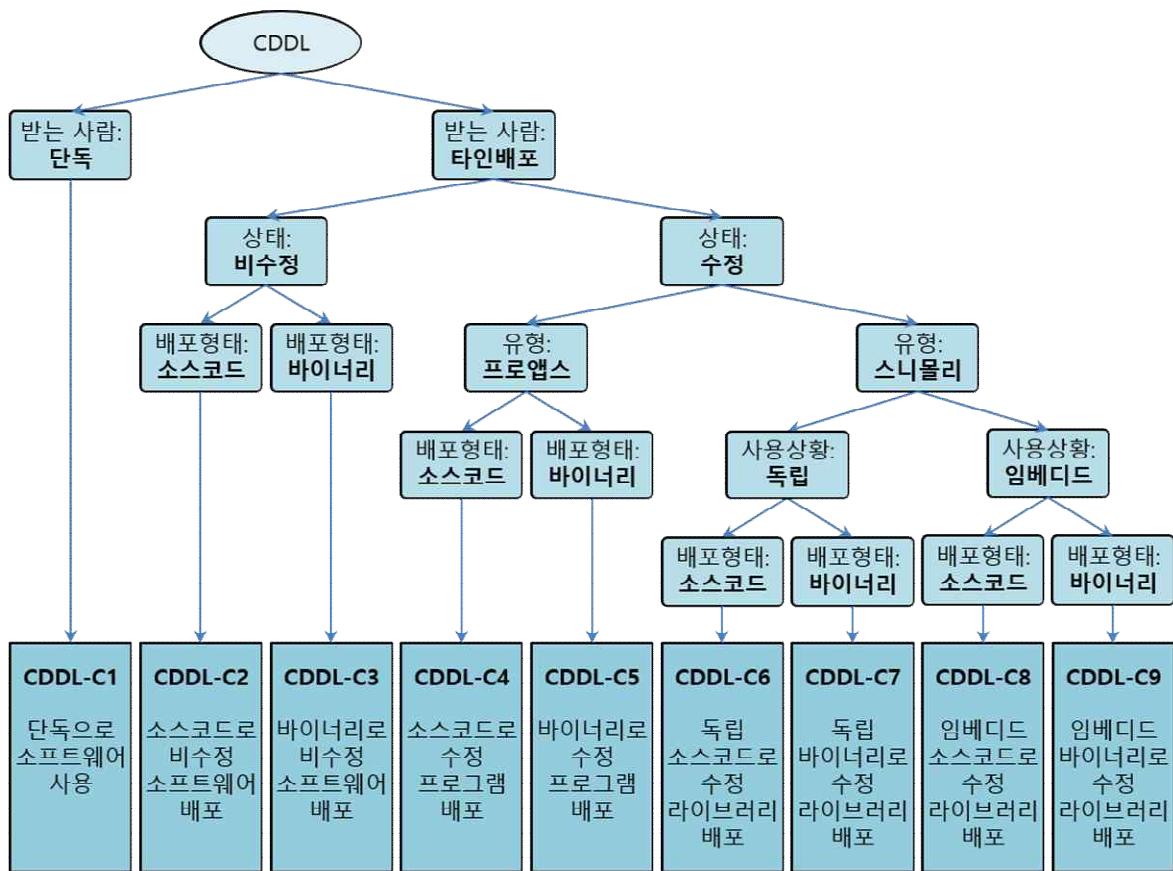
고지사항 및 BSD 면책조항)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드 패키지로부터 바이너리를 컴파일하고, 이 컴파일에서 라이선싱 파일을 자동으로 생성 및 포함하지 않을 경우, 카피라이트 고지사항, BSD 의무사항, 및 BSD 면책조항을 소스코드에서 찾은 형태로 생성하고, 해당 파일들을 사용자 배포버전에 수동으로 삽입해야 한다.

- 끝으로, 우리는 사용한 오픈소스 소프트웨어와 그 라이선스를 애플리케이션의 카피라이트 위젯의 일부로서, 언급하기 위해 일반 (오픈소스) 형식의 힌트를 삽입하길 권장해 왔다. 이는 BSD 라이선스에 의해 요구되지는 않으나, 일반적으로 권장되는 사항이다. 또한, 오픈소스 소프트웨어를 사용 및 수정하고 수정된 버전을 재배포할 수 있는 자유가 있으므로, 누락되더라도 레퍼런스를 삽입할 수 있다. 이에 따라 관련 오픈소스 사용 사례에 대한 세 번째 자발적 의무사항을 추가하였다.

## 6.5 CDDL 라이선스 소프트웨어 [추후보완]

또한, [...]

따라서 마찬가지로 간단하게 처리 가능한 작업 리스트를 찾기 위해 CDDL 특정 오픈소스 사용 사례 구조<sup>529)</sup>를 사용할 수 있다.



529) 일반적인 OSUC 파인더에 대한 자세한 사항 → OSLiC, pp.88

### **6.5.1 CDDL-1: 단독으로 소프트웨어 사용**

**의미:** 사용자는 제 3자에게 양도하지 않고, 단독으로 사용할 목적으로, CDDL 라이선스 소프트웨어를 획득하였다.

**적용범위 :** OSUC-01, OSUC-03, OSUC-06, OSUC-09530)

**요구사항:** ...

**금지사항:** ...

### **6.5.2 CDDL-2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포**

**의미:** CDDL 라이선스의 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>531)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

### **6.5.3 CDDL-3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포**

**의미:** 사용자는 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드 단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인 등 각기 다른 형태라도 배포하는 것을 차별하지 않는다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>532)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

---

530) 자세한 사항 → OSLiC, pp.96-106

531) 자세한 사항 → OSLiC, pp.96-106

532) 자세한 사항 → OSLiC, pp.96-106

#### **6.5.4 CDDL-4: 수정한 프로그램을 소스코드 형태로 배포**

**의미 :** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04S<sup>533)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

#### **6.5.5 CDDL-5: 수정한 프로그램을 바이너리 형태로 배포**

**의미 :** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 프로그램, 애플리케이션, 또는 서버(proapse)를 획득하였다.

**적용범위:** OSUC-04S<sup>534)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

#### **6.5.6 CDDL-6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포**

**의미 :** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>535)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

---

533) 자세한 사항 → OSLiC, pp.98

534) 자세한 사항 → OSLiC, pp.98

535) 자세한 사항 → OSLiC, pp.103

### **6.5.7 CDDL-7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포**

**의미 :** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 하지만, 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>536)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

### **6.5.8 CDDL-8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포**

**의미 :** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>537)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

### **6.5.9 CDDL-9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포**

**의미 :** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, CDDL 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>538)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

....

**금지사항:** ...

---

536) 자세한 사항 → OSLiC, pp.103

537) 자세한 사항 → OSLiC, pp.105

538) 자세한 사항 → OSLiC, pp.106

### 6.5.10 논의 및 설명

CDDL ~~~~의 거의 모든 요구를 포함하여 ~~~를 제공한다.<sup>539)</sup> 단지 일부 ~~를 위하여...

•

### 6.6 EPL-1.0 라이선스 소프트웨어

Eclipse Public 라이선스에서는 소스코드 형태의 배포와 바이너리 형태의 배포를 명확히 구분 짓는다. 먼저, 해당 라이선스는 소스코드와 오브젝트 코드의 이클립스 라이선스의 프로그램 배포를 허용한다.<sup>540)</sup> 그 다음, 이 조건 하에 바이너리 세트로서, 프로그램을 배포할 수 있음을 규정하고 있다.<sup>541)</sup> 이러한 조건 중, 배포자는 역시 소스코드를 사용할 수 있게 한다는 점이 대략적으로 설명되고 있다.<sup>542)</sup> 좀 더 자세히 말하면, EPL-1.0은 위크 카피레프트(Weak Copyleft) 라이선스로 간주할 수 있다. 또 다른 조건에서는 어떤 형태나 상태를 사용하는 것과 상관없이 일반적인 배포를 언급하고 있다.<sup>543)</sup> 그래서, 전체적으로 볼 때, EPL-1.0은 주로 소프트웨어 배포에 초점을 둔다. 따라서 적절하고 간단하게 처리 가능한 의무사항 리스트를 찾기 위해서는, 다음 EPL-1.0 상세 오픈소스 사용 사례 구조<sup>544)</sup>를 사용할 수 있다.

---

539) cf. *Open Source Initiative*: The CDDL-1.0, 2004, wp. §3.

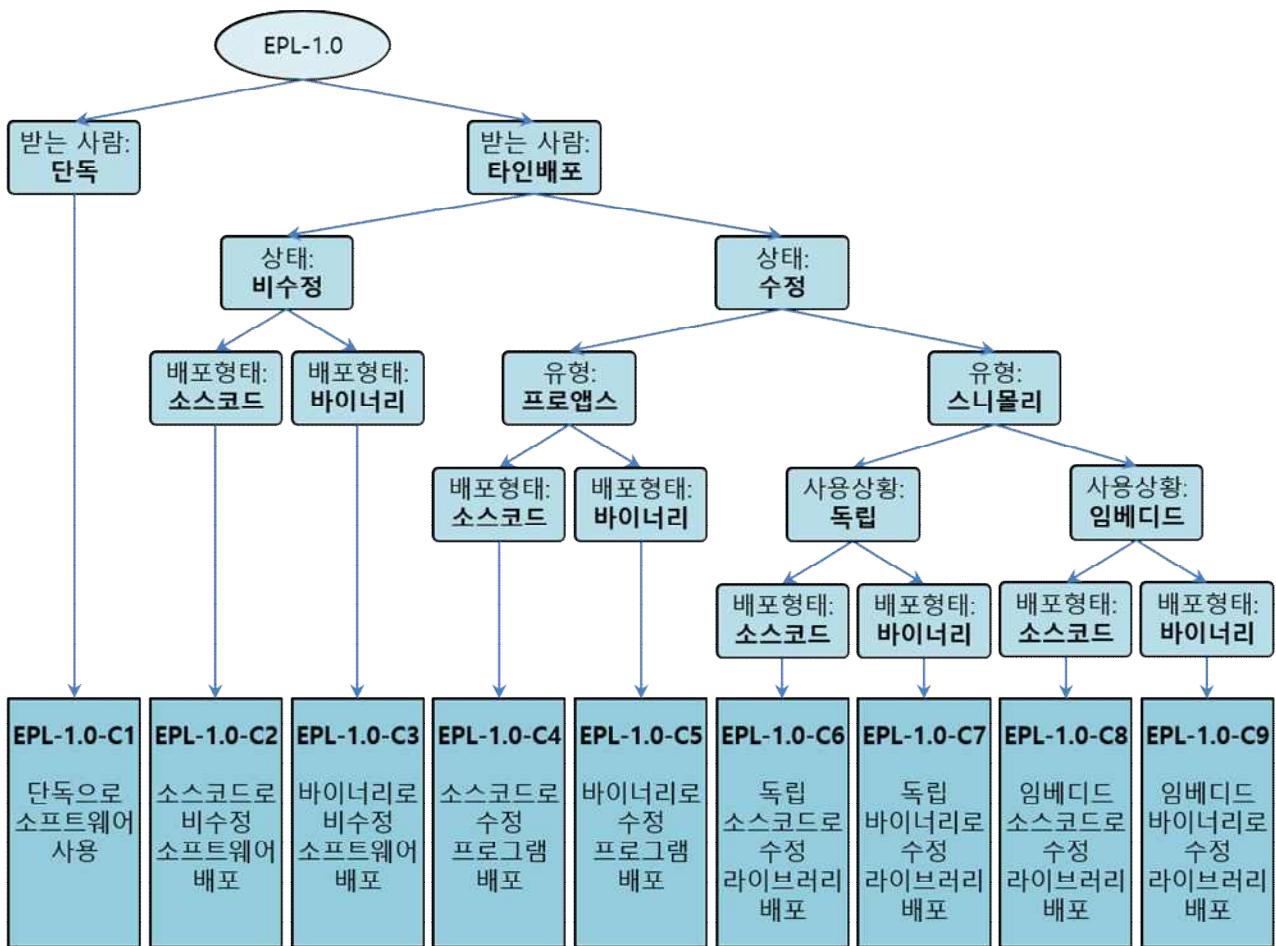
540) cf. *Open Source Initiative*: EPL-1.0, 2005, wp. §3.

541) cf. id., l.c., wp. §3 top area.

542) cf. id., l.c., wp. §3 mid area.

543) cf. id., l.c., wp. §3 bottom area.

544) 일반적인 OSUC 파인더에 대한 자세한 사항 → OSLiC, pp.88



### 6.6.1 EPL-1.0-C1: 단독으로 소프트웨어 사용

**의미:** 사용자는 EPL-1.0 라이선스의 소프트웨어를 획득하였으며, 본인만을 위해서 사용하고 어떤 의미에서든 제 3자에게 양도하지 않는 것을 의미한다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09NL<sup>545)</sup>

**요구사항:** Eclipse Public 1.0 라이선스의 조건을 충족하기 위한 의무사항은 없다.

- 사용자가 제 3자에게 소프트웨어를 배포하지 않는 한, 별 다른 의무사항 없이 어떠한 상황에서도 EPL-1.0 소프트웨어를 사용할 수 있다.

**금지사항:**

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

545) 자세한 사항→ OSLiC, pp.96-106

### 6.6.2 EPL-1.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포

**의미:** EPL-1.0 라이선스의 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>546)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 수령자에게 EPL-1.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다. 라이선싱 파일 내 라이선스 전문에서 명시적으로 해당 패키지를 EPL-1.0으로 선언하고 있지 않다면, 추가적으로 올바른 EPL-1.0 라이선싱 파일을 삽입해야 한다.<sup>547)</sup>
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

**금지사항:**

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.6.3 EPL-1.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포

**의미:** 사용자는 수정하지 않은 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 소프트웨어를 획득하였다. 이러한 경우, 독립적인 단위 또는 임베디드 단위로서, 사용자가 프로그램, 애플리케이션, 서버, 코드일부, 모듈, 라이브러리 또는 플러그인을 배포하는 것에 문제가 없다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>548)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태

546) 자세한 사항→ OSLiC, pp.96-106

547) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

548) 자세한 사항→ OSLiC, pp.96-106

그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.

- [필수적:] "보증의 부인"이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다.
- [필수적:] 사용자가 소프트웨어를 수정하지 않았더라도, 사용자의 컨트롤 하에 저장소를 통하여 소스코드를 이용할 수 있도록 해야 한다.
  - 소스코드 패키지를 인터넷 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 해당 저장소는 적당한 기간 동안 다운로드가 가능하도록 유지해야 한다.
- [필수적:] 사용자 배포물 또는 추가 자료에 다운로드 저장소에 대해 쉽게 확인할 수 있는 힌트를 삽입하고 소스코드를 어떻게 획득할 수 있는지 설명해야 한다.
- [필수적:] 소스코드 배포를 위해 EPL-1.0-C2 사용 사례의 과업 목록을 실행해야 한다.<sup>549)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지 사항 내용, 소프트웨어명에 대한 한트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

#### 금지사항:

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.4 EPL-1.0-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 프로그램, 애플리케이션, 또는 서버(prolapse)를 획득하였다.

**적용범위:** OSUC-04S <sup>550)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.
- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] 수령자에게 EPL-1.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다. 라이선싱 파일 내 라이선스 전문에서 명시적으로 해당 패키지를 EPL-1.0으로 선언하고 있지 않다면, 추가적으로 올바른 EPL-1.0 라이선싱 파일을 삽입해야 한다.<sup>551)</sup>

549) 저작권을 통해 소스코드에 액세스 가능함은 소프트웨어를 소스코드 형태로 배포하는 것을 의미한다. 그러므로, 사용자는 반드시 상용하는 사용 사례의 의무사항을 수행해야 한다.

550) 자세한 사항→ OSLiC, pp.98

551) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다. 만약 새로운 소스 코드 파일을 추가했다면, 사용자의 카피라이트 라인과 EPL-1.0의 적절한 라이선스 전문을 포함한 헤더부분을 삽입해야 한다.
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 컴퓨터들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다. 기존의 소스코드 카피라이트 화면을 업데이트하여 동일한 정보를 보여주도록 해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

**금지사항:**

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.5 EPL-1.0-C5: 수정한 프로그램을 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 프로그램, 애플리케이션, 또는 서버(prolapse)를 획득하였다.

**적용범위:** OSUC-04B 552)

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.
- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다.
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다. 기존의 소스코드 카피라이트 화면을 업데이트하여 동일한 정보를 보여주도록 해야 한다.
- [필수적:] 사용자의 컨트롤 하에 저장소를 통하여 프로그램의 소스코드를 이용할 수 있도록 해야 한다.
  - 소스코드 패키지를 인터넷 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 해당 저장소는 적당한 기간 동안 다운로드가 가능하도록 유지해야 한다.

---

552) 자세한 사항→ OSLiC, pp.99

- [필수적:] 사용자 배포물 또는 추가 자료에 다운로드 저장소에 대해 쉽게 확인할 수 있는 힌트를 삽입하고 소스코드를 어떻게 획득할 수 있는지 설명해야 한다.
- [필수적:] 소스코드 배포를 위해 EPL-1.0-C4 사용 사례의 과업 목록을 실행해야 한다<sup>553)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다. 특히, 카피라이트 하위섹션에 포함한다.

**금지사항:**

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.6 EPL-1.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snakemol)을 획득하였다. 하지만, 해당 파일들이 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08S<sup>554)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.
- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] 수령자에게 EPL-1.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다. 라이선싱 파일 내 라이선스 전문에서 명시적으로 해당 패키지를 EPL-1.0으로 선언하고 있지 않다면, 추가적으로 올바른 EPL-1.0 라이선싱 파일을 삽입해야 한다.<sup>555)</sup>
- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다. 만약 새로운 소스 코드 파일을 추가했다면, 사용자의 카피라이트 라인과 EPL-1.0의 적절한 라이선스 전문을 포함한 헤더부분을 삽입해야 한다.
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지

553) 저장소를 통해 소스코드에 액세스 가능함은 소프트웨어를 소스코드 형태로 배포하는 것을 의미한다. 그러므로, 사용자는 반드시 상용하는 사용 사례의 의무사항을 수행해야 한다.

554) 자세한 사항→ OSLiC, pp.103

555) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

#### 금지사항:

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.7 EPL-1.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(sniolib)을 획득하였다. 하지만, 해당 파일들이 다른 큰 소프트웨어 단위에 탑재된 것은 아니다.

**적용범위:** OSUC-08B<sup>556)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.
- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다.
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가해야 한다.
- [필수적:] 사용자의 컨트롤 하에 저장소를 통하여 수정된 라이브러리의 소스코드를 이용할 수 있도록 해야 한다.
  - 소스코드 패키지를 인터넷 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 해당 저장소는 적당한 기간 동안 다운로드가 가능하도록 유지해야 한다.
- [필수적:] 사용자 배포물 또는 추가 자료에 다운로드 저장소에 대해 쉽게 확인할 수 있는 힌트를 삽입하고 소스코드를 어떻게 획득할 수 있는지 설명해야 한다.
- [필수적:] 소스코드 배포를 위해 EPL-1.0-C6 사용 사례의 과업 목록을 실행해야 한다<sup>557)</sup>
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지; 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

556) 자세한 사항→ OSLiC, pp.103

557) 저장소를 통해 소스코드에 액세스 가능함은 소프트웨어를 소스코드 형태로 배포하는 것을 의미한다. 그러므로, 사용자는 반드시 상응하는 사용 사례의 의무사항을 수행해야 한다.

### 금지사항:

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.6.8 EPL-1.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** 사용자는 수정하여 소스코드 파일 또는 소스코드 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드 일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10S<sup>558)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.
- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] 수령자에게 EPL-1.0 라이선스의 사본을 제공해야 한다. 소프트웨어 패키지의 일부에서 빠져 있다면 이를 추가해야 한다. 라이선싱 파일 내 라이선스 전문에서 명시적으로 해당 패키지를 EPL-1.0으로 선언하고 있지 않다면, 추가적으로 올바른 EPL-1.0 라이선싱 파일을 삽입해야 한다.<sup>559)</sup>
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가하고, 사용자 소유의 중요 프로그램의 카피라이트 화면에서 임베디드 컴포넌트에 대한 상세사항과 동일한 정보를 표시해야 한다.
- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다. 만약 새로운 소스코드 파일을 추가했다면, 사용자의 카피라이트 라인과 EPL-1.0의 적절한 라이선스 전문을 포함한 헤더부분을 삽입해야 한다.
- [자발적:] 포함된 EPL-1.0 라이선스와 라이선싱 파일이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 소스코드 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지 사항 내용, 소프트웨어명에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다.

558) 자세한 사항→ OSLiC, pp.105

559) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

특히, 카피라이트 하위섹션에 포함한다.

#### 금지사항:

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.9 EPL-1.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미 :** 사용자는 수정하여 바이너리 파일 또는 바이너리 패키지 형태로 제 3자에게 배포할 목적으로, EPL-1.0 라이선스의 코드일부, 모듈, 라이브러리, 또는 플러그인(snimoli)을 획득하였다. 해당 코드일부, 모듈, 라이브러리, 또는 플러그인은 임베디드 컴포넌트로서, 다른 큰 소프트웨어 단위에 포함된다.

**적용범위:** OSUC-10B<sup>560)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항이 다음과 같다.

- [필수적:] 라이선스 요소(특히 카피라이트 고지사항, 보증의 부인 및 책임의 면책)는 획득한 형태 그대로 사용자의 패키지에 포함해야 한다. 사용자가 소스코드로부터 바이너리를 컴파일 할 경우, 라이선스 요소 또한 해당 패키지에 전부 포함되어야 한다.
- [필수적:] 수정사항에 대한 텍스트파일이 존재하지 않을 경우, 생성하고, 사용자의 수정사항에 대한 L

일반적인 상세설명을 수정사항 텍스트 파일에 추가한 후, 이를 배포 패키지에 포함시켜야 한다.

- [필수적:] 모든 소스코드의 수정사항을 프로그램의 수정된 소스코드 내에 모두 표시해야 한다.
- [필수적:] “보증의 부인”이라는 명확한 문구가 배포된 소스코드 패키지 안에 존재하지 않는다면, 이를 포함시켜야 한다. 문구의 내용으로는, 해당 소프트웨어에 기여한 모든 기여자들에게 소프트웨어 퀄리티에 대한 어떠한 책임도 전가할 수 없다라는 것을 확실히 명시해야 한다. 그런 다음, 파일 내에 EPL-1.0 라이선스로부터의 보증의 부인과 책임의 면책 조항을 추가하고, 사용자 소유의 중요 프로그램의 카피라이트 화면에서 임베디드 컴포넌트에 대한 상세사항과 동일한 정보를 표시해야 한다.
- [필수적:] 사용자의 컨트롤 하에 저장소를 통하여 임베디드 라이브러리의 소스코드를 이용할 수 있도록 해야 한다.
  - 소스코드 패키지를 인터넷 저장소에 업로드한 후, 인터넷을 통해 다운로드가 가능하도록 해야 한다. 해당 저장소는 적당한 기간 동안 다운로드가 가능하도록 유지해야 한다.
- [필수적:] 사용자 배포물 또는 추가 자료에 다운로드 저장소에 대해 쉽게 확인할 수 있는 힌트를 삽입하고 소스코드를 어떻게 획득할 수 있는지 설명해야 한다.
- [필수적:] 기존의 EPL 라이선스가 적용되는 방식으로 수정물을 구성해야 한다.
- [필수적:] 소스코드 배포를 위해 EPL-1.0-C8 사용 사례의 과업 목록을 실행해야 한다<sup>561)</sup>

560) 자세한 사항→ OSLiC, pp.107

561) 저장소를 통해 소스코드에 액세스 가능함은 소프트웨어를 소스코드 형태로 배포하는 것을 의미한다. 그러므로,

- [자발적:] 포함된 EPL-1.0 라이선스와 라이선싱 파일이 임베디드 라이브러리인 것을 확실히 명시하고, 사용자 소유의 중요 저작물의 라이선싱에 영향을 미치지 않도록 사용자의 바이너리 배포를 준비한다. 추가적으로 라이브러리, 모듈, 코드일부 또는 플러그인과 같은 임베디드 컴포넌트를 모든 추가적인 라이선싱 요소들을 포함하는 분리된 디렉토리에 저장하는 것을 권장한다.
- [자발적:] 사용자의 배포 그리고/또는 사용자의 추가 자료의 문서 또한, 기존의 카피라이트 고지 사항 내용, 소프트웨어에 대한 힌트, 홈페이지 링크, EPL-1.0 라이선스 링크를 포함해야 한다. 특히, 카피라이트 하위섹션에 포함한다.

#### 금지사항:

- 획득한 소프트웨어 패키지 내에 포함된 카피라이트 고지 제거 또는 변경
- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.6.10 논의 및 설명

EPL-1.0의 간결한 “요구사항”<sup>562)</sup> 섹션에서는 “상업용 배포(Commercial Distribution)”<sup>563)</sup>와 관한 정의를 보충하는 내용을 포함하고 있다. 먼저 배포자가 바이너리 세트로서, 이를립스 라이선스 프로그램을 올바르게 배포하기 위한 의무사항에 대해 설명하고 있다. 그 다음에 소스코드로서의 소프트웨어를 배포하기 위해 준수해야하는 라이선스 의무사항에 대해 설명한다. 마지막으로 어떠한 경우에도 이행되어야 하는 두 가지 조건을 열거하고 있다.<sup>564)</sup> 이러한 구조와 관련하여, 우리는 다음과 같은 의무사항을 발견할 수 있다.

- EPL-1.0은 일반적으로 “기여자는 프로그램에 포함된 카피라이트 고지사항을 삭제 또는 변경할 수 없다”라는 의무사항을 요구한다.<sup>565)</sup> 여기서 ‘기여자’라는 단어는 프로그램을 배포하는 모든 사람 또는 실체(entity)로 해석하고, ‘프로그램’이라는 단어는 최초 기여물 및 그것의 수정물을 의미한다.<sup>566)</sup> 적어도 엄격한 해석에서의 EPL-1.0은 EUPL과 마찬가지로 이러한 요구사항을 소프트웨어 배포(타인에 대한 배포)에 제한하지 않는다. 하지만, 실제로 소프트웨어를 단독으로 사용하면서 소프트웨어에 대한 의무사항을 준수하며 사용하기는 어려울 것이다. 그러나 EUPL과 반대로 EPL-1.0은 명확하게 이 금지사항을 포함하고 있다. OSLiC는 메시지를 복제하여 실제적으로 불일치로 이어지는 부분을 해결한다. 한편으로는 타인에게 배포하는 사용 사례(EPL-1.0-C2 – EPL-1.0-C9)에 대해 부정적인 조항을 필수적이고 긍정적인 조항으로 다시 고쳐 쓰고 있다. 이는 획득한 그대로의 형태와 동일한 형태의 카피라이트 고지사항을 유지하는 활동을 강조한다. 이와 달리 OSLiC에서는 이에 상응하는 금지사항을 단독으로 사용(EPL-1.0-C1 – EPL-1.0-C9)의 금지사항 섹션에 삽입하고 있다.
- 게다가, EPL-1.0은 “각 기여자는 반드시 수령자가 기여물의 창작자를 확인할 수 있는 방식으로

사용자는 반드시 상응하는 사용 사례의 의무사항을 수행해야 한다.

562) Open Source Initiative: EPL-1.0, 2005, wp §3 참고

563) 위와 동일, wp §4 참고.

564) 위와 동일, wp §3 참고.

565) 위와 동일

566) 위와 동일, wp §1 참고.

기여물의 창작자임을 밝혀야 한다.”<sup>567)</sup>라는 의무사항을 요구하고 있다. 이러한 경우에, 기여물(Contribution)은 기여물의 모든 후속 수정사항을 포함한 초기 코드와 문서로 해석해야 한다.<sup>568)</sup> 이러한 의무사항을 성실하게 준수하기 위해, 개발자는 소스코드 내 자신의 수정사항을 표시하고, 상세설명을 기입해야 한다. 그리고 배포자는 특정 파일(일반적으로는 CHANGES라고 불림)에 소프트웨어 기능의 일반적인 수준에서 이러한 수정사항에 대한 상세설명을 기입해야 한다. 언뜻 보기에는 소스코드 내에 소스코드 수정사항을 문서화하는 요구사항이 소스코드 형태의 수정된 EPL-1.0 소프트웨어의 배포에 대한 사용 사례로 제한된 것처럼 보인다. 하지만, EPL-1.0에서는 배포자가 해당 코드를 획득할 수 있는 위치를 명시하는 경우에도 바이너리 형태의 배포를 허용한다.<sup>569)</sup> 따라서 바이너리 배포는 소스코드 배포를 암시한다. 이에 따라 OSLiC에서는 수정된 EPL-1.0 소프트웨어(EPL-1.0-C4 - EPL-1.0-C9)의 배포에 관한 모든 사용사례에 대해서 두 가지 필수적인 조항을 추가하고 있다.

- 소스코드 형태의 모든 배포에 대해 EPL-1.0에서는 “소프트웨어는 EPL-1.0에 기반하여 사용할 수 있으며, 이 라이선스의 사본은 해당 프로그램의 각 복제버전에 반드시 포함되어야 한다.<sup>570)</sup>”라고 요구하고 있다. 따라서 OSLiC는 이러한 사용 사례(EPL-1.0-C4, EPL-1.0-C6, EPL-1.0-C8)에 대해 각각의 필수적인 조항을 추가하고 있다. 하지만, EPL-1.0은 위크 카피레프트(Weak Copyleft)<sup>571)</sup> 라이선스이다. 이에 따라, 이러한 조항들은 임베디드 라이브러리(EPL-1.0-C8)을 사용한 중요 프로그램에 적용되지 않는다.
- 추가적으로 EPL은 매우 범용적인 의미<sup>572)</sup>에서 배포자가 모든 기여자를 대신하여 보증 및 조항들을 효과적으로 부인하고 손해에 대한 책임을 배제할 경우, 바이너리 형태의 소프트웨어 배포를 허용한다. 이러한 책임에 대한 제한을 정하는 것은 EPL-1.0에서 매우 중요하다. 따라서 EPL-1.0에서는 상업적 배포(Commercial Distribution) 섹션에서 한 번 더 이러한 측면을 세분화하여 설명하고 있다. 여기에서는 더 이상 바이너리 형태의 소프트웨어 배포에만 중점을 두고 있지는 않는다.<sup>573)</sup> 따라서 OSLiC에서는 배포와 관련된 모든 사용 사례(EPL-1.0의 보증의 부인<sup>574)</sup>과 책임의 면책<sup>575)</sup>에 대한 단락은 배포 패키지 및 카피라이트 화면(기술적으로 가능할 경우)에서 반드시 명시적으로 표시되어야 한다)에 대해 필수적인 조항을 추가하고 있다.
- 이와 별개로, EPL-1.0은 바이너리 형태로의 소프트웨어 배포를 다음 조건이 충족될 경우에만 허용한다. 배포자가 “기여자 (배포자)로부터 프로그램 소스코드를 받을 수 있다”고 명확하게 말해야 하며, 추가로 “사용자들이 (소스코드를) 받을 수 있는 합리적인 방법을 알려주어야 한다.”<sup>576)</sup> 이 요구사항은 배포자 자신이 저장소를 통해 소스코드를 제공하지 않는 이상 충족하기 어렵다. 기타

567) Open Source Initiative: EPL-1.0, 2005, wp §3 참고

568) 위와 동일, wp §1 참고.

569) 위와 동일, wp §3 참고.

570) 위와 동일

571) (→ OSLiC, p.28)

572) 위와 동일

573) 위와 동일, wp §4 참고.

574) Open Source Initiative: EPL-1.0, 2005, wp §5 참고

575) 위와 동일, wp §6 참고.

576) 위와 동일, wp §3 참고.

외부 저장소 위치를 알려주는 것만으로는 불충분하다. 따라서 바이너리 형태의 배포와 연관이 있는 모든 사용사례에 대해, OSLiC은 요구사항을 EPL-1.0에 적힌대로 적용한다 (EPL-1.0-C3, EPL-1.0-C5, EPL-1.0-C7, EPL-1.0-C9).

- 또한 앞의 조건은 실제 소스코드 배포를 암시한다고 명시해야하며, 이에 따라 소프트웨어 배포의 조건들 또한 적용된다. OSLiC에서는 바이러니 배포의 경우에 상응하는 소스코드 사용 사례의 요구사항도 지켜져야한다고 정의한다.
- 마지막으로, EPL-1.0은 특히 관련 조항을 포함한다. “프로그램은 받은 자가 프로그램이 자신의 특허를 침해한다고 주장하며 소송을 제기할 경우, (EPL-1.0이 허용한) 사용자의 권한을 종료시킨다.”<sup>577)</sup> 이에 따라 OSLiC에서는 소프트웨어와 연관된 특히 관련 법적 대응을 금한다 (EPL-1.0-C1 – EPL-1.0-C9).

## 6.7 EUPL-1.1 라이선스 소프트웨어

European Union Public License에서는 소스코드 배포와 바이너리 배포를 명확히 구분 짓고 있다. 라이선스의 제 3장인 “소스코드 전달(Communication of the Source Code)”에 따르면, “소스코드 형태로 또는 실행 가능 코드(Executable Code)로 저작물을 제공”할 수 있다.<sup>578)</sup> 그러나 EUPL 라이선스 소프트웨어의 일부가 바이너리 패키지로 배포된다면 라이선스에서는 배포자가 바이너리와 함께 추가로 직접 “기계 판독이 가능한 소스코드 사본을 제공”하거나<sup>579)</sup> “저작권자가 저작물을 지속적으로 배포하는 동안 소스코드를 쉽고 자유롭게 이용할 수 있는 저장소를 표기할 것을 요구한다.”<sup>580)</sup> 소프트웨어가 수정되었는지 여부 및 기타 모든 “라이선스 의무사항”이 위 두 가지 형태에 해당하는지의 여부와 상관없이 이 조건들은 준수되어야 한다.<sup>581)</sup>

그러나 EUPL-1.1 라이선스 준수를 위해 고려해야 하는 특징이 있다. 라이선스 문구 그대로 해석하면, EUPL은 분명히 약한 카피레프트를 갖는 라이선스다. 그러나 이 특징은 저작권자가 라이선스의 “호환성 조항”<sup>582)</sup> 및 부록에 기재된, 약한 카피레프트를 갖는 라이선스의<sup>583)</sup> 조건을 만족할 때에 소프트웨어를 다시 라이선싱 할 수 있다는 특징에 의거한 것이다. 따라서 라이선스 준수를 잘 하기 위한 방법에 대한 질문과 관련하여 EUPL은 강한 효력의 카피레프트를 갖는 라이선스로 간주하는 것이 안전하겠다. 라이브러리를 임베디드 컴포넌트로 사용하는 경우에 있어서, 강한 카피레프트 라이선스일 경우 수정하지 않은 라이브러리를 사용할 시에도 동일한 카피레프트 라이선스를 적용해야 하다고 암시하고 있다. 따라서 간단히 처리 가능한 작업 리스트를 알아보기 위해 다음의 EUPL-1.1 오픈소스 사용 사례 구조를 참고할 수 있다.<sup>584)</sup>

577) 위와 동일, wp §7 참고.

578) Open Source Initiative: EUPL-1.1 (OSI), 2007, §3. 참조

579) 인용한 글과 동일, wp. §5. 참조

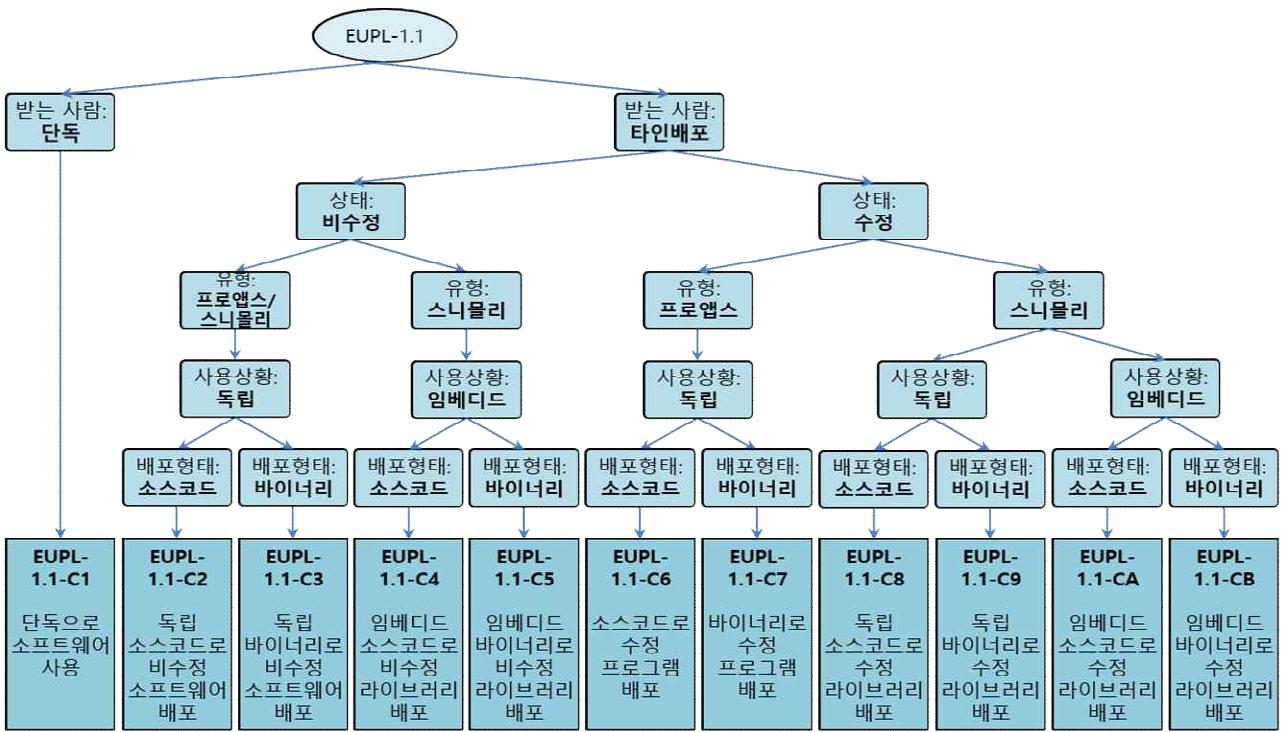
580) 인용한 글과 동일, wp. §3. 참조

581) 인용한 글과 동일, wp. §5. 참조

582) 인용한 글과 동일, wp. §5. 참조

583) (→ OSLiC, p. 31)

584) 기본 오픈소스 사용 사례 파인더에 대한 설명은 OSLiC, pp. 88 참조



### 6.7.1 EUPL-1.1-C1: 단독으로 소프트웨어 사용

**의미:** 사용자는 EUPL-110 라이선스의 소프트웨어를 획득하였으며, 본인만을 위해서 사용하고 어떤 의미에서든 제 3자에게 양도하지 않는다.

**적용 범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>585)</sup>

**요구사항:** 해당 사용 사례에 관련하여 EUPL-1.1 라이선스를 준수하기 위해 지켜야 할 의무사항은 없다.

- 사용자가 제 3자에게 소프트웨어를 배포하지 않는 한, 별 다른 의무사항 없이 어떠한 상황에서도 EUPL-1.1 소프트웨어를 사용할 수 있다.

#### 금지 사항:

- 소프트웨어 파일을 설명할 때 합리적이고, 관례적인 사용이 요구되는 경우를 제외하고, 소프트웨어 와 연관된 제품명, 트레이드 마크, 서비스 마크에 기반하여, 사용자의 서비스를 홍보

### 6.7.2 EUPL-1.1-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포

**의미:** EUPL-1.1로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 소스코드 파일 또는 소스코드 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

585) 자세한 내용은 OSLiC, pp. 96-106 참조

**적용범위:** OSUC-02S, OSUC-05S<sup>586)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>587)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

### 6.7.3 EUPL-1.1-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포

**의미:** EUPL-1.1로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>588)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>589)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 코드를 찾고 받을 수 있는 방법에 대한 설명을 배포 패키지 내에 포함한다. 이 저장소는 소프트웨어 배포를 지속하는 동안 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 EUPL-1.1-C2의 과업 목록을 수행한다.<sup>590)</sup>

586) 자세한 내용은 OSLiC, pp. 96-106 참조

587) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

588) 자세한 내용은 OSLiC, pp.96-106 참조

589) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

590) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

#### 6.7.4 EUPL-1.1-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** EUPL-1.1로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 소스코드 파일 또는 소스코드 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07S<sup>591)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특히 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>592)</sup>
- [필수적:] 라이브러리를 포함한 프로그램을 EUPL-1.1로 라이선싱 한다. 온탑 개발에도 EUPL-1.1 라이선스 의무사항이 적용될 수 있도록 소스를 구성한다.
- [자발적:] 온탑 개발의 저작권 문구에 EUPL-1.1 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 EUPL-1.1으로 라이선싱 되었다는 것을 명확히 표시한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

#### 6.7.5 EUPL-1.1-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** EUPL-1.1로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 바이너리 파일 또는 바이너리 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

591) 자세한 내용은 OSLiC, pp.102 참조

592) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

**적용범위:** OSUC-07B<sup>593)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>594)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 코드를 찾고 받을 수 있는 방법에 대한 설명을 배포 패키지 내에 포함한다. 이 저장소는 소프트웨어 배포를 지속하는 동안 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 라이브러리를 포함한 프로그램을 EUPL-1.1로 라이선싱 한다. 온탑 개발에도 EUPL-1.1 라이선스 의무사항이 적용될 수 있도록 바이너리를 구성한다.
- [필수적:] 사용 사례 EUPL-1.1-C4의 과업 목록을 수행한다.<sup>595)</sup>
- [자발적:] 온탑 개발의 저작권 문구에 EUPL-1.1 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 EUPL-1.1으로 라이선싱 되었다는 것을 명확히 표시한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

#### 6.7.6 EUPL-1.1-C6: 수정한 프로그램을 소스코드 형태로 배포

**의미 :** EUPL-1.1로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 소스코드 파일 형태 또는 소스코드 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04S<sup>596)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한을 뜻하는 고지문 등은 받았던 형태로 패키지에 포함한다.

593) 자세한 내용은 OSLiC, pp. 102 참조

594) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

595) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

596) 자세한 내용은 OSLiC, pp.98 참조

- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지의 일부에서 빠져있다면 이를 추가한다.<sup>597)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 프로그램 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 EUPL-1.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

#### 금지 사항:

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

#### 6.7.7 EUPL-1.1-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** EUPL-1.1로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 바이너리 파일 형태 또는 바이너리 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04B<sup>598)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>599)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 기존의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다.
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 코드를 찾고 받을 수 있는 방법에 대한 설명을 배포 패키지 내에 포함한다. 이 저장소는 소프트웨어 배포를 지속하는 동안 온라인에서 사용할 수 있도록 한다.

597) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

598) 자세한 내용은 OSLiC, pp.99 참조

599) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 EUPL-1.1-C6의 과업 목록을 수행한다.<sup>600)</sup>
- [자발적:] 프로그램 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

#### 금지 사항:

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 흥보

### 6.7.8 EUPL-1.1-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미 :** EUPL-1.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>601)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>602)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 라이브러리 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 EUPL-1.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

#### 금지 사항:

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고

600) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

601) 자세한 내용은 OSLiC, pp.103 참조

602) 자세한 내용은 OSLiC, pp.103 참조

EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

### 6.7.9 EUPL-1.1-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미 :** EUPL-1.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08B<sup>603)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특히 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>604)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 기존의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다.
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 코드를 찾고 받을 수 있는 방법에 대한 설명을 배포 패키지 내에 포함한다. 이 저장소는 소프트웨어 배포를 지속하는 동안 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 EUPL-1.1-C8의 과업 목록을 수행한다.<sup>605)</sup>
- [자발적:] 라이브러리 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

603) 자세한 내용은 OSLiC, pp.109 참조

604) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

605) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

### 6.7.10 EUPL-1.1-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미** : EUPL-1.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위**: OSUC-10S<sup>606)</sup>

**요구사항**: 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특허 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>607)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 기존의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 EUPL-1.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.
- [필수적:] 라이브러리를 포함한 프로그램을 EUPL-1.1로 라이선싱 한다. 온탑 개발에도 EUPL-1.1 라이선스 의무사항이 적용될 수 있도록 소스코드를 구성한다.
- [필수적:] 임베디드된 라이브러리의 소스코드 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [자발적:] 온탑 개발의 저작권 문구에 EUPL-1.1 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 EUPL-1.1으로 라이선싱 되었다는 것을 명확히 표시한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항**:

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

### 6.7.11 EUPL-1.1-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미** : EUPL-1.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를

606) 자세한 내용은 OSLiC, pp.105 참조

607) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

갖고 있다.

**적용범위:** OSUC-10B<sup>608)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 저작권, 특히 또는 상표 고지문 그리고 라이선스와 보증 제한 고지문 등은 받았던 형태로 패키지에 포함한다. 소스에서 바이너리를 컴파일한다면 라이선스 요소는 전부 패키지에 포함하도록 한다.
- [필수적:] 받는 사람에게 EUPL-1.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>609)</sup>
- [필수적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 설명을 추가한다.
- [필수적:] 자신이 배포한 임베디드 라이브러리 및 온탑 개발의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 코드를 찾고 받을 수 있는 방법에 대한 설명을 배포 패키지 내에 포함한다. 이 저장소는 소프트웨어 배포를 지속하는 동안 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 EUPL-1.1-CA의 과업 목록을 수행한다.<sup>610)</sup>
- [필수적:] 기준의 EUPL-1.1 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다.
- [필수적:] 라이브러리를 포함한 프로그램을 EUPL-1.1로 라이선싱 한다. 온탑 개발에도 EUPL-1.1 라이선스 의무사항이 적용될 수 있도록 바이너리를 구성한다.
- [자발적:] 임베디드 라이브러리 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 저작권 고지문 텍스트 파일 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, EUPL-1.1 라이선스 링크를 제공한다.

**금지 사항:**

- 사용한 소프트웨어 출처를 공정하게 설명하고 저작권 고지문을 복제하는데 필요한 경우를 제외하고 EUPL-1.1 소프트웨어에 연관된 명칭, 상표명, 상표, 서비스 마크에 의한 서비스 또는 제품의 홍보

### 6.7.12 논의 및 설명

- EUPL-1.1에서는 일반적으로 “저작물의 출처를 설명하고 저작권 고지문 내용을 복제하는 것에 합당하고 관례적으로 사용할 필요가 있는 경우를 제외하고는 상표명, 상표, 서비스 마크 또는 저작권자 이름 사용을 허가하지 않는다.”<sup>611)</sup> 따라서 OSLiC에서는 그 요소들로 어떠한 서비스 또는

608) 자세한 내용은 OSLiC, pp.106 참조

609) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

610) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

제품 홍보를 금한다(EUPL-1.1-C1 ~ EUPL-1.1-CB).

- EUPL-1.1에서는 일반적으로 “저작권자는 모든 저작권, 특히 또는 상표 고지문 그리고 라이선스와 보증 제한 문구를 그대로 두어야 한다.”<sup>612)</sup> 엄격하게 해석하면 EUPL-1.1에서는 이 의무사항의 적용을 소프트웨어 배포에만 제한하지 않고 다른 경우에도 확대한다. 그러나 현실적으로는 배포가 이루어지지 않는 한 (즉 자신이 개인적으로 사용할 경우), 라이선스가 준수되고 있는지 확인 및 통제가 불가능하다. 이에 따라, OSLiC에서는 이 요구사항을 타인 배포 사용 사례에만 필수 의무 사항으로 표시한다 (EUPL-1.1-C2 ~ EUPL-1.1-CB).
- 또한 이 라이선스에서는 “배포된 모든 저작물과 함께 EUPL-1.1 라이선스 사본을 포함해야 한다.”<sup>613)</sup> 이에 따라 모든 타인 배포 사용 사례에 이 의무사항을 포함한다 (EUPL-1.1-C2 ~ EUPL-1.1-CB).
- 또한 EUPL-1.1은 “파생저작물을 배포할 시에는 저작물이 수정되었다는 설명과 수정 날짜를 나타낸 고지문을 포함해야 한다”<sup>614)</sup>고 규정하고 있다. 따라서 OSLiC에서는 코드를 수정했을 모든 사용 사례에 수정한 부분에 대한 고지문 파일 생성(업데이트)을 의무사항으로 포함시켰다 (EUPL-1.1-C6 ~ EUPL-1.1-CB).
- EUPL-1.1에서는 소프트웨어 배포자가 “저작물을 배포하는 한 소스를 쉽고 자유롭게 이용할 수 있는 저장소를 (명시하고) “소스코드의 기계 판독 가능한 사본을 제공”해야 한다.<sup>615)</sup> 이에 따라, OSLiC에서는 바이너리를 배포하는 모든 사용 사례에 해당 의무사항을 포함시켰다 (EUPL-1.1-C3, EUPL-1.1-C7, EUPL-1.1-C9, EUPL-1.1-CB).
- 마지막으로, 이 라이선스에는 “원저작물 또는 파생저작물의 사본을 배포할 경우 EUPL-1.1 라이선스 조항에 따라 배포한다는 카피레프트 조항”을 포함한다. 임베디드 컴포넌트 사용과 관계되지 않는 사용 사례 (EUPL-1.1-C2 ~ EUPL-1.1-C9)에서 이 카피레프트 조항은 수정된 소스를 배포하거나 저장소를 통해 이를 이용할 수 있게 하므로 이미 충족이 된다. EUPL-1.1로 라이선스 된 컴포넌트를 프로그램에 임베디드된 형태로 배포하는 경우 (EUPL-1.1-CA ~ EUPL-1.1-CB), 전체 프로그램의 소스코드 또한 공개되어야 한다. EUPL-1.1-CA 사용 사례와 관련하여 이는 정의에 의해 이미 충족되고 있다. 따라서 OSLiC에서는 EUPL-1.1-CB 사례에서만 이 조항을 관점에서만 언급하고 있다.<sup>616)</sup>

611) Open Source Initiative: EUPL-1.1 (OSI), 2007, wp §5. 참조

612) 위와 동일

613) 위와 동일

614) 위와 동일

615) European Community a. European commission Joinup: EUPL-1.1/EN, 2001, wp §5. 참조. 더 정확히 말하면 EUPL-1.1에서는 바이너리 패키지와 함께 소스코드를 직접 배포할 수 있다(인용한 글과 동일, wp. §3 참조). 안정적인 방법만을 제공한다는 OSLiC 원칙에 의하여 OSLiC에서는 이 옵션을 간소화하였다(저장소 해결방법만을 요구한다)

616) 공식적으로 EUPL-1.1은 약한 카피레프트를 갖는 라이선스다. 그러나 이는 소프트웨어를 다시 라이선싱 하는 것을 허용하기 때문이다(→ OSLiC, 31쪽). 따라서 “EUPL-1.1의 제5항에 따라 적합한 라이선스” 목록과 관련하여 임베디드 라이브러리를 다시 라이선싱하지 않는 한 임베디드된 EUPL-1.1 컴포넌트가 포함된 저작물의 코드를 공개해야 한다.

## 6.8 GPL 라이선스 소프트웨어

GNU General Public License의 두 버전은 소스코드 배포와 바이너리 배포를 명확히 구분 짓고 있다. GPL-2.0에서는 소스코드 복사와 배포에 대해 주로 다루지만<sup>617)</sup> 또한 “오브젝트 코드 또는 실행 가능 형태의 프로그램을 복사하고 배포하기 위한” 구체적 조건에 대해서도 언급하고 있다.<sup>618)</sup> 반면에, GPL-3.0에서는 “비소스형태 전달(Conveying Non-Source-Form)” 규정을 설명하기에 앞서<sup>619)</sup> “Verbatim Copy 전달” 또는 “수정된 소스 버전 전달(Conveying Modified Source Version)”을 위한 조건과 “기본 허가(Basic Permission)”를 설명한다.<sup>620)</sup>

또한 GPL-2.0와 GPL-3.0에서는 주로 소프트웨어 복사와 배포에 대해 설명한다(개인사용은 거의 명시하지 않고 있다). GPL-2.0에서는 “프로그램 사본”<sup>621)</sup>을 복제하고 배포하는 경우의 제약사항만을 언급하고 있으며, GPL-3.0에서는 “배포하지 않는 한 제약없이 라이선스 된 프로그램을 사용, 작성, 실행 및 확장할 수 있다”고 명시하고 있다.<sup>622)</sup> 또한 GPL은 강한 카피레프트를 갖는 라이선스로서 GPL로 라이선싱된 라이브러리를 포함하는 애플리케이션은 라이브러리와 동일한 라이선스 (즉 GPL)로 라인선싱 되어야 한다고 규정한다.

마지막으로, GPL-2.0와 GPL-3.0에서는 라이선스 준수를 위해 실행되어야하는 작업 목록이 거의 동일하다. 따라서 같은 챕터 안에서 두 라이선스를 다루고 공통의 오픈소스 사용 사례 구조를 제공하는 것이 적절하겠다.<sup>623)</sup> 그러나 과업 목록 자체는 별도로 제공한다.

다음 그림에서 GPL-\*-\*C1 (GPL-\*-\*C2, ..., GPL-\*-\*CB)은 GPL-2.0-C1 혹은 GPL-3.0-C1을 뜻하고 있다.

---

617) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §1, §2. 참조

618) 위 책과 동일, wp. §3. 참조

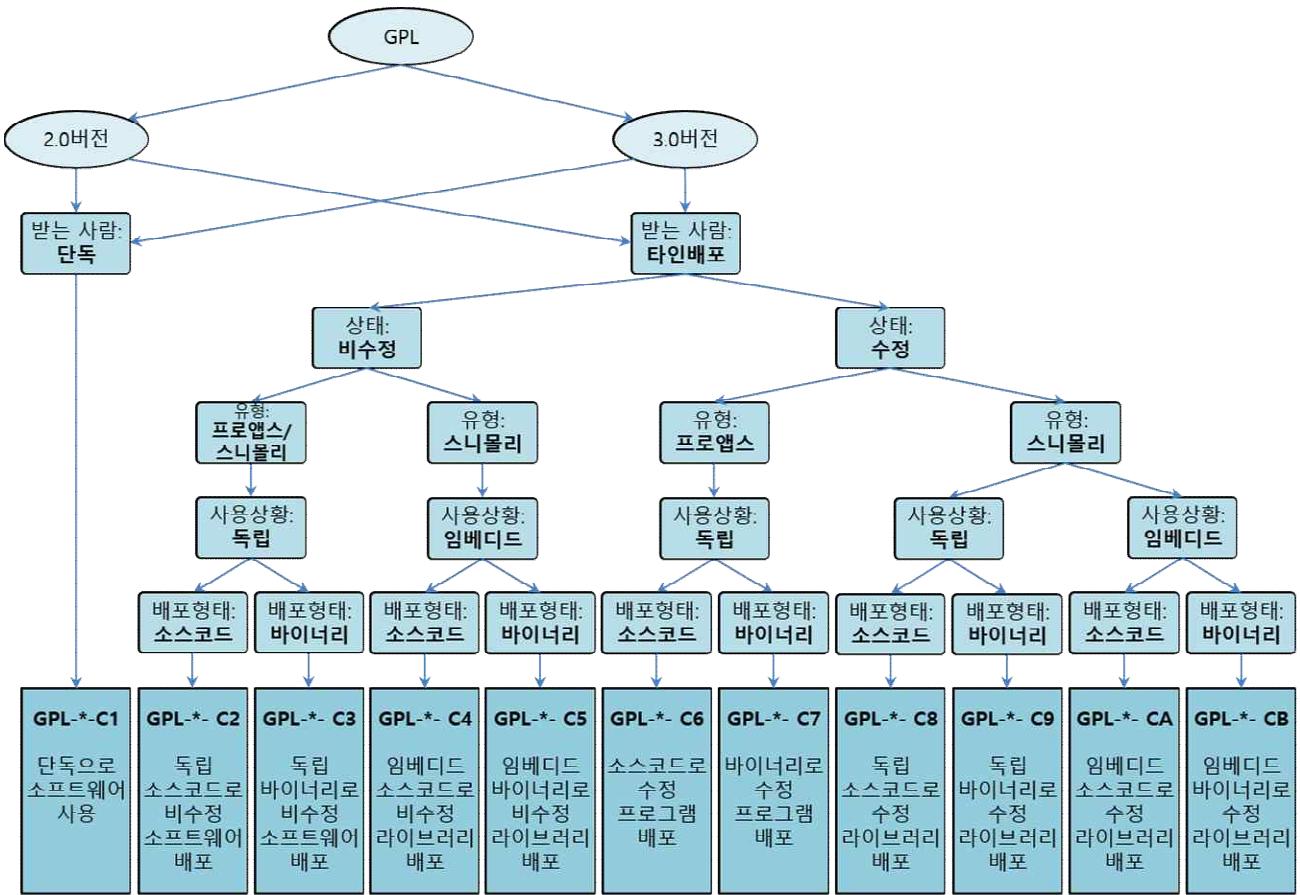
619) 위와 동일

620) 위 책과 동일, wp. §2, §4, §5. 참조

621) 위 책과 동일, wp. §1, §2, §4. 참조

622) Open Source Initiative: The GPL-3.0 License (OSI), 1991, wp. §2. 참조

623) 기본 오픈소스 사용 사례 파인더에 대한 설명은 OSLiC, pp. 88 참조



### 6.8.1 GPL-2.0-C1: 단독으로 소프트웨어 사용

**의미:** GPL-2.0로 라이선스 된 소프트웨어를 개인적으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>624)</sup>

**요구사항 :** 해당 사용 사례에 관련하여 GPL-2.0 라이선스를 준수하기 위해 지켜야할 의무사항은 없다.

- 제3자에게 소프트웨어를 제공하지 않는 한, 상황, 유형에 상관없이 GPL-2.0 소프트웨어 사용을 허가한다. 이 때, 사용에 따른 그 어떤 의무사항도 제반되지 않는다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

### 6.8.2 GPL-2.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 소스코드 파일 또는 소스코드 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

624) 자세한 내용 → OSLIC, pp.96-106 참조

**적용범위:** OSUC-02S, OSUC-05S<sup>625)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>626)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

**금지 사항:**

- 이 사용 사례와 관련하여 명확한 금지사항 없음

#### 6.8.3 GPL-2.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>627)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>628)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한

625) 자세한 내용 → OSLiC, pp.96-106 참조

626) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

627) 자세한 내용 → OSLiC, pp.96-106 참조

628) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년동안 더 온라인에서 사용할 수 있도록 한다.

- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 GPL-2.0-C2의 과업 목록을 수행한다.<sup>629)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

#### 6.8.4 GPL-2.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 소스코드 파일 또는 소스코드 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07S<sup>630)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>631)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발물의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 소스코드를 구성한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

629) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

630) 자세한 내용 → OSLiC, pp.102 참조

631) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

#### 6.8.5 GPL-2.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 바이너리 파일 또는 바이너리 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07B<sup>632)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>633)</sup>
- [필수적:] 라이브러리를 포함하는 프로그램의 소스코드 전체를 공개한다(즉 라이브러리 자체의 소스코드도 함께 공개). 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 온탑 개발물의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 바이너리를 구성한다.
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 GPL-2.0-C4의 과업 목록을 수행한다.<sup>634)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

632) 자세한 내용 → OSLiC, pp.102 참조

633) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

634) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

### 6.8.6 GPL-2.0-C6: 수정한 프로그램을 소스코드 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 소스코드 파일 형태 또는 소스코드 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04S<sup>635)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>636)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발물의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 해더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>637)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

**금지 사항:**

- 이 사용 사례와 관련하여 명확한 금지사항 없음

### 6.8.7 GPL-2.0-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** GPL-2.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 바이너리

635) 자세한 내용 → OSLiC, pp.98 참조

636) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

637) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

파일 형태 또는 바이너리 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04B<sup>638)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>639)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 프로그램의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발 물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다. 이 조건들이 충족되지 않았다면 누락된 요소를 추가한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 해더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>640)</sup>
- [필수적:] 프로그램의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-2.0-C6의 과업 목록을 수행한다.<sup>641)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

---

638) 자세한 내용 → OSLiC, pp.99 참조

639) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

640) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

641) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

#### 6.8.8 GPL-2.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미 :** GPL-2.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>642)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>643)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>644)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

642) 자세한 내용은 OSLiC, pp.103 참조

643) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

644) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

### 6.8.9 GPL-2.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미** : GPL-2.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위**: OSUC-08B<sup>645)</sup>

**요구사항**: 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>646)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-2.0-C6의 과업 목록을 수행한다.<sup>647)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>648)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

**금지 사항**:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

645) 자세한 내용은 OSLiC, pp.103 참조

646) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

647) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

648) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

### 6.8.10 GPL-2.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미** : GPL-2.0로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위**: OSUC-08B<sup>649)</sup>

**요구사항**: 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>650)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발물의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다.
- [필수적:] 임베디드 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>651)</sup>
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 소스코드를 구성한다.
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

**금지 사항**: 이 사용 사례와 관련하여 명확한 금지사항 없음

649) 자세한 내용은 OSLiC, pp.103 참조

650) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

651) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

### 6.8.11 GPL-2.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미** : GPL-2.0로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위**: OSUC-10B<sup>652)</sup>

**요구사항**: 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-2.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, GPL-2.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 GPL-2.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>653)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 라이브러리를 포함한 프로그램의 전체 소스코드를 공개한다 (즉 라이브러리 자체의 소스코드 또한 공개한다). 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-2.0-CA의 과업 목록을 수행한다.<sup>654)</sup>
- [필수적:] 온탑 개발물의 저작권 문구에 GPL-2.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 함께 제공한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>655)</sup>
- [필수적:] 기존의 GPL-2.0 라이선스가 적용될 수 있는 방식으로 바이너리를 구성한다.
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자

652) 자세한 내용은 OSLiC, pp.106 참조

653) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

654) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

655) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

신이 수정한 부분에 대한 기능적인 설명을 추가한다.

- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 이 사용 사례와 관련하여 명확한 금지사항 없음

### 6.8.12 GPL-3.0-C1: 단독으로 소프트웨어 사용

**의미:** GPL-3.0로 라이선스 된 소프트웨어를 개인적으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>656)</sup>

**요구사항 :** 해당 사용 사례에 관련하여 GPL-2.0 라이선스를 준수하기 위해 지켜야할 의무사항은 없다.

- 제3자에게 소프트웨어를 제공하지 않는 한, 상황, 유형에 상관없이 GPL 소프트웨어 사용을 허가 한다. 이 때, 사용에 따른 그 어떤 의무사항도 제반되지 않는다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.8.13 GPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 소스코드 파일 또는 소스코드 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S<sup>657)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>658)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.

656) 자세한 내용 → OSLiC, pp.96-106 참조

657) 자세한 내용 → OSLiC, pp.96-106 참조

658) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

**금지 사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.8.14 GPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>659)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>660)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 GPL-3.0-C2의 과업 목록을 수행한다.<sup>661)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

**금지 사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

---

659) 자세한 내용 → OSLiC, pp.96-106 참조

660) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

661) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

### 6.8.15 GPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 소스코드 파일 또는 소스코드 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07S<sup>662)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>663)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발의 저작권 문구에 GPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 소스코드를 구성한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

**금지 사항:**

- 소프트웨어가 특허 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

### 6.8.16 GPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 바이너리 파일 또는 바이너리 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07B<sup>664)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로

662) 자세한 내용 → OSLiC, pp.102 참조

663) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

664) 자세한 내용 → OSLiC, pp.102 참조

로 패키지에 포함한다.

- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>665)</sup>
- [필수적:] 라이브러리를 포함하는 프로그램의 소스코드 전체를 공개한다 (즉 라이브러리 자체의 소스코드도 함께 공개). 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 온탑 개발의 저작권 문서에 GPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 바이너리를 구성한다.
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 GPL-3.0-C4의 과업 목록을 수행한다.<sup>666)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.8.17 GPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 소스코드 파일 형태 또는 소스코드 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04S<sup>667)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>668)</sup>

665) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

666) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

667) 자세한 내용 → OSLiC, pp.98 참조

- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램의 저작권 문구에 GPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다. 누락된 요소가 있는 경우 이를 추가한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 프로그램에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>669)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.8.18 GPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** GPL-3.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 바이너리 파일 형태 또는 바이너리 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04B<sup>670)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>671)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 온탑 개발의 저작권 문구에 GPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-2.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소

668) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

669) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

670) 자세한 내용 → OSLiC, pp.99 참조

671) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다.

- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>672)</sup>
- [필수적:] 프로그램의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-3.0-C6의 과업 목록을 수행한다.<sup>673)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특허 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

#### 6.8.19 GPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미 :** GPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>674)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>675)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.

672) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

673) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

674) 자세한 내용은 OSLiC, pp.103 참조

675) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-2.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>676)</sup>
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.8.20 GPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미 :** GPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08B<sup>677)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>678)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-3.0-C8의 과업 목록을 수행한다.<sup>679)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다.

676) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

677) 자세한 내용은 OSLiC, pp.103 참조

678) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

679) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>680)</sup>

- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.8.21 GPL-3.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미 :** GPL-3.0로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-10S<sup>681)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>682)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발의 저작권 문구에 GPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>683)</sup>
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 소스코드를 구성한다.
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자

680) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

681) 자세한 내용은 OSLiC, pp.105 참조

682) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

683) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

신이 수정한 부분에 대한 기능적인 설명을 추가한다.

- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-2.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.8.22 GPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미** : GPL-3.0로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위**: OSUC-10B<sup>684)</sup>

**요구사항**: 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 GPL-3.0와 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문을 포함한다. 이 요소가 누락되었다면, 주요 저작권 고지문을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 GPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>685)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 라이브러리를 포함한 프로그램의 전체 소스코드를 공개한다 (즉 라이브러리 자체의 소스코드 또한 공개한다). 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 GPL-3.0-CA의 과업 목록을 수행한다.<sup>686)</sup>
- [필수적:] 온탑 개발의 저작권 문구에 GPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 GPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을

684) 자세한 내용은 OSLiC, pp.106 참조

685) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

686) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 GPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>687)</sup>

- [필수적:] 기존의 GPL-3.0 라이선스가 적용될 수 있는 방식으로 바이너리를 구성한다.
- [자발적:] 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, GPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.8.23 논의 및 설명

GPL-2.0에서는 “받은 프로그램의 전체 소스코드의 verbatim copy를 복사하고 배포할 수 있는데, 단 (a) 적합한 저작권 고지 및 보증 제한 사본을 각각 적절히 눈에 띄게 공표하고; (b) 이 라이선스와 보증 제한 고지문을 인용한 모든 고지를 그대로 두고; (c) 그리고 프로그램과 함께 이 라이선스 사본을 배포한다는 조건으로 허용한다.”<sup>688)</sup> 또한 이 라이선스에서는 “제 1조항에 의거 프로그램 또는 그 일부분을 수정한 저작물의 복사 및 배포를 허용하고”<sup>689)</sup> 바이너리는 “제 1조항과 2조항에 의거 배포를 허용”한다.<sup>690)</sup> 그러나 저작물을 개인적인 용도로 사용할 경우 라이선스 준수를 위한 그 어떠한 작업도 요구하지 않는다. 따라서 제 1조항의 조건들은 GPL-2.0으로 라이선스된 저작물을 배포하는 경우에만 의무적으로 지켜져야 한다. (GPL-C2 ~ GPL-CB).

GPL-3.0에서는 동일한 요구사항으로 구성된 내용과 유사한 구조를 사용한다. §4에서는 “받은 프로그램의 소스코드의 verbatim copy를 복사하고 배포할 수 있는데, 단 적합한 저작권 고지 및 보증 제한 사본을 각각 적절히 눈에 띄게 공표하고; 섹션 7에 따라 추가된 비허용적 조항과 라이선스는 코드에 적용된다는 점이 명시된 고지를 그대로 두고; 보증 제한에 대한 고지문을 그대로 유지하고; 그리고 프로그램과 함께 이 라이선스 사본을 배포한다는 조건으로 허용한다. §5에서는 또한 “제 4조항에 의거 수정한 저작물을 전달할 수 있으며, §6에서는 “제 4조항 및 제 5조항에 의거 오브젝트 형태로 저작물을 전달할 수 있다.”<sup>691)</sup> GPL-2.0와 반대로, GPL-3.0에서는 “(GPL-3.0) 라이선스 효력이 지속되고 저작물을 전달[배포]하지 않는다는 조건 하에 (사용자는) 저작물을 작성, 실행 및 유포할 수 있음을” 명백히 밝히고 있다.<sup>692)</sup> 또한 구매자를 위해 대신해서 파생저작물을 실행하였고 수정 버전을 어떠한 제3자에게 배포하지 않는다면 수정 버전을 다시 확보하기 위해 제3자에게 제공하는 행위를 배포 사례로 간주하지 않는다.<sup>693)</sup> GPL-3.0 조항을 전부 종합해 본다면 GPL-2.0 의무사항을 충족하기 위한 일련의 작업들이 GPL-3.0 조건에도 적합하다는 결론을 내릴 수 있다.

687) 자세한 내용은 GPL-2.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

688) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp §1 참조

689) 인용한 글과 동일, wp. §2. 참조

690) 인용한 글과 동일, wp. §4. 참조

691) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp §4, §5, §6 참조

692) 위와 동일, wp. §2 참조

693) 위와 동일

GPL-2.0에서는 “소프트웨어 교환에 통상적으로 사용되는 매체를 통해 기계 판독이 가능한 전체 소스코드 사본을 첨부한다는 조건 하에 오브젝트 코드 또는 실행 가능한 형태로 프로그램 (또는 파생저작물)의 복사 및 배포”를 허용한다.<sup>694)</sup> 이 기본 조건에 대한 대안으로 GPL-2.0은 바이너리 배포 패키지에 “최소 3년 동안 유효한 written offer를 첨부하여 제3자에 전달”하는 것을 허용한다.<sup>695)</sup> 이 때, “소프트웨어 교환에 통상적으로 사용되는 매체를 통해 기계 판독이 가능한 전체 소스코드 사본을 첨부하는 방식보다 비용이 높으면 안 된다.” OSLiC에서는 현재 소프트웨어 교환에 사용되는 매체에 배포물로서 인터넷을 통해 파일을 다운로드하는 일반 기법을 설명한다. 이에 따라 OSLiC에서는 바이너리 배포를 포함하는 오픈소스 사용 사례(GPL-2.0-C3, GPL-2.0-C7, GPL-2.0-C9, GPL-2.0-CB)에 대해 배포자가 제어 가능한 인터넷 저장소를 통해 라이브러리 소스코드 자체를 공개할 것을 요구한다.

또한 GPL-3.0에서도 바이너리 배포의 경우 소스코드를 공개할 것을 명시하고 있다. 그러나 GPL-2.0과 반대로, GPL-3.0에서는 “무료로 네트워크 서버에 접근하여 해당 소스코드를 복사할 수 있는 권한을” 주어도 된다는 옵션을 제공한다.<sup>696)</sup> 따라서 바이너리 배포의 경우에도 GPL-2.0 의무사항을 충족하기 위한 일련의 작업들이 GPL-3.0 조건에도 적합하다는 결론을 내릴 수 있다.

이럴 경우 “[바이너리를 받은] 제3자가 소스코드를 반드시 복사하지 않아도 되는” 약점이 발생하나 GPL-2.0은 이를 간접적으로 허용한다.<sup>697)</sup> 그러나 제어 가능한 저장소를 통해 소스코드를 제공해야하는 의무사항은 통상적으로 허용되지 않는다. GPL-2.0은 “비상업적 배포”의 경우에만 외부 소스코드 저장소 링크를 통한 재배포를 허용한다.<sup>698)</sup>

GPL-2.0과 GPL-3.0은 다음과 같은 매우 유사한 제한 및 조건 하에서만 “프로그램이나 프로그램 일부의 사본의 복사 및 배포, 사본의 수정, 그리고 그 수정물의 복사 및 배포”를 허용한다.<sup>699)</sup>

- 첫째, 수정된 파일에서 수정된 부분 및 수정된 날짜를 표시해야 한다.<sup>700)</sup> 이 조건은 수정된 저작물 배포와 관계된 모든 오픈소스 사용 사례에서 준수되어야한다 (GPL-2.0-C6/GPL-3.0-C6 – GPL-2.0-C9/GPL-3.0-C9). 왜냐하면 주로 바이너리를 배포할 의향이더라도 나중에 소스코드 또한 제공해야하기 때문이다. OSLiC에서는 각 수정 파일을 표시하는 의무적 조건과 changelog를 업데이트/생성하는 자발적 조건으로 이 요구사항을 다시 고쳐 쓰고 있다.
- 둘째, GPL에서는 메시지 표시 수단 또는 대화형 인터페이스를 사용하는 수정된 소프트웨어 사본의 경우 “적합한 저작권 및 보증 제한 고지를 제공하고, 사용자가 이 조건 하에 프로그램을 재배포 할 수 있다는 고지를 포함하고, 사용자에게 이 라이선스 사본을 열람할 수 있는 방법을 알려주는

694) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §3, §3a. 참조

695) 인용한 글과 동일, wp. §3b. 참조

696) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §6, §6b 참조.

697) Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §3, 말미. 참조

698) 인용한 글과 동일, wp. §3c. 참조

699) 인용한 글과 동일, wp. §2. 참조

700) GPL-2.0에 대해서는 위 책과 동일 참조 참조.

GPL-3.0에 대해서는 Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §5d를 참조.

안내를 인쇄하거나 표시할 것을” 요구한다.<sup>701)</sup> OSLiC에서는 저작물에 관해서는 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 그 홈페이지 링크, 보증 제한 문구 그리고 소프트웨어와 함께 제공해야 하는 GPL-2.0 파일 (혹은 GPL-3.0 파일) 링크를 저작권 다이얼로그를 통해 복제하는 형태로 이 조건을 다시 고쳐 쓰고 있다. 이 조건은 프로그램을 받아 수정한 후 재배포하거나(GPL-2.0-C6, GPL-2.0-C7, GPL-3.0-C6, GPL-3.0-C7) (수정된) 라이브러리를 임베디드 컴포넌트로 사용하는 자체 소유 프로그램을 배포하게 되면 (GPL-2.0-CA, GPL-2.0-CB, GPL-3.0-CA, GPL-3.0-CB) 지켜져야 한다. 라이브러리를 받아서 수정한 후 재배포한 경우에 대해서는 OSLiC에서 언급하지 않는다. 라이브러리, 플러그인 또는 스니펫은 일반적으로 자체 저작권 다이얼로그를 제공하기 않기 때문이다.

- 셋째, GPL에서는 “(저작물을 배포/공표할 시) 프로그램 전체 또는 그 일부를 포함하거나, 프로그램으로부터 파생된 저작물에 대해서는 GPL 조항에 기반하여 (제3자에게 무료로) 전체가 라이선싱”되어야 한다고 요구한다.<sup>702)</sup> 이 요구사항은 소프트웨어가 배포되는 형태에 의해 좌우되지 않는다. OSLiC에서는 다음과 같이 이 설명을 채택하였다.
  - 배포와 관계된 모든 오픈소스 사용 사례의 경우(GPL-2.0-C2 ... GPL-2.0-CB, GPL-3.0-C2 ... GPL-3.0-CB), OSLiC에서는 이 조항을 기존의 라이선싱 요소들을 모두 유지해야한다는 필수적 요구사항으로 개정하였다.
  - 소프트웨어의 수정 버전을 배포하는 행위를 다루는 모든 사용 사례의 경우 (GPL-2.0-C6 ... GPL-2.0-CB, GPL-3.0-C6 ... GPL-3.0-CB), OSLiC에서는 GPL-2.0 (혹은 GPL-3.0) 라이선스가 적용되는 방식으로 수정사항을 구성해야 한다는 요구사항을 추가하였다.
  - 임베디드 라이브러리를 배포하는 행위를 다루는 사용 사례의 경우 (GPL-2.0-C4, GPL-2.0-C5, GPL-2.0-CA, GPL-2.0-CB, GPL-3.0-C4, GPL-3.0-C5, GPL-3.0-CA, GPL-3.0-CB), OSLiC에서는 온탑 개발에 대해서도 GPL-2.0 (혹은 GPL-3.0)으로 라인선싱 되어야 한다고 요구한다.
- 마지막으로, 바이너리 형태의 배포와 관련되는 작업 리스트의 일부로서, OSLiC에서는 일치하는 소스를 접근 불가능하게 하면서 바이너리를 배포하는 행위는 GPL에 의해 허용되지 않으므로 일치하는 소스코드 사용 사례를 실행할 것을 독자에게 상기시킨다.

마지막으로 세밀성에 관한 문제를 다룬다.

GPL-3.0에서는 “상응하는 소스코드 (corresponding source)를 전달할 경우 저작물을 오브젝트 코드 형태로 전달”하는 것을 허용한다.<sup>703)</sup> 소스코드의 범위를 이해하기 위해서, “상응하는 소스코드”가 의미하는 바를 알아야한다. 다행이도, GPL-3.0은 이 단어를 올바르게 이해할 수 있도록 단서를 제공한다.

- “오브젝트 코드 형태의 저작물의 ‘상응하는 소스코드’란, 오브젝트 코드를 생성, 설치, 그리고 (실

701) GPL-2.0에 대해서는 Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §2c를 참조.  
GPL-3.0에 대해서는 Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §5d를 참조.

702) GPL-2.0에 대해서는 Open Source Initiative: The GPL-2.0 License (OSI), 1991, wp. §2b를 참조.  
GPL-3.0에 대해서는 Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §5c를 참조.

703) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §6 참조.

행 파일의 경우) 실행하는데에 필요한 모든 소스코드를 의미한다. 이러한 작업들을 제어하기 위한 스크립트도 포함된다.”<sup>704)</sup>

- 범위를 제한하기 위해 GPL에서는 ‘상응하는 소스코드’에 저작물의 시스템 라이브러리, 일반적인 도구들, 일반적으로 수정하지 않은 채로 사용가능한 자유 프로그램”은 포함되지 않는다고 정의한다.<sup>705)</sup> 이 정의를 명확하게 이해하기 위해서는, 시스템 라이브러리라는 단어를 해석해야 한다.
- 따라서 GPL에서는 실행 파일의 ‘시스템 라이브러리’란, 다음 조건들을 만족하는 것이라고 정의한다: (a) 주요 컴포넌트 (major component) 를 패키징하는 일반적인 형태에 포함되나, 주요 컴포넌트의 일부는 아니며, (b) 주요 컴포넌트와 함께 저작물을 활성화하는 용도, 또는 대중이 사용가능한 소스코드 형태의 구현을 위해 표준 인터페이스를 구현하는 용도로 사용된다.<sup>706)</sup> 이 문장을 제대로 이해하기 위해서는, ‘주요 컴포넌트’가 의미하는 바를 이해해야 한다.
- GPL은 ‘주요 컴포넌트’를 다음과 같이 정의한다: “특정 운영체제의 주요 필수 컴포넌트(커널, 윈도우 시스템 등)로서, 이 운영체제에서는 실행 가능한 작업 실행, 컴파일러의 저작물 생성 또는, 저작물을 실행하는 오브젝트 코드 인터프리터가 동작할 수 있다.”<sup>707)</sup>

이 규격에 따라, 배포되는 GPL 바이너리 프로그램에 상응하는 소스코드의 어느 정도 레벨까지 접근 권한을 부여해야 할지에 대한 기준을 마련할 수 있다.

- GPL로 라이선스된 바이너리 프로그램을 배포할 경우, 이와 함께 다음 요소들의 소스코드도 배포해야 한다.
  - 배포 프로그램
  - 수정된 모든 임베디드 컴포넌트
  - 자유로운 접근이 불가능한 모든 임베디드 컴포넌트
  - 프로그램의 소스코드를 자유롭게 컴파일/개발 가능하게 하기 위한 모든 툴, 스크립트, 데이터 그러나 자체 기본 저장소로부터 자유롭게 다운로드 가능한 수정하지 않은 스탠더드 라이브러리, 컴파일러, 툴에 대해서는 소스코드를 제공할 필요가 없다.
- GPL로 라이선스된 스크립트를 배포할 경우, 이와 함께 다음 요소들의 소스코드도 배포해야 한다.
  - 메인 스크립트에 의해 추가된 수정된 모든 임베디드 스크립트 컴포넌트
  - 메인 스크립트에 의해 추가된 자유로운 접근이 불가능한 모든 임베디드 스크립트 컴포넌트
  - 자유롭게 접근이 가능한 인터프리터에 의해 메인 스크립트가 실행 가능하게 하기 위한 모든 툴, 스크립트, 데이터
  - 자유롭게 접근이 불가능할 경우, 인터프리터 자체그러나 자체 기본 저장소로부터 자유롭게 다운로드 가능한 수정하지 않은 스탠더드 스크립트 라이브러리, 인터프리터, 툴에 대해서는 소스코드를 제공할 필요가 없다.

---

704) 인용한 글과 동일, wp. §1 참조

705) 위와 동일

706) 위와 동일

707) 위와 동일

## 6.9 LGPL 라이선스 소프트웨어

GNU Lesser General Public License 두 버전은 소스코드 배포와 바이너리 배포를 명확히 구분 짓고 있다. LGPL-2.1에서는 주로 소스코드 복사 및 배포에 관한 내용을 다룬다.<sup>708)</sup> 또한 “오브젝트 코드 또는 실행 가능한 형태로 라이브러리를 복사 및 배포”하기 위한 특정 조건을 직접적으로 언급하고 있다.<sup>709)</sup> 반면에, LGPL-3.0과 GPL-3.0(GPL-3.0은 LGPL-3.0에 포함되므로 함께 고려해야 함<sup>710)</sup>)은 소스코드 배포와 오브젝트 코드 배포를 동일한 현상이지만 다른 측면으로 다루고 있다.<sup>711)</sup> 또한 LGPL-2.1과 LGPL-3.0에서는 주로 소프트웨어 복제 및 배포에 대해 설명한다(개인적인 용도의 사용은 거의 구체적으로 명시되지 않았다.<sup>712)</sup>) 마지막으로 LGPL-2.1과 LGPL-3.0에서는 거의 동일한 라이선스에 따른 작업을 요구하여 동일한 결과와 동일한 취지를 목표로 두고 있다. 이에 따라, 하나의 장에서 두 가지 버전을 다루고<sup>713)</sup> 쉽게 처리 가능한 해당 작업 리스트를 찾기 위해 동일한 LGPL 특정 오픈소스 사용 사례 구조를 제공하는 것이다.<sup>714)</sup> 가장 적절하다.

---

708) Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp §1, §2, §5, §6. 참조

709) 인용한 글과 동일, wp. §4. 참조

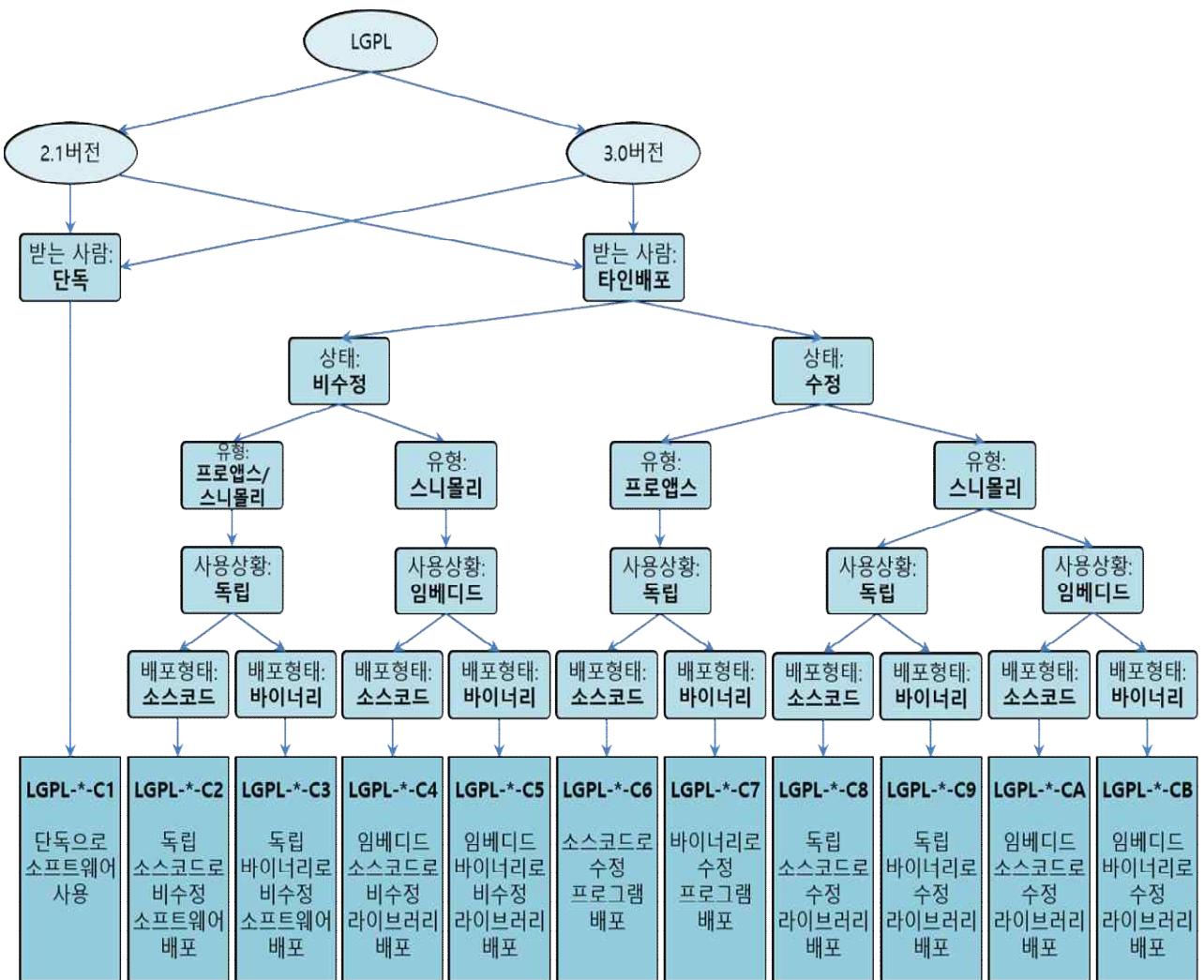
710) Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp §0 바로 앞. 참조

711) GPL-3.0에는 “오브젝트 코드 형태의 저작물을 전달하는” 조건을 설명하는 “비소스 형태 전달(Conveying Non-Source Form)” 섹션을 포함하고 있다 (Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §6 참조))., LGPL-3.0에서는 “라이브러리 헤더 파일의 자료를 포함하는 오브젝트 코드를” 명확히 다루고 있다”(Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp. §3 참조)).

712) LGPL-2.1에서는 “라이브러리 카피” 복사 및 배포 행위와 관해서만 ‘제한 사항’을 열거한다(Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. §1, §2, §4 참조, 외 비슷한 내용 있음). GPL-3.0에서는 “라이선스가 지속적으로 효력을 갖는 조건 하에 배포하지 않는 저작물을 작성, 실행 및 유포할 수 있음을” 명시하고 있다(Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §2 참조).

713) 예외사항은 LGPL 조항에 의거 수정 프로그램, 애플리케이션 또는 서버를 배포하는 행위와 관계된다.

714) 일반적 OSUC 파인더에 관한 자세한 내용 → OSLiC, pp. 104, ??



### 6.9.1 LGPL-2.1-C1: 단독으로 소프트웨어 사용

**의미:** LGPL-2.1로 라이선스 된 소프트웨어를 개인적으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>715)</sup>

**요구사항 :** 해당 사용 사례에 관련하여 LGPL-2.1 라이선스를 준수하기 위해 지켜야할 의무사항은 없다.

- 제3자에게 소프트웨어를 제공하지 않는 한, 상황, 유형에 상관없이 LGPL-2.1 소프트웨어 사용을 허가한다. 이 때, 사용에 따른 그 어떤 의무사항도 제반되지 않는다.

### 금지 사항:

- 명시된 사항 없음

715) 자세한 내용 → OSLIC, pp.96-105 참조

### **6.9.2 LGPL-2.1-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포**

**의미:** LGPL-2.1로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 소스코드 파일 또는 소스코드 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S<sup>716)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>717)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.

**금지 사항:**

- 명시된 사항 없음

### **6.9.3 LGPL-2.1-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포**

**의미:** LGPL-2.1로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>718)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제

716) 자세한 내용 → OSLiC, pp.96-99 참조

717) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

718) 자세한 내용 → OSLiC, pp.97-100 참조

한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)

- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>719)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 LGPL-2.1-C2의 과업 목록을 수행한다.<sup>720)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.

#### 금지 사항:

- 명시된 사항 없음

#### 6.9.4 LGPL-2.1-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** LGPL-2.1로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 소스코드 파일 또는 소스코드 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07S<sup>721)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>722)</sup>

719) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

720) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

721) 자세한 내용 → OSLiC, pp.107 참조

722) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.102

- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

**금지 사항:** 명시된 사항 없음

#### 6.9.5 LGPL-2.1-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** LGPL-2.1로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 바이너리 파일 또는 바이너리 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07B<sup>723)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>724)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용자가 직접 애플리케이션을 링킹할 수 있도록 온탑 개발 및 라이브러리의 모든 오브젝트 파일을 제공하기 위해 온탑 개발 또는 라이브러리를 동적 링킹 가능한 형태로 배포하거나 정적 링킹된 애플리케이션을 최소 3년 동안 유효한 written offer와 함께 배포한다.
- [필수적:] 사용 사례 LGPL-2.1-C4의 과업 목록을 수행한다.<sup>725)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

723) 자세한 내용 → OSLiC, pp.102 참조

724) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

725) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

#### 금지 사항:

- 명시된 사항 없음

#### 6.9.6 LGPL-2.1-C6: 수정한 프로그램을 소스코드 형태로 배포

**의미:** LGPL-2.1로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapce)를 수정하였고 이를 소스코드 파일 형태 또는 소스코드 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04S<sup>726)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] LGPL-2.1을 참조하는 모든 파일의 고지문이 이 라이선스가 아닌 GNU General Public License, version 2를 참조하도록 변경한다.

#### 금지 사항:

- 저작물을 수정하여 “수정된 저작물”이 더 이상 소프트웨어 라이브러리가 아니게 하는 것 (즉 결과물이 라이브러리가 아닌 프로그램이 되는 것). **수정된 프로그램을 LGPL-2.1 라이선스로 배포하는 것을 금한다.**<sup>727)</sup>

#### 6.9.7 LGPL-2.1-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** LGPL-2.1로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapce)를 수정하였고 이를 바이너리 파일 형태 또는 바이너리 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04B<sup>728)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] LGPL-2.1을 참조하는 모든 파일의 고지문이 이 라이선스가 아닌 GNU General Public License, version 2를 참조하도록 변경한다.

#### 금지 사항:

- 저작물을 수정하여 “수정된 저작물”이 더 이상 소프트웨어 라이브러리가 아니게 하는 것 (즉 결과물이 라이브러리가 아닌 프로그램이 되는 것). **수정된 프로그램을 LGPL-2.1 라이선스로 배포하는 것을 금한다.**<sup>729)</sup>

726) 자세한 내용 → OSLiC, pp.98 참조

727) LGPL-2.1는 “수정된 저작물이 소프트웨어 라이브러리여야 한다”고 명시한다. (Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. x2a 참조). 자세한 내용은 → OSLiC, p.217 참조

728) 자세한 내용 → OSLiC, pp.99 참조

729) LGPL-2.1는 “수정된 저작물이 소프트웨어 라이브러리여야 한다”고 명시한다. (Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. x2a 참조). 자세한 내용은 → OSLiC, p. 217 참조

### 6.9.8 LGPL-2.1-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** LGPL-2.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>730)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>731)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-2.1 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-2.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>732)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

**금지 사항:**

- 라이브러리를 수정하여 더 이상 라이브러리가 아니게 만드는 것

### 6.9.9 LGPL-2.1-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미 :** LGPL-2.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

730) 자세한 내용은 OSLiC, pp. 103 참조

731) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

732) 자세한 내용은 LGPL-2.1의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

**적용범위:** OSUC-08B<sup>733)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>734)</sup>
- [필수적:] 배포 소프트웨어의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 패키지 안에 어디서 어떻게 코드를 수령할 수 있는지에 대한 설명을 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 LGPL-2.1-C8의 과업 목록을 수행한다.<sup>735)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-2.1 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-2.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>736)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

**금지 사항:**

- 라이브러리를 수정하여 더 이상 라이브러리가 아니게 만드는 것

#### 6.9.10 LGPL-2.1-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미 :** LGPL-2.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈,

733) 자세한 내용은 OSLiC, pp.103 참조

734) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

735) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

736) 자세한 내용은 LGPL-2.1의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-10S<sup>737)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>738)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-2.1 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-2.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>739)</sup>
- [필수적:] 라이브러리의 구조적 독립을 유지한다.
- [필수적:] 온탑 개발의 저작권 문구에 LGPL-2.1 라이선스 라이브러리를 사용하고 있다는 것과 개발 물 또한 LGPL-2.1로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 함께 제공한다.
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

**금지 사항:**

- 라이브러리를 수정하여 더 이상 라이브러리가 아니게 만드는 것

#### 6.9.11 LGPL-2.1-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미 :** LGPL-2.1로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈,

737) 자세한 내용은 OSLiC, pp.105 참조

738) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

739) 자세한 내용은 LGPL-2.1의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-10B<sup>740)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-2.1과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-2.1에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다. (고지문이 라이선스에 이미 포함되어 있어 반복적인 절차가 되더라도 이를 이행한다.)
- [필수적:] 받는 사람에게 LGPL-2.1 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>741)</sup>
- [필수적:] 임베디드 라이브러리의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 패키지 안에 어디서 어떻게 코드를 수령할 수 있는지에 대한 설명을 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 LGPL-2.1-CA의 과업 목록을 수행한다.<sup>742)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-2.1 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-2.1을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>743)</sup>
- [필수적:] 라이브러리의 구조적 독립을 유지한다.
- [필수적:] 온탑 개발의 저작권 문구에 LGPL-2.1 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 LGPL-2.1로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 함께 제공한다.
- [필수적:] 사용자가 직접 애플리케이션을 링킹할 수 있도록 온탑 개발 및 라이브러리의 모든 오브젝트 파일을 제공하기 위해 온탑 개발 또는 라이브러리를 동적 링킹 가능한 형태로 배포하거나 정적 링킹된 애플리케이션을 최소 3년 동안 유효한 written offer와 함께 배포한다.
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에

740) 자세한 내용은 OSLiC, pp.106 참조

741) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

742) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

743) 자세한 내용은 LGPL-2.1의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.

- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

#### 금지 사항:

- 라이브러리를 수정하여 더 이상 라이브러리가 아니게 만드는 것

### 6.9.12 LGPL-3.0-C1: 단독으로 소프트웨어 사용

**의미:** LGPL-3.0으로 라이선스 된 소프트웨어를 개인적으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>744)</sup>

**요구사항:** 해당 사용 사례에 관련하여 LGPL-3.0 라이선스를 준수하기 위해 지켜야할 의무사항은 없다.

제3자에게 소프트웨어를 제공하지 않는 한, 상황, 유형에 상관없이 LGPL-3.0 소프트웨어 사용을 허가한다. 이 때, 사용에 따른 그 어떤 의무사항도 제반되지 않는다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.9.13 LGPL-3.0-C2: 수정하지 않은 소프트웨어를 독립적인 소스코드 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 소스코드 파일 또는 소스코드 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02S, OSUC-05S<sup>745)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>746)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에

744) 자세한 내용 → OSLiC, pp.96-105 참조

745) 자세한 내용 → OSLiC, pp.96-100 참조

746) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.9.14 LGPL-3.0-C3: 수정하지 않은 소프트웨어를 독립적인 바이너리 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 소프트웨어를 수정하지 않은 상태로 단독 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포하고자 하는 경우를 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리 또는 플러그인 등 배포유형은 상관이 없다.

**적용범위:** OSUC-02B, OSUC-05B<sup>747)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>748)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 사용 사례 LGPL-3.0-C2의 과업 목록을 수행한다.<sup>749)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

747) 자세한 내용 → OSLiC, pp.97-100 참조

748) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

749) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

### 6.9.15 LGPL-3.0-C4: 수정하지 않은 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 소스코드 파일 또는 소스코드 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07S<sup>750)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>751)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 온탑 개발의 저작권 문서에 LGPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 LGPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 함께 제공한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

**금지 사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

### 6.9.16 LGPL-3.0-C5: 수정하지 않은 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 스니펫, 모듈 또는 라이브러리를 받아 수정하지 않은 상태의 바이너리 파일 또는 바이너리 패키지 형태로 더 큰 유닛의 임베디드 컴포넌트로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-07B<sup>752)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

750) 자세한 내용 → OSLiC, pp.102 참조

751) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

752) 자세한 내용 → OSLiC, pp.102 참조

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>753)</sup>
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 수정하지 않았더라도 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 온탑 개발의 저작권 문구에 LGPL-3.0 라이선스 라이브러리를 사용하고 있다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 사용자가 직접 애플리케이션을 링킹할 수 있도록 온탑 개발 및 라이브러리의 모든 오브젝트 파일을 제공하기 위해 온탑 개발 또는 라이브러리를 동적 링킹 가능한 형태로 배포하거나 정적 링킹된 애플리케이션을 최소 3년 동안 유효한 written offer와 함께 배포한다.
- [필수적:] 사용 사례 LGPL-3.0-C4의 과업 목록을 수행한다.<sup>754)</sup>
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-2.1 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

#### **금지 사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

#### **6.9.17 LGPL-3.0-C6: 수정한 프로그램을 소스코드 형태로 배포**

**의미:** LGPL-3.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapce)를 수정하였고 이를 소스코드 파일 형태 또는 소스코드 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04S<sup>755)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.

753) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

754) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

755) 자세한 내용 → OSLiC, pp.98 참조

- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>756)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램의 저작권 문구에 LGPL-3.0 라이선스 라이브러리를 사용하고 있다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 함께 제공한다. 누락된 요소가 있으면 이를 추가한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>757)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.9.18 LGPL-3.0-C7: 수정한 프로그램을 바이너리 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 프로그램, 애플리케이션 또는 서버(proapse)를 수정하였고 이를 바이너리 파일 형태 또는 바이너리 패키지로 제3자에게 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-04B<sup>758)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에

756) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

757) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

758) 자세한 내용 → OSLiC, pp.99 참조

사본이 빠져있다면 이를 추가한다.<sup>759)</sup>

- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 프로그램(proapse) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>760)</sup>
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 수정사항을 구성한다. 만약 새로운 소스코드 파일을 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.
- [필수적:] 자신이 배포한 소프트웨어의 소스코드는 자신이 제어 가능한 저장소(repository)를 통해 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 LGPL-3.0-C4의 과업 목록을 수행한다.<sup>761)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.9.19 LGPL-3.0-C8: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** LGPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>762)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.

759) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

760) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

761) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

762) 자세한 내용은 OSLiC, pp.103 참조

- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>763)</sup>
- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>764)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.9.20 LGPL-3.0-C9: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미 :** LGPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 배포하려는 의도를 갖고 있다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08B<sup>765)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>766)</sup>
- [필수적:] 배포 소프트웨어의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고

763) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

764) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

765) 자세한 내용은 OSLiC, pp.103 참조

766) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.

- [필수적:] 기존의 모든 저작권 고지문을 유지한다.
- [필수적:] 배포 패키지 안에 어디서 어떻게 코드를 수령할 수 있는지에 대한 설명을 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 LGPL-3.0-C8의 과업 목록을 수행한다.<sup>767)</sup>
- [필수적:] 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>768)</sup>
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

#### 6.9.21 LGPL-3.0-CA: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미 :** LGPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 소스코드 파일 또는 소스코드 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-10S<sup>769)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 소스코드 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>770)</sup>

767) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

768) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

769) 자세한 내용은 OSLiC, pp.105 참조

- [필수적:] 온탑 개발의 저작권 문구에 LGPL-3.0 라이선스 라이브러리를 사용하고 있다는 것과 개발물 또한 LGPL-3.0으로 라이선싱 되었다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 임베디드 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>771)</sup>
- [필수적:] 라이브러리의 구조적 독립을 유지한다.
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

#### **금지 사항:**

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특허 소송을 제기

#### **6.9.22 LGPL-3.0-CB: 수정한 라이브러리를 임베디드 바이너리 형태로 배포**

**의미 :** LGPL-3.0으로 라이선스 된 코드 스니펫, 모듈, 라이브러리 또는 플러그인(snimoli)를 수정하였고 이를 바이너리 파일 또는 바이너리 패키지 형태로 제3자에게 임베디드 컴포넌트로서 스니펫, 모듈, 라이브러리 또는 플러그인을 포함하는 다른 더 큰 소프트웨어 유닛과 함께 배포하려는 의도를 갖고 있다.

**적용범위:** OSUC-10B<sup>772)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소, 특히 LGPL-3.0과 보증 제한과 연관된 고지문은 받았던 형태로 그대로 패키지에 포함한다.
- [필수적:] 배포된 바이너리 패키지에는 명확하고 쉽게 찾아볼 수 있는 저작권 고지문과 보증 제한을 포함한다. 이 요소들이 누락되었다면, LGPL-3.0에 의해 정의된 형태로 주요 저작권 고지문과 보증 제한을 포함하는 파일을 새로 추가한다.
- [필수적:] 받는 사람에게 LGPL-3.0 라이선스 사본을 제공한다. 이미 구성한 소프트웨어 패키지에 사본이 빠져있다면 이를 추가한다.<sup>773)</sup>

770) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

771) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

772) 자세한 내용은 OSLiC, pp.106 참조

773) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

- [필수적:] 임베디드 라이브러리의 소스코드를 공개한다. 인터넷 저장소에 소스코드 패키지를 올리고 인터넷을 통한 다운로드를 허용한다. 이 저장소는 소프트웨어 패키지 배포를 그만둔 후 최소 3년 동안 더 온라인에서 사용할 수 있도록 한다.
- [필수적:] 배포 패키지 안에 어디서 어떻게 코드를 수령할 수 있는지에 대한 설명을 눈에 띄게 삽입한다.
- [필수적:] 사용 사례 LGPL-3.0-CA의 과업 목록을 수행한다.<sup>774)</sup>
- [필수적:] 온탑 개발의 저작권 문구에 LGPL-3.0 라이선스 라이브러리를 사용하고 있다는 것을 명확히 표시한다. 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 함께 제공한다.
- [필수적:] 임베디드 라이브러리(snimoli) 소스코드의 수정사항을 소스코드 안에 전부 표시한다. 이 때 수정 날짜를 포함한다.
- [필수적:] 기존의 LGPL-3.0 라이선스가 적용될 수 있는 방식으로 임베디드 라이브러리 수정사항을 구성한다. 만약 새로운 소스코드 파일을 임베디드 라이브러리에 추가했다면, 헤더에 자신의 저작권 문구와 그 문구가 LGPL-3.0을 충족시키도록 라이선싱 한다는 내용을 추가한다.<sup>775)</sup>
- [필수적:] 라이브러리의 구조적 독립을 유지한다.
- [필수적:] 사용자가 직접 애플리케이션을 링킹할 수 있도록 온탑 개발 및 라이브러리의 모든 오브젝트 파일을 제공하기 위해 온탑 개발 또는 라이브러리를 동적 링킹 가능한 형태로 배포하거나 정적 링킹된 애플리케이션을 최소 3년 동안 유효한 written offer와 함께 배포한다.
- [자발적:] 수정 텍스트 파일이 아직 존재하지 않는다면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 자신이 수정한 부분에 대한 기능적인 설명을 추가한다.
- [자발적:] 배포 문서 및 추가 자료를 통해 또한 기존의 저작권 고지문 내용, 소프트웨어 명칭에 대한 힌트, 홈페이지 링크, 보증 제한, LGPL-3.0 라이선스 링크를 제공한다.
- [자발적:] 기존의 모든 저작권 고지문을 유지한다.

#### 금지 사항:

- 소프트웨어가 특히 침해에 해당한다고 주장하는 자에 대해 특히 소송을 제기

#### 6.9.23 논의 및 설명

- LGPL-2.1에서는 “받은 라이브러리의 전체 소스코드의 verbatim copy를 복사하고 배포할 수 있는데, 단 (a) 적합한 저작권 고지 및 보증 제한 사본을 각각 적절히 눈에 띄게 공표하고; (b) 이 라이선스와 보증 제한 고지문을 인용한 모든 고지를 그대로 두고; (c) 그리고 라이브러리와 함께 이 라이선스 사본을 배포한다는 조건으로 허용한다.<sup>776)</sup> 또한 이 라이선스에서는 “제 1조항에 의거” 수정한 소스코드의 배포를 허용하고<sup>777)</sup> 바이너리는 “제 1조항과 2조항에 의거” 배포를 허용한다.<sup>778)</sup> 그러나 저작물을 개인적인 용도로 사용할 경우

774) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

775) 자세한 내용은 LGPL-3.0의 “조항을 새로운 프로그램에 적용시키기” 섹션을 참조

776) Open Source Initiative: The GPL-2.1 License (OSI), 1999, wp §1 참조

777) 인용한 글과 동일, wp. §2 참조

라이선스 준수를 위한 그 어떠한 작업도 요구하지 않는다. 따라서 “제 1조항”의 조건들은 LGPL-2.1로 라이선스된 저작물을 배포하는 경우에만 의무적으로 지켜져야 한다. (LGPL-2.1-C2 - LGPL-2.1-CB)<sup>779)</sup>

- LGPL의 두 버전은 받았던 형태로 저작권 고지를 유지할 것을 명확히 요구하지 않더라도 이 요소들을 변경하지 않는 것이 바람직하다 (LGPL-2.1-C2 – LGPL-2.1-CB). 반면에 LGPL-3.0의 경우, 저작권 고지를 유지할 것을 명확하게 요구한다 (LGPL-3.0-C2 – LGPL-3.0-CB).<sup>780)</sup>
- LGPL-2.1에서는 “소프트웨어 교환에 통상적으로 사용되는 매체를 통해 기계 판독이 가능한 전체 소스코드 사본을 첨부한다는 조건 하에 오브젝트 코드 또는 실행 가능한 형태로 프로그램 (또는 파생저작물)의 복사 및 배포”를 허용한다. 또한 이 라이선스에서는 직접적으로 배포하지 않더라도 오브젝트 코드에 접근할 수 있는 방법을 제공한다면, 동일한 방법으로 소스코드를 “다운로드” 가능하게 하여도 위 조건을 충족시킬 수 있다고 설명한다.<sup>781)</sup> 따라서 문구 그대로 엄격히 해석하면 LGPL에서는 동일한 방법으로 소스코드와 오브젝트 코드를 함께 배포해야 한다. 예를 들어, 둘 다 DVD를 통해 제공하거나 둘 다 저장소를 통해 제공하는 것은 허용된다. 그러나 둘 중 하나는 DVD를 통해 제공하고, 다른 하나는 저장소를 통해 제공하는 것은 허용되지 않는다. 그러나 첫 번째 설명에서는 또한 “기계 판독 가능한 상응하는 전체 소스코드는 소프트웨어 교환에 통상적으로 사용되는 매체”를 통해 배포해야 한다고 언급하고 있다.<sup>782)</sup> OSPLIC에서는 현재 소프트웨어 교환에 통상적으로 사용되는 매체를 이용한 배포로서 인터넷을 통해 파일을 다운로드하는 가능성을 인식한다. 이에 따라 여기에서는 바이너리 배포를 포함하는 오픈소스 사용 사례(LGPL-2.1-C3, LGPL-2.1-C5, LGPL-2.1-C7, LGPL-2.1-C9, LGPL-2.1-CA)에서 인터넷 저장소를 통해 상응하는 라이브러리의 소스코드를 이용할 수 있게 하는 것을 요구하고 있다. 반면에 LGPL-3.0에 포함된 GPL-3.0은, 인터넷 서버를 통한 소스코드 배포를 명백히 허용한다 (→ OSPLIC, p.238 참조). 이에 따라, 상응하는 LGPL-2.1의 요구사항을 충족하는 일련의 작업들은 GPL-3.0과 LGPL-3.0의 조건들을 충족한다고 결론지을 수 있다.
- LGPL에서는 특정 제한사항 및 조건 하에서만 “라이브러리 사본을 수정하고 그 수정물을 복사 및 배포”할 수 있다.<sup>783)</sup>
  - 첫째, 수정된 파일은 수정되었다는 사실을 표시하고 수정 날짜도 표시되어야 한다.<sup>784)</sup> 이 조건은 수정한 저작물 배포와 연관이 있는 오픈소스 사용 사례에서 준수되어야 한다

778) 인용한 글과 동일, wp. §4. 참조

779) LGPL-3.0에 포함된 GPL-3.0은 동일한 요구사항을 갖기 위해 유사한 구조를 갖는다 (→OSPLIC, p. 194). 이에 따라, 상응하는 LGPL-2.1의 요구사항을 충족하는 일련의 작업들은 GPL-3.0의 조항들을 충족한다고 결론지을 수 있으며, 또한 LGPL-3.0의 조항도 충족한다.

780) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §4 참조

781) Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §4 참조

782) 위 책과 동일 참조

783) 인용한 글과 동일, wp. §2. 참조

784) LGPL-2.1에 대해서는 Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. §2를 참조. GPL-3.0(LGPL-3.0에 포함)에 대해서는 Open Source Initiative: The GPL-3.0 License (OSI), 2007, wp. §5를 참조.

(LGPL-\*~C4 – LGPL-\*~C9). 왜냐하면, 주로 바이너리를 배포할 의향이더라도 소스코드도 또한 제공해야 하기 때문이다. OSLiC에서는 이 조건을 각 수정 파일을 표시해야한다는 필수 요구사항과 changelog 파일을 업데이트/생성해야한다는 자발적 요구사항으로 ‘대체’하고 있다.

- 둘째, 이 라이선스에서는 “애플리케이션에서 그 함수 또는 테이블을 제공하지 않는 경우에도 기능(facility)이 여전히 해당 목적의 일환으로 수행된다는 보장”을 하지 않더라도, 수정된 버전이 외부 데이터 구조에 의존하지 않을 것을 요구한다.<sup>785)</sup> OSLiC에서는 임베디드 컴포넌트로 수정된 라이브러리를 사용하는 경우에 구조적 독립성을 유지하는 의무사항으로 이 조건을 다시 고쳐 쓰고 있다 (LGPL-\*~C8 – LGPL-\*~C9).
- 셋째, LGPL-2.1에서는 “수정된 저작물은 소프트웨어 라이브러리여야 한다”고 분명히 요구한다.<sup>786)</sup> 이 조건은 라이브러리 수정을 포함하는 모든 사용 사례의 금지사항으로 직접 포함될 수 있다 (LGPL-2.1~C8 – LGPL-2.1~CB). 그러나 LGPL-2.1 조항에 기반하여 받았던 프로그램을 수정하고자 한다면 이 조건을 지키기가 어렵다. 원칙적으로는, 애플리케이션을 작성하고 이를 LGPL-2.1로 라이선싱할 수 있다. 그러나 그 결과물이 라이브러리가 될 수 있고, 그럴 경우 이 저작물 수정을 저해한다. LGPL-3.0에서는 이 요구사항을 포함하지 않고 있다. 이에 따라 OSLiC에서는 LGPL-3.0으로 라이선싱된 경우에만 수정한 프로그램의 배포를 허용한다 (LGPL-\*~C6, LGPL-\*~C7). LGPL-2.1 프로그램의 경우, GPL-2.0 (또는 개인 판단에 따라 GPL-3.0도 가능)에 의해 소프트웨어를 다시 라이선싱하는 것이 유일한 선택지이다. 이는 LGPL-2.1 라이선스에서 명확하게 허용되고 있다: “임의의 라이브러리 사본에 대해 이 라이선스 대신 GNU General Public License의 조항을 적용할 수 있다.”<sup>787)</sup>

- 또한 LGPL-2.1에서는 “위의 섹션의 예외사항으로서” “자신이 선택한 라이선스로 허가된 라이브러리”<sup>788)</sup>와 함께 저작권자가 라이브러리의 온탑 개발물인 프로그램<sup>789)</sup> (LGPL-2.1에서는 이를 “라이브러리를 사용하는 저작물”<sup>790)</sup>이라고 칭함) 을 배포하는 것을 허용한다. 단 라이선시는 다음의 추가 조건들을 충족해야 한다.  
첫째, (수정된) 라이브러리에 따라 온탑 개발물이 달라진다는 점이 명시되어야 한다. 둘째, LGPL 은 배포된 패키지에 추가되어야 한다.<sup>791)</sup> LGPL-3.0에서도 이 조건은 유사하다.  
한편으로, “결합 저작물”은 “라이브러리와 애플리케이션을 결합하거나 링킹하여 생성된 저작물”로 정의한다.<sup>792)</sup>

785) LGPL-2.1에 대해서는 Open Source Initiative: The LGPL-3.0 License (OSI), 1999, wp. §2d를 참조.  
LGPL-3.0에 대해서는 Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp. §2a를 참조.

786) Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp. §2. 참조

787) 인용한 글과 동일, wp. §3 참조

788) 혹은 다른 라이브러리

789) 인용한 글과 동일, wp. §5, §6. 참조

790) 인용한 글과 동일, wp. §6. 참조

791) 인용한 글과 동일

792) Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp. §0 참조

반면에, LGPL-3.0에서는 다음 조건들이 충족되면 “결합 저작물을 (자신이) 선택한 조항으로 배포 할 수 있다”고 설명한다: (a) 온탑 개발물이 LGPL로 라이선스된 라이브러리를 사용한다고 명시하고, (b) 패키지의 일부로서 LGPL-3.0과 GPL-3.0 라이선스를 배포하고, (c) 기존 저작권 문구에 이 (라이선스) 정보를 포함하고, (d) 적절한 쉐어드 라이브러리 메커니즘을 필요로 하고, (e) 설치 관련 정보를 제공해야 한다.<sup>793)</sup> 이 요구사항은 두 가지 LGPL 버전에 대해 각 사용 사례 (LGPL-\*-CA, LGPL-\*-CB)의 조건으로 직접 삽입될 수 있다.

- LGPL-2.1에서 가장 까다로운 요구사항은 바이너리 형태로서의 배포와 관계가 있다. 엄격히 볼 때 LGPL에서는 온탑 개발물과의 라이브러리의 동적 링킹을 요구하지 않는다. 처음에 LGPL에서는 “(라이브러리를 사용하는) 저작물은 별개로 봤을 때 라이브러리의 파생저작물이 아니다”라고 언급한다. 그러나 라이브러리와 링킹된다면, 이 결과물인 실행 가능 프로그램은 “라이브러리의 파생저작물”이 되므로 “이 라이선스(LGPL-2.1) 조항들이 적용된다.” 하지만, LGPL-2.1에서는 “섹션 6에서는 그 실행 가능 프로그램 배포에 대한 조항을 명시한다”는 힌트로 이 설명을 이어 나간다.<sup>794)</sup> 마지막으로, 이 섹션 6에서는 다음의 설명으로 시작한다. “위의 섹션들의 예외사항으로서, ‘라이브러리를 사용하는 저작물’을 라이브러리와 결합하거나 링크하여 라이브러리 부분을 포함하는 저작물을 생성하고 선택 조항에 의거 그 저작물을 배포할 수 있다.”<sup>795)</sup>

이 사항은 알아 두는 것이 중요하다. 왜냐하면 섹션 6의 내용이 언급되기 전까지 LGPL-2.1에서는 프로그램과 라이브러리를 동적으로 링크하는 행위를 정적 링크하는 행위와 구분 지었다는 사실을 직간접적으로 추론할 수 없기 때문이다. LGPL에서는 라이브러리의 바이너리를 새로운 버전으로 대체할 수 있도록 보장하는 것만을 기대한다. 이 점은 섹션 6에 의해 요구된다.<sup>796)</sup> 실용적인 측면에서 볼 때 이는 온탑 개발물의 바이너리와 라이브러리가 함께 “적절한 쉐어드 라이브러리 메커니즘으로 사용”<sup>797)</sup>되는 것을 보장하거나, “컴파일은 되었지만 링크되지 않은 온탑 개발물의 오브젝트 파일과 라이브러리와 함께 직접적으로 또는 간접적으로 (최소한 3년간 유효한 written offer) 제공”<sup>798)</sup>하는 것을 보장할 수 있으면 된다. 첫 번째 경우에 사용자는 받은 라이브러리 버전을 교체하여 자동적으로 그 애플리케이션을 다시 링크시킬 수 있다. 두 번째 경우에는 이를 수동으로 해야 한다. 정적 링킹된 저작물을 배포하려 하거나 배포해야 한다면 이 방법들이 존재한다는 점을 알아 둘 필요가 있다. LGPL-2.1에서는 정적 링크된 애플리케이션 배포를 금하지 않는다. 그러나 받는 사람이 저작물을 재링크할 수 있어야 한다고 요구 한다.

LGPL-3.0은 특수한 방법으로 이와 같은 복잡한 조건들을 줄였다. 첫째, 여기서는 “정적링크” 또는 “동적 링크”란 단어를 사용하지 않는다. 둘째, “애플리케이션과 라이브러리를 결합 또는 연결한” 결과물만을 결합 저작물로 정의한다.<sup>799)</sup> 그러나 그 다음에는 결합 저작물 배포에 대해서

793) Open Source Initiative: The LGPL-3.0 License (OSI), 2007, wp §4 참조

794) Open Source Initiative: The LGPL-2.1 License (OSI), 1999, wp §5 참조.

795) 인용한 글과 동일, wp. §6. 참조

796) 위 책과 동일 참조

797) 위 책과 동일 참조

798) 위 책과 동일 참조

799) Open Source Initiative: LGPL-3.0 License (OSI), 2007, wp. §0. 참조

"이 라이선스 조항에 의거 최소의 상응하는 소스코드를 전달하고 그리고 사용자가 애플리케이션을 링크된 버전의 수정 버전과 재결합하거나 재링크하여 수정된 결합 저작물을 생성하는 행위를 허가하는 조항에 기반하여 이에 적합한 형태로 상응하는 애플리케이션 코드를 전달"해야 하거나 혹은 받는 사람은 "라이브러리의 수정 버전과 함께 적절히 동작하는 라이브러리와의 링크를 위한 적절한 쉐어드 라이브러리 메커니즘을" 사용할 것을 요구하고 있다.<sup>800)</sup> 마지막으로, LGPL-3.0에서는 첫 번째 경우에, 재링크가 가능한 이런 자료는 "상응하는 소스를 전달하기 위해 GNU GPL-3.0 섹션 6에 의해 지정된 방식으로" 배포해야 한다고 덧붙이고 있다.<sup>801)</sup> GPL-3.0 섹션 6에서는 "타인에게 무료로 네트워크 서버에서 상응하는 소스코드에 접근하여 복사할 수 있도록 written offer와 함께 오브젝트 코드를 전달하는" 잘 알려진 방법을 허용한다.<sup>802)</sup>

이에 따라, OSLiC에서는 이 조건들을 다음 요구사항으로 정리할 수 있다. 타인이 애플리케이션을 재링크하기 위한 전체 자료를 전달하기 위한 수단으로 동적 링킹이 가능한 파트들을 배포하거나, "최소 3년 동안 유효한 written offer(와 함께)" 정적으로 링크된 애플리케이션을 제공해야한다.<sup>803)</sup> 이 요구사항은 사용 사례 LGPL-\*-C5와 LGPL-\*-CB에서만 유효하다.

## 6.10 MIT 라이선스 소프트웨어

MIT 라이선스는 가장 관대한 라이선스 중 하나로 알려져 있다. 따라서 MIT 상세 파인더를 간단하게 나타낼 수 있다.

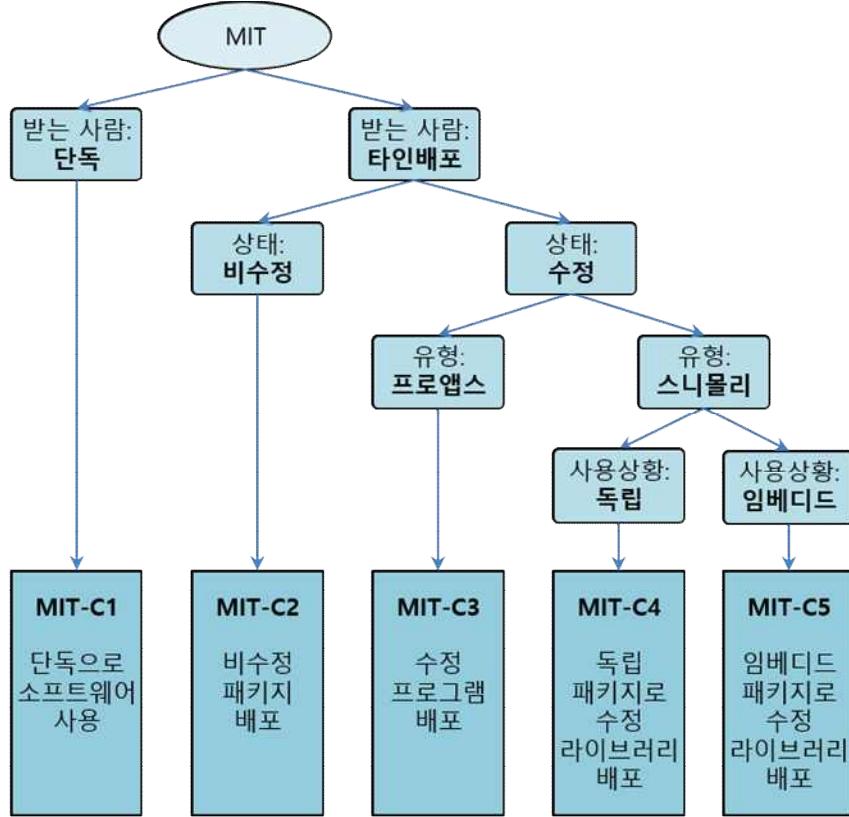
---

800) 인용한 글과 동일, wp. §4. 참조

801) 위 책과 동일 참조

802) Open Source Initiative: GPL-3.0 License (OSI), 2007, wp. §6. 참조

803) Open Source Initiative: LGPL-2.1 License (OSI), 1999, wp. §6. 참조



#### 6.10.1 MIT-C1: 단독으로 소프트웨어 사용

**의미:** MIT 라이선스가 적용된 소프트웨어를 배포하지 않고 단독으로 사용하고, 제3자에게 배포하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>804)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- 제3자에게 소프트웨어를 배포하지 않거나 저작권 문구나 승인 고지 문구를 수정하지 않는다면 MIT 라이선스가 적용된 소프트웨어는 어떠한 사용형태와 조건에 관계없이 사용이 가능하다.

**금지사항:**

- 명시된 사항 없음

#### 6.10.2 MIT-C2: 수정하지 않은 소프트웨어 배포

**의미:** MIT 라이선스가 적용된 소프트웨어의 수정되지 않은 소스코드 파일이나 수정되지 않은 바이너리를 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그인을 독립 패키지로 배포하는 형태와 동일한 형태이다.

804) 자세한 내용 → OSLiC, pp.96-105 참조

**적용범위:** OSUC-02S, OSUC-02B, OSUC-05S, OSUC-05B, OSUC-07S, OSUC-07B<sup>805)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(원저작자의 저작권 고지, 허가 고지, MIT 권리포기 사항이 포함된 MIT 라이선스 텍스트)는 받았던 형태로 패키지에 포함시켜야 한다.
- [자발적:] 배포 문서 및 추가 구성 요소들은 원본 소프트웨어(프로젝트) 및 홈페이지의 링크를 포함한다.

**금지사항:** 명시된 사항 없음

#### 6.10.3 MIT-C3: 수정한 프로그램 배포

**의미:** MIT 라이선스가 적용된 프로그램, 애플리케이션, 서버(proapse)를 수정한 뒤 수정된 버전을 제3자에게 바이너리나 소스코드 파일 형태로 배포하는 형태이다.

**적용범위:** OSUC-04S, OSUC-04B<sup>806)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(원저작자의 저작권 고지, 허가 고지, MIT 권리포기 사항이 포함된 MIT 라이선스 텍스트)는 받았던 형태로 패키지에 포함시켜야 한다.
- [자발적:] 코드의 배포관계 없이 소스코드가 수정된 부분을 표시한다.
- [자발적:] 배포 문서 및 추가 구성 요소들은 원본 소프트웨어(프로젝트) 및 홈페이지의 링크를 포함한다.
- [자발적:] 자신의 저작물 또는 수정에 관한 정보를 기준 저작권 내용에 추가할 수 있다.
- [자발적:] 프로그램 실행과 함께 저작권 정보를 표시하는 것과 MIT 라이선스가 적용된 컴포넌트를 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 그리고, 컴포넌트의 홈페이지 및 다운로드 페이지 링크를 포함한다.

**금지사항:**

- 명시된 사항 없음

#### 6.10.4 MIT-C4: 수정한 라이브러리를 독립적으로 배포

**의미:** MIT 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 바이너리나 소스코드 파일 형태로 제3자에게 배포하는 형태이며, 다른 더 큰 소프트웨어에 포함하지 않는 형태이다.

---

805) 자세한 내용 → OSLiC, pp.96-102 참조

806) 자세한 내용 → OSLiC, pp.98-99 참조

**적용범위:** OSUC-08S, OSUC-08B<sup>807)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(원저작자의 저작권 고지, 허가 고지, MIT 권리포기 사항이 포함된 MIT 라이선스 텍스트)는 받았던 형태로 패키지에 포함시켜야 한다.
- [자발적:] 코드의 배포관계 없이 소스코드가 수정된 부분을 표시한다.
- [자발적:] 배포 문서 및 추가 구성 요소들은 원본 소프트웨어(프로젝트) 및 홈페이지의 링크를 포함한다.

**금지사항:**

- 명시된 사항 없음

#### 6.10.5 MIT-C5: 수정한 라이브러리를 임베디드 컴포넌트로 배포

**의미:** MIT 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snippets)을 수정한 뒤 수정된 버전을 바이너리나 소스코드 파일 형태로 제3자에게 배포하는 형태이며, 이는 스니펫, 모듈, 라이브러리, 임베디드 컴포넌트와 같은 플러그인을 포함하는 더 큰 소프트웨어를 바이너리나 소스코드 파일에 관계 없이 함께 배포하는 형태이다.

**적용범위:** OSUC-10S, OSUC-10B<sup>808)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(원저작자의 저작권 고지, 허가 고지, MIT 권리포기 사항이 포함된 MIT 라이선스 텍스트)는 받았던 형태로 패키지에 포함시켜야 한다.
- [자발적:] 코드의 배포관계 없이 소스코드가 수정된 부분을 표시한다.
- [자발적:] 프로그램 실행과 함께 저작권 정보를 표시하는 것과 MIT 라이선스가 적용된 컴포넌트를 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 그리고, 컴포넌트의 홈페이지 및 다운로드 페이지 링크를 포함한다.
- [자발적:] 배포 문서 및 추가 구성 요소들은 원본 소프트웨어(프로젝트) 및 홈페이지의 링크를 포함한다.
- [자발적:] 라이선싱 원본 요소(원저작자의 저작권 고지, 허가 고지, MIT 권리포기 사항이 포함된 MIT 라이선스 텍스트)는 임베디드 라이브러리만을 명확히 참조하고 자신이 보유한 지배적인 저작물의 라이선싱을 침해하지 않도록 배포물을 구성한다. 라이브러리, 모듈, 스니펫, 플러그인을 모든 라이선싱 구성요소를 포함한 디렉토리에 포함시키는 것을 권장한다.

---

807) 자세한 내용 → OSLiC, pp.103-104 참조

808) 자세한 내용 → OSLiC, pp.105-106 참조

#### 금지사항:

- 명시된 사항 없음

#### 6.10.6 논의 및 설명

MIT라이선스는 가장 관대한 라이선스로 알려져 있다. (0) 저작권 고지, (1) 사용자가 하려는 모든 행위가 거의 허용된다는 절의 내용을 포함하고, (2) “모든 사본 또는 소프트웨어의 모든 부분”에 기존의 저작권 고지와 허가 고지를 “포함” 해야 한다는 조건이 이어지며, 그리고 (3) 잘 알려진 권리포기 내용을 포함한 매우 간결한 내용의 라이선스다.<sup>809)</sup> 그러나 이 라이선스에서는 소스코드와 오브젝트 코드의 차이에 대해 언급하지 않았다. 따라서 스스로 올바른 방법을 찾아야한다. 우리의 해석은 다음과 같다.

- 자신의 목적을 위한 것이 아니거나, 제3자에게 배포하지 않거나, MIT가 적용된 애플리케이션을 수정하지 않는다면 모든 저작권 고지와 허가 고지는 이미 올바른 것으로 결론지을 수 있다.
- 그럼에도 혼자 사용하는 사용 사례에서 라이선싱 요소를 수정하지 않기 위해 힌트를 추가하였다. 이는 MIT 라이선스 자체에 의해 유발된다. 이 라이선스에서는 “위의 저작권 고지와 허가 고지는 소프트웨어의 모든 사본 또는 상당 부분에 포함해야 한다”<sup>810)</sup>고 명백히 요구한다. 따라서 자신의 목적을 위해 작성한 부분에도 포함되어야 한다.
- 어떠한 목적이든 MIT가 적용된 애플리케이션을 수정한다면 기존의 저작권 고지와 허가 고지를 삭제하거나 수정할 수 없다. 새 조건 하에 자체 수정물을 추가할 수 있지만 기존 고지사항은 존속해야 한다.
- MIT 권리 포기사항을 유지해야 한다. 이는 라이선스가 요구하는 필수 조건은 아니다. 반드시 포함해야 하는 허가 고지는 저작권 고지 및 권리 포기사항 텍스트 사이에 포함되어야 한다. 가능성이 있는 또 다른 해석으로는 라이선스 텍스트는 어떠한 허가 고지가 참고되고 있다는 것이다.

#### 6.11 MPL-2.0 라이선스 소프트웨어

Mozilla Public License는 명확하게 바이너리 배포와 소스코드 배포를 구분하고 있다. 첫째, “소스 코드 형태 배포”<sup>811)</sup>를 허용한다. 그 다음 “실행 가능 형태의” 배포의 조건을 명시하였다.<sup>812)</sup> 추가적으로, MPL-2.0은 “보호된 소프트웨어(Covered Software)의 배포” 및 “조합 저작물(Larger Work)의 배포”와 대조적이다.<sup>813)</sup> MPL-2.0은 소프트웨어 배포에 중점을 둔다. 따라서 처리 가능한 적합한 작업 리스트를 찾기 위해서는 다음의 MPL-2.0 상세 오픈소스 사용 사례 구조<sup>814)</sup>를 사용할 수 있다.

---

809) Open Source Initiative: The MIT License, 2012, wp. 참조

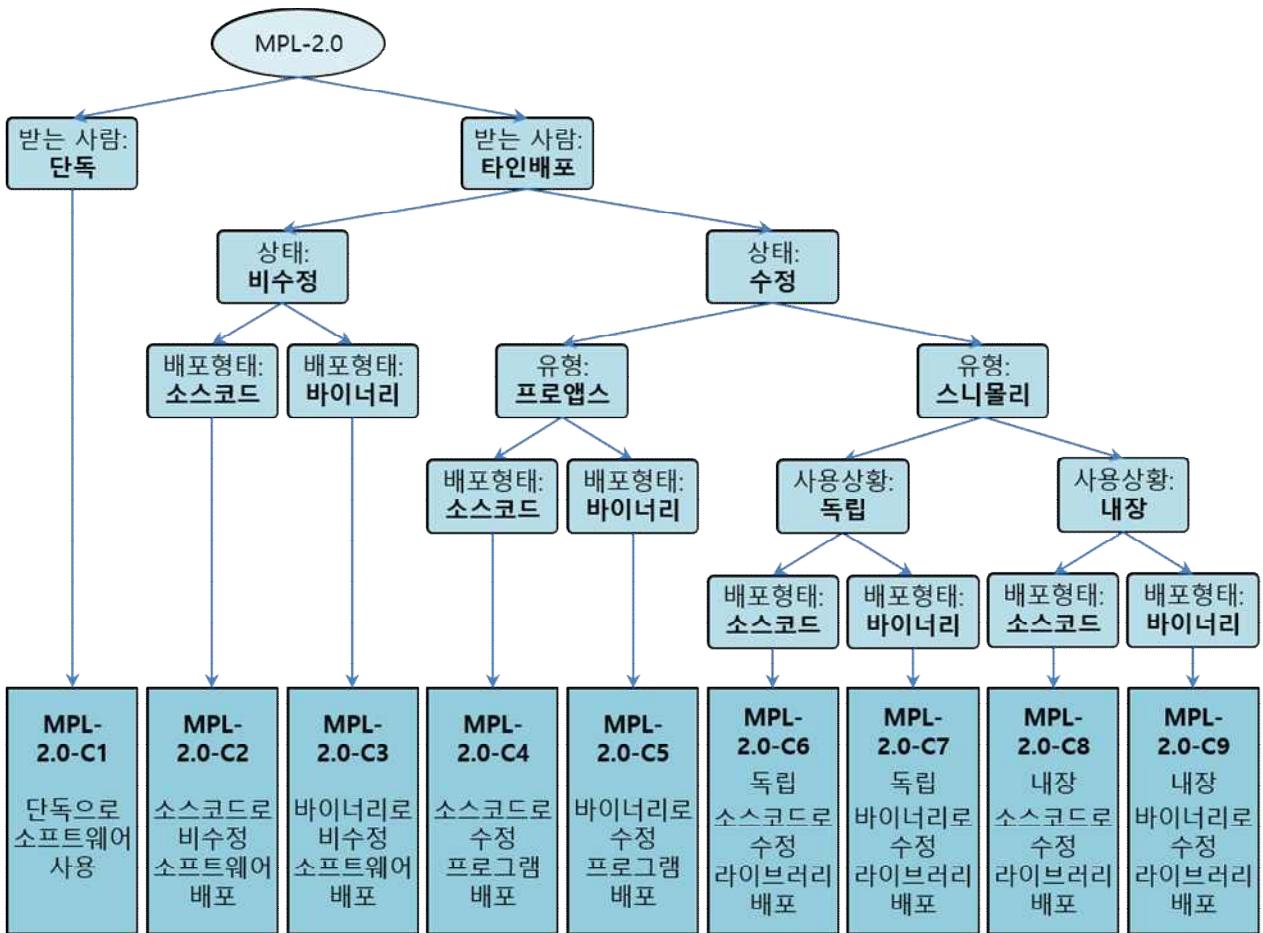
810) Open Source Initiative: The MIT License, 2012, wp. 참조

811) Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp §3.1 참조

812) 위와 동일, wp §3.2 참조

813) 위와 동일, wp §3.3 참조

814) 일반적인 OSUC 파인더의 자세한 사항 → OSLiC, pp.89



### 6.11.1 MPL-2.0-C1: 단독으로 소프트웨어 사용

**의미:** MPL-2.0 라이선스가 적용된 소프트웨어를 배포하지 않고 단독으로 사용하고, 제3자에게 배포하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>815)</sup>

**요구사항:** Mozilla Public License 2.0 라이선스 조건을 충족하기 위한 작업은 없다.

- 제3자에게 소프트웨어를 배포하지 않는다면 MPL-2.0 라이선스가 적용된 소프트웨어는 어떠한 사용형태와 조건에 관계없이 사용이 가능하다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

815) 자세한 내용 → OSLiC, pp.96-105 참조

### 6.11.2 MPL-2.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포

**의미:** MPL-2.0이 적용된 소프트웨어를 수정되지 않은 소스코드 파일이나 수정되지 않은 소스코드 패키지를 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그인을 독립되거나 임베디드 유닛으로 배포하는 형태와 동일한 형태이다.

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>816)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [필수적:] 받는 사람에게 MPL-2.0 라이선스 사본을 제공한다. 만약, 소프트웨어 패키지에서 빠져 있다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0 라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>817)</sup>
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

**금지사항:**

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다

### 6.11.3 MPL-2.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포

**의미:** MPL-2.0 라이선스가 적용된 소프트웨어를 수정되지 않은 바이너리 파일이나 수정되지 않은 바이너리 패키지를 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그인을 독립되거나 임베디드 유닛으로 배포하는 형태와 동일한 형태이다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>818)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

816) 자세한 내용 → OSLiC, pp.96-102 참조

817) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

818) 자세한 내용 → OSLiC, pp.97-103 참조

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다. 소스코드를 바이너리로 컴파일 한다면 모든 라이선싱 요소들은 패키지 안에 포함되도록 한다
- [필수적:] 배포되는 소프트웨어의 소스코드는 제어 가능한 저장소(Repository)를 통해 이용할 수 있도록 한다. 인터넷을 통해 저장소에 소스코드 패키지를 저장하거나 다운로드 할 수 있도록 적용한다. 다운로드에 대한 비용을 부과하지 않는다. 저장소는 소프트웨어를 배포한 뒤 충분한 기간 동안 사용할 수 있도록 한다.
- [필수적:] 배포 및 추가 자료에 다운로드 저장소의 힌트를 눈에 띄게 삽입한다.
- [필수적:] 사용자가 생산한 소스코드를 위해 사용 사례 MPL-2.0-C2를 실행한다.<sup>819)</sup>
- [자발적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0 라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>820)</sup>
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

#### 6.11.4 MPL-2.0-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** MPL-2.0 라이선스가 적용된 프로그램, 애플리케이션, 서버(proapse)를 수정한 버전의 소스코드 파일이나 소스코드 패키지 형태로 제3자에게 배포한다.

**적용범위:** OSUC-04S<sup>821)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [필수적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다.

819) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

820) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

821) 자세한 내용 → OSLiC, pp.98 참조

만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/.><sup>822)</sup>

- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함하는 방식으로 수정물을 구성한다. 새로운 소스 코드 파일을 추가한다면 사용자의 저작권 라인을 포함한 헤더나 MPL-2.0을 충족하는 라이선싱 성명을 추가한다.
- [자발적:] 만약, 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] 소스코드 안에 소스코드의 모든 수정사항을 표시한다.
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

#### 6.11.5 MPL-2.0-C5: 수정한 프로그램을 바이너리 형태로 배포

**의미:** MPL-2.0이 적용된 프로그램, 애플리케이션, 서버(proapce)를 수정한 버전의 바이너리 파일이나 바이너리 패키지 형태로 제3자에게 배포한다.

**적용범위:** OSUC-04S<sup>823)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소( 모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제는 받았던 형태로 패키지에 포함한다. 소스코드를 바이너리로 컴파일 한다면 모든 라이선싱 요소들은 패키지 안에 포함되도록 한다.
- [필수적:] 배포되는 소프트웨어의 소스코드는 제어 가능한 저장소(Repository)를 통해 이용할 수 있도록 한다. 인터넷을 통해 저장소에 소스코드 패키지를 저장하거나 다운로드 할 수 있도록 적용한다. 다운로드에 대한 비용을 부과하지 않는다. 저장소는 소프트웨어를 배포한 뒤 충분한 기간 동안 사용할 수 있도록 한다.
- [필수적:] 어디서/어떻게 코드를 취득할 수 있는지에 대한 설명을 배포 패키지에서 쉽게 찾을 있

822) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

823) 자세한 내용 → OSLiC, pp.98 참조

도록 한다.

- [필수적:] 사용자가 생산한 소스코드를 위해 사용 사례 MPL-2.0-C4를 실행한다. 824)
- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함되는 방식으로 수정물을 구성한다.
- [자발적:] 만약, 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>825)</sup>
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

#### 6.11.6 MPL-2.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** MPL-2.0 라이너스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 제3자에게 소스코드 파일이나 소스코드 패키지 형태로 배포한다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08S<sup>826)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [필수적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0라이선싱 파일을 삽입해야 한다. : This

824) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

825) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

826) 자세한 내용 → OSLiC, pp.103 참조

Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>827)</sup>

- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함하는 방식으로 수정물을 구성한다. 새로운 소스코드 파일을 추가한다면 사용자의 저작권 라인을 포함한 헤더나 MPL-2.0을 충족하는 라이선싱 성명을 추가한다.
- [자발적:] 만약, 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] 소스코드 안에 소스코드의 모든 수정사항을 표시한다.
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

## 금지사항

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

### 6.11.7 MPL-2.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** MPL-2.0으로 라이선스 된 스니펫, 모듈, 라이브러리, 플러그인(스니몰리)을 수정한 뒤 수정된 버전을 제3자에게 바이너리 파일이나 바이너리 패키지 형태로 배포한다. 단, 이를 다른 더 큰 소프트웨어 유닛에 삽입하지는 않는다.

**적용범위:** OSUC-08B<sup>828)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다. 소스코드를 바이너리로 컴파일 한다면 모든 라이선싱 요소들은 패키지 안에 포함되도록 한다.
- [필수적:] 배포되는 소프트웨어의 소스코드는 제어 가능한 저장소(Repository)를 통해 이용할 수 있도록 한다. 인터넷을 통해 저장소에 소스코드 패키지를 저장하거나 다운로드 할 수 있도록 적용한다. 다운로드에 대한 비용을 부과하지 않는다. 저장소는 소프트웨어를 배포한 뒤 충분한 기

827) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

828) 자세한 내용 → OSLiC, pp.103 참조

간 동안 사용할 수 있도록 한다.

- [필수적:] 어디서/어떻게 코드를 취득할 수 있는지에 대한 설명을 배포 패키지에서 쉽게 찾을 있도록 한다.
- [필수적:] 사용자가 생산한 소스코드를 위해 사용 사례 MPL-2.0-C6를 실행한다.<sup>829)</sup>
- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함되는 방식으로 수정물을 구성한다.
- [자발적:] 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0 라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/><sup>830)</sup>
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

#### 6.11.8 MPL-2.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** MPL-2.0으로 라이선스 된 스니펫, 모듈, 라이브러리, 플러그인(스니몰리)을 수정한 뒤 수정된 버전을 제3자에게 소스코드 파일이나 소스코드 패키지 형태로 배포한다. 이 경우 단편 코드, 모듈, 라이브러리, 플러그인을 임베디드 컴포넌트와 같은 형태로 포함한 더 큰 소프트웨어 유닛을 함께 배포한다.

**적용범위:** OSUC-10S<sup>831)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소( 모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았

829) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

830) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

831) 자세한 내용 → OSLiC, pp.105 참조

던 형태로 패키지에 포함한다

- [필수적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>832)</sup>
- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함하는 방식으로 수정물을 구성한다. 새로운 소스 코드 파일을 추가한다면 사용자의 저작권 라인을 포함한 헤더나 MPL-2.0을 충족하는 라이선싱 성명을 추가한다.
- [자발적:] 소스코드 배포를 준비한다. 라이선싱 요소(MPL-2.0 라이선스 텍스트 및 라이선싱 파일)는 임베디드 라이브러리만 참고하며 사용자가 작성한 저작물의 라이선싱에 영향을 주지 않는다. 라이브러리, 모듈, 스니펫, 플러그인과 같은 독립 디렉토리에 있는 임베디드 컴포넌트는 모든 추가적인 라이선싱 요소를 포함한다.
- [자발적:] 만약, 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] 소스코드 안에 소스코드의 모든 수정사항을 표시한다.
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, 소프트웨어의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

#### 6.11.9 MPL-2.0-C9: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** MPL-2.0으로 라이선스 된 스니펫, 모듈, 라이브러리,

플러그인(스니클리)을 수정한 뒤 수정된 버전을 제3자에게 바이너리 파일이나 바이너리 패키지 형태로 배포한다. 이 경우 스니펫, 모듈, 라이브러리, 플러그를 임베디드 컴포넌트와 같은 형태로 포함한 더 큰 소프트웨어 유닛을 함께 배포한다.

**적용범위:** OSUC-10B<sup>833)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

832) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

833) 자세한 내용 → OSLiC, pp.106 참조

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다. 소스코드를 바이너리로 컴파일 한다면 모든 라이선싱 요소들은 패키지 안에 포함되도록 한다.
- [필수적:] 배포되는 소프트웨어의 소스코드는 제어 가능한 저장소(Repository)를 통해 이용할 수 있도록 한다. 인터넷을 통해 저장소에 소스코드 패키지를 저장하거나 다운로드 할 수 있도록 적용한다. 다운로드에 대한 비용을 부과하지 않는다. 저장소는 소프트웨어를 배포한 뒤 충분한 기간 동안 사용할 수 있도록 한다.
- [필수적:] 어디서/어떻게 코드를 취득할 수 있는지에 대한 설명을 배포 패키지에서 쉽게 찾을 있도록 한다.
- [필수적:] 사용자가 생산한 소스코드를 위해 사용 사례 MPL-2.0-C8를 실행한다.<sup>834)</sup>
- [필수적:] 기존 MPL-2.0 라이선싱 사항에 포함하는 방식으로 수정물을 구성한다.
- [자발적:] 만약, 고지문 파일이 존재하지 않으면 수정 텍스트 파일을 작성한다. 수정 텍스트 파일에 수정된 부분에 대한 일반적인 내용을 작성한다. 이를 배포 패키지에 포함한다.
- [자발적:] MPL-2.0 사본을 제공한다. 만약, 소프트웨어 패키지에 포함되어 있지 않다면 추가한다. 만약, 패키지의 라이선싱 파일에 있는 라이선스 원문에 MPL-2.0으로 배포되고 있다는 내용이 명확하게 표현되지 않았다면 추가적으로 올바른 MPL-2.0 라이선싱 파일을 삽입해야 한다. : This Source Code Form is subject to the terms of the Mozilla Public License, v 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.<sup>835)</sup>
- [자발적:] 소스코드 배포를 준비한다. 라이선싱 요소(MPL-2.0 라이선스 텍스트 및 라이선싱 파일)는 임베디드 라이브러리만 참고하며 사용자가 작성한 저작물의 라이선싱에 영향을 주지 않는다. 라이브러리, 모듈, 스니펫, 플러그인과 같은 독립 디렉토리에 있는 임베디드 컴포넌트는 모든 추가적인 라이선싱 요소를 포함한다.
- [자발적:] 배포물 문서나 추가 자료는 현재 저작권 고지 텍스트 파일, MPL-2.0 라이선스가 적용된 컴포넌트의 이름, 홈페이지 링크, MPL-2.0 라이선스의 링크와 같은 콘텐츠를 제공한다.

#### 금지사항:

- 소프트웨어 패키지에 포함된 라이선스 요소(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)를 수정하거나 삭제하는 것을 금지한다.
- 사용된 소프트웨어를 공정하게 설명하고 저작권 고지를 재생성 하는 경우를 제외하고 MPL-2.0 소프트웨어에 연결된 로고, 상표, 서비스 마크로 자신의 서비스를 해당 소프트웨어에 근거하여 홍보하는 것을 금지한다.

834) 저장소를 통해 코드를 공개한다는 것은 소프트웨어를 소스코드 형태로 배포한다는 것과 동일하다. 따라서 해당 사용 사례의 모든 의무사항 또한 준수해야 한다.

835) 파일에 대한 양도를 올바르게 수행 → OSLIC, p.107

### 6.11.10 논의 및 설명

MPL-2.0에서는 거의 모든 요구사항을 포함하는 “책임” 세션을 제공한다.<sup>836)</sup> 일부 부차적인 측면을 위해 측면을 위해 다른 단락을 또한 숙고해야 한다.<sup>837)</sup> 이 구성과 관련하여 여기에서 다음의 작업을 알 수 있다.

- 일반적으로 MPL-2.0에서는 라이선스의 다른 요구사항을 “준수하는데” 필요한 경우를 제외하고는 기여자의 로고, 상표, 서비스 마크에 대한 권리를 부여하지 않는다.<sup>838)</sup> OSLiC에서는 이 메시지를 자체 서비스와 제품을 그러한 요소로 홍보하는 행위를 금지하는 것으로 다시 작성하였다.
- “소스코드 형태 안에 포함된 어떠한 라이선스 고지(저작권 고지, 특히 고지, 보증 제한 또는 책임 배제를 포함)의 실체를 제거하거나 변경할 수 없다”고 규정한다.<sup>839)</sup> 라이선스 고지의 실체에 집중하는 부분은 “알려진 사실에 기반을 둔 부정확성을 개선하는데 필요한 정도까지 라이선스 고지의 변경”을 허용한다는 것을 의미한다.<sup>840)</sup> 신뢰성 있는 방법을 제공하고 일치하지 않는 부차적인 중요성을 간과하는 원칙에 따라 OSLiC에서는 라이선싱 자료를 수정하기 위해 이 일반 규정 조건을 간소화하였다(즉, 모든 사용 사례(MPL-2.0-C1 ~ MPL-2.0-C9). 그러나 능동적으로 수행해야 하는 작업임을 강조하기 위해 OSLiC에서는 추가로 확보하였던 형태로 라이선싱 고지를 보유하는 작업으로서 모든 타인배포 사용 사례(MPL-2.0-C2 ~ MPL-2.0-C9)에 이 금지사항을 다시 작성하였다.
- 또한, MPL-2.0에서는 모든 “소스 코드 형태의 배포”에 대해 모든 소프트웨어의 수정은 “MPL-2.0 조항에 기반하며”, 배포자는 “모든 ‘받는 사람’에게 소프트웨어는 MPL-2.0 조항에 의거 규제된다”는 사실과 받는 사람이 라이선스 사본을 확보할 수 있는 방법을 알려야 한다”는 점을 요구한다.<sup>841)</sup> 각 사용 사례(MPL-2.0-C2, MPL-2.0-C4, MPL-2.0-C6, MPL-2.0-C8)에 대해서 OSLiC에서는 각MPL-2.0 소스코드 패키지는 텍스트파일, 추가 라이선싱 파일 또는 MPL-2.0의 부록으로 제공된 텍스트를 염격히 따르는 설명으로 MPL-2.0을 의무적으로 포함하도록 이 조건을 수정하였다.<sup>842)</sup> MPL-2.0은 약한 효력의 카피레프트를 갖는 “유일한” 라이선스이므로, OSLiC에서는 연속된(enclosing) 프로그램에서MPL-2.0 라이선스가 적용된 임베디드 컴포넌트를 분리할 것을 제안한다. (MPL-2.0-C8)
- 그러나, MPL-2.0에서는 모든 수정을 표시할 것을 분명히 요구하지 않는다. 그래도 이는 컴퓨터 엔지니어링의 측면에 해당한다. 이에 따라 수정된 소스코드 배포의 경우와 관련하여(MPL-2.0-C4, MPL-2.0-C6, MPL-2.0-C8) OSLiC에서는 소스코드 내부의 모든 수정사항을 표시하고 기능적 변화

836) Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp §3 참조.

837) cf. id., l.c., wp pars pro to cf..

838) cf. id., l.c., wp x2.3.

839) Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp §3.4 참조.

840) 위와 동일.

841) Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp §3.1 참조.

842) 위와 동일, wp Exhibit A 참조

에 관한 설명을 업데이트할 것을 제안한다. 바이너리 형태로 수정된 소프트웨어를 배포하는 경우, 기능적 수준에서만 수정사항을 설명하는 것이 충분하다.

- 또한 MPL-2.0에서는 “보호된 소프트웨어(Covered Software)”를 “실행 가능 형태(Executable Form)”로 배포하는 경우(MPL-2.0-C3, MPL-2.0-C5, MPL-2.0-C7, MPL-2.0-C9)는 “소스코드 형태로 이용할 수 있게” 하고 배포자는 “받는 사람에게 적시에 합당한 수단에 의해 소스코드 형태 사본을 확보할 수 있는 방법과 실행 가능 형태에 대해 알려야 하며 소스코드를 배포하는 비용 이상은 청구할 수 없다.”<sup>843)</sup> OSLiC에서는 무료로 다운로드 서비스를 제공하고 배포된 패키지 내부의 이 서비스를 가리키는 의무로 이 조건을 다시 해석하였다.
- 사용상황 측면에서 MPL에서는 “실행 가능 형태의 라이선스가 이 라이선스에 의거 소스코드를 받는 사람의 권리를 제한하거나 변경하려 하지 않는다면” 다른 라이선스 조항에 의거 바이너리를 배포할 수 있다.<sup>844)</sup> 이 가능성은 라이선스 호환성이 명확히 취급될(managed) 경우에 중요할 수 있다. 일반적으로, MPL-2.0에 의거 바이너리를 배포하는데 이는 충분하다. 따라서 바이너리를 배포하는 경우(MPL-2.0-C3, MPL-2.0-C5, MPL-2.0-C7, MPL-2.0-C9), OSLiC에서는 배포 패키지에 MPL-2.0, 추가 라이선싱 파일 또는 MPL-2.0의 부록으로 제공된 텍스트를 엄격히 따르는 설명에 삽입할 것을 제안하였다.<sup>845)</sup> 그러나 MPL-2.0은 약한 효력의 카피레프트를 갖는 “유일한” 라이선스이므로, OSLiC에서는 지배적인(overarching) 프로그램에서 MPL-2.0 라이선스가 적용된 임베디드 컴포넌트를 분리할 것을 제안한다(MPL-2.0-C9).
- 마지막으로, 위의 과정은 실제 소스코드 배포 행위를 발생시키므로 소프트웨어 배포 규정을 따라야 함을 명시해야 한다. 따라서 OSLiC에서는 바이너리 배포의 경우에 각 소스코드 사용 사례의 작업 리스트를 실행할 것을 요구한다.

## 6.12 Microsoft Public 라이선스

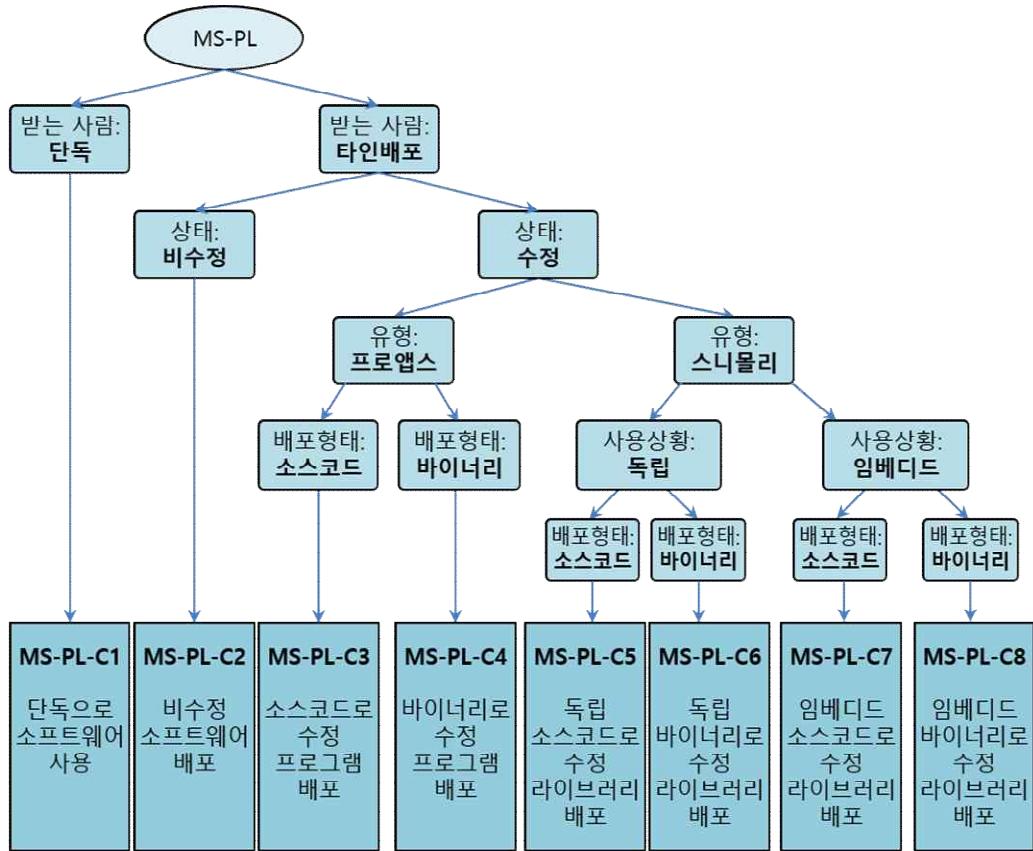
MS-PL 라이선스도 가장 관대한 라이선스 중 하나다. 따라서 MS-PL 상세 파인더는 다음과 같이 간략하게 나타낼 수 있다.

---

843) Open Source Initiative: The MPL-2.0 License (OSI), 2013, wp §3.2.a 참조.

844) 위와 동일, wp §3.2.b 참조.

845) 위와 동일, wp Exhibit A 참조



### 6.12.1 MS-PL-C1: 단독으로 소프트웨어 사용

**의미:** MS-PL 라이선스가 적용된 소프트웨어를 단독으로 사용하고, 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>846)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- 제3자에게 소프트웨어를 배포하지 않는다면 어떠한 의무가 부과되지 않는 정황에서 어떠한 유형의 MS-PL 라이선스 소프트웨어를 사용할 수 있다.

**금지사항:**

- 허가 없이 기여자의 이름, 로고, 트레이드마크를 사용할 수 없음

### 6.12.2 MS-PL-C2: 수정하지 않은 소프트웨어 배포

**의미:** MS-PL 라이선스가 적용된 소프트웨어의 비 수정 버전을 바이너리 형태로 또는 소스코드 파일로 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 단편 코드, 모듈, 라이브러리, 플러그를 독립 패키지로 배포하는 것과 다르지 않다.

846) 자세한 내용 → OSLiC, pp.96-105 참조

**적용범위:** OSUC-02S, OSUC-02B, OSUC-05S, OSUC-05B, OSUC-07S, OSUC-07B<sup>847)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [필수적:] 소스코드 또는 바이너리 패키지의 배포 여부와 관계없이 MS-PL 라이선스 사본을 패키지에 전부 포함한다.<sup>848)</sup>
- [자발적:] 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.

**금지사항:**

- 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

### 6.12.3 MS-PL-C3: 수정한 프로그램을 소스코드 형태로 배포

**의미:** MS-PL 라이선스가 적용된 소프트웨어의 수정된 버전을 소스코드 파일로 또는 소스코드 패키지로 제3자에게 배포한다.

**적용범위:** OSUC-04S<sup>849)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 작업은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [필수적:] 소스코드 또는 바이너리 패키지의 배포 여부와 관계없이 MS-PL 라이선스 사본을 패키지에 전부 포함한다.
- [필수적:] MS-PL 라이선스로 사용자가 수정한 내용을 배포하고 싶지 않다면, 사용자의 소스코드와 라이선싱 문서가 적용된 작업의 초기 요소들로부터 깔끔하게 분리한다.
- [자발적:] 소스코드 내 수정사항을 표시한다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 사용자가 수정한 부분을 언급하기 위해 프로그램의 저작권 고지에 추가할 수 있다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 프로그램 실행과 함께 MS-PL라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 이미 프로그램을 수정하였기 때문에 초기 저작권 고지에 그 설명이 결여되어 있다면 그런 힌트를 추가할 수 있다.

**금지사항:** 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

847) 자세한 내용 → OSLiC, pp.96-102 참조

848) OSLiC, p.239 참조

849) 자세한 내용 → OSLiC, pp.98 참조

#### 6.12.4 MS-PL-C4: 수정한 프로그램을 바이너리 형태로 배포

**의미:** MS-PL 라이선스가 적용된 프로그램, 애플리케이션 또는 서버(proapse)의 수정 버전을 바이너리 파일 또는 바이너리 패키지로 제3자에게 배포한다.

**적용범위:** OSUC-04B<sup>850)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 프로그램 실행과 함께 MS-PL라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 이미 프로그램을 수정하였기 때문에 초기 저작권 고지에 그 설명이 결여되어 있다면 그런 힌트를 추가할 수 있다.

**금지사항:**

- 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

#### 6.12.5 MS-PL-C5: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** MS-PL라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 소스코드 파일이나 소스코드 패키지 형태로 제3자에게 배포하는 형태이며, 다른 더 큰 소프트웨어 유닛에 포함하지 않는 형태이다.

**적용범위:** OSUC-08S<sup>851)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 패키지에 포함한다.
- [필수적:] MS-PL 라이선스 사본을 패키지에 포함한다.
- [필수적:] MS-PL 라이선스로 사용자가 수정한 내용을 배포하고 싶지 않다면, 사용자의 소스코드 와 라이선싱 문서가 적용된 작업의 초기 요소들로부터 깔끔하게 분리한다.
- [자발적:] 소스코드 내 수정사항을 표시한다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.

---

850) 자세한 내용 → OSLiC, pp.99 참조

851) 자세한 내용 → OSLiC, pp.103 참조

#### **금지사항:**

- 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

#### **6.12.6 MS-PL-C6: 수정한 라이브러리를 독립적인 바이너리 형태로 배포**

**의미:** MS-PL라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 바이너리 파일이나 바이너리 패키지 형태로 제3자에게 배포하는 형태이며, 다른 더 큰 소프트웨어 유닛에 포함하지 않는 형태이다.

**적용범위:** OSUC-08B<sup>852)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어 (프로젝트)와 그 홈페이지 링크를 포함시킨다.

**금지사항:** 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

#### **6.12.7 MS-PL-C7: 수정한 라이브러리를 임베디드 소스코드 형태로 배포**

**의미:** MS-PL라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 소스코드파일이나 소스코드 패키지 형태로 제3자에게 배포하는 형태이며, 스니펫, 모듈, 라이브러리, 플러그인을 임베디드 컴포넌트 형태로 포함한 더 큰 소프트웨어 유닛과 함께 배포하는 형태다.

**적용범위:** OSUC-10S<sup>853)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [자발적:] 라이선싱 하는 요소(모든 저작권 고지, 특히 고지, 보증 제한 또는 책임 배제)는 받았던 형태로 패키지에 포함한다.
- [자발적:] MS-PL 라이선스 사본을 패키지에 포함한다.
- [자발적:] MS-PL 라이선스로 사용자가 수정한 내용을 배포하고 싶지 않다면, 사용자의 소스코드 와 라이선싱 문서가 적용된 작업의 초기 요소들로부터 깔끔하게 분리한다.
- [자발적:] 소스코드 내 수정사항을 표시 한다
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어 (프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 프로그램 실행과 함께 MS-PL라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다.

852) 자세한 내용 → OSLiC, pp.103 참조

853) 자세한 내용 → OSLiC, pp.105 참조

원 출처의 프로젝트 홈페이지를 추가하는 방법을 사용하여도 된다.

**금지사항:** 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

#### 6.12.8 MS-PL-C8: 수정한 라이브러리를 임베디드 바이너리 형태로 배포

**의미:** MS-PL라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 바이너리 파일이나 바이너리 패키지 형태로 제3자에게 배포하는 형태이며, 스니펫, 모듈, 라이브러리, 플러그인을 임베디드 컴포넌트 형태로 포함한 더 큰 소프트웨어 유닛과 함께 배포하는 형태다.

**적용범위:** OSUC-10B<sup>854)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 작업은 다음과 같다.

- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 아래 명시된 금지사항과 상충하지 않는 한 프로그램 실행과 함께 MS-PL라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다.  
원 출처의 프로젝트 홈페이지를 추가하는 방법을 사용하여도 된다.

**금지사항:** 추가적인 승인 또는 법에 의한 승인 없이 기여자의 이름, 로고 또는 상표사용을 금지한다.

#### 6.12.9 논의 및 설명

MS-PL은 매우 관대하고 간결한 라이선즈다. 여기에서는 (a) 기존의 라이선싱 하는 요소를 보존해야 한다. (b) MS-PL 라이선스 하에 소스코드의 전체 또는 "부분"으로 소스코드를 배포해야 한다. (c) 소스코드 일부라도 배포한다면 라이선스 사본을 추가해야 한다. (d) 바이너리 패키지를 배포한다면 "MS-PL 라이선스에 부합하는" 라이선스에 의거 저작물 (일부)을 배포해야 한다.<sup>855)</sup>

가장 애매한 조항은 아마도 "이 라이선스에 부합하는 라이선스에 근거하여 컴파일 된 형태나 또는 오브젝트 코드 형태만으로 소프트웨어 부분을 배포하는" 조건일 것이다. 하지만, 좀 더 검토해 보면 다음 상황을 비추고 있다. 바이너리를 배포하는 정황을 뜻하는 유일한 다른 라이선스 조건은 (a) 상표를 남용하지 않고, (b) 기여자에게 특허 소송을 제기하지 않으며, 그리고 (c) 배포된 부분과 관련하여 어떠한 보증 또는 보장을 기대하지 않는 요구사항이다.<sup>856)</sup>

---

854) 자세한 내용 → OSLiC, pp.106 참조

855) Open Source Initiative: MS-PL, 2013, wp. 참조

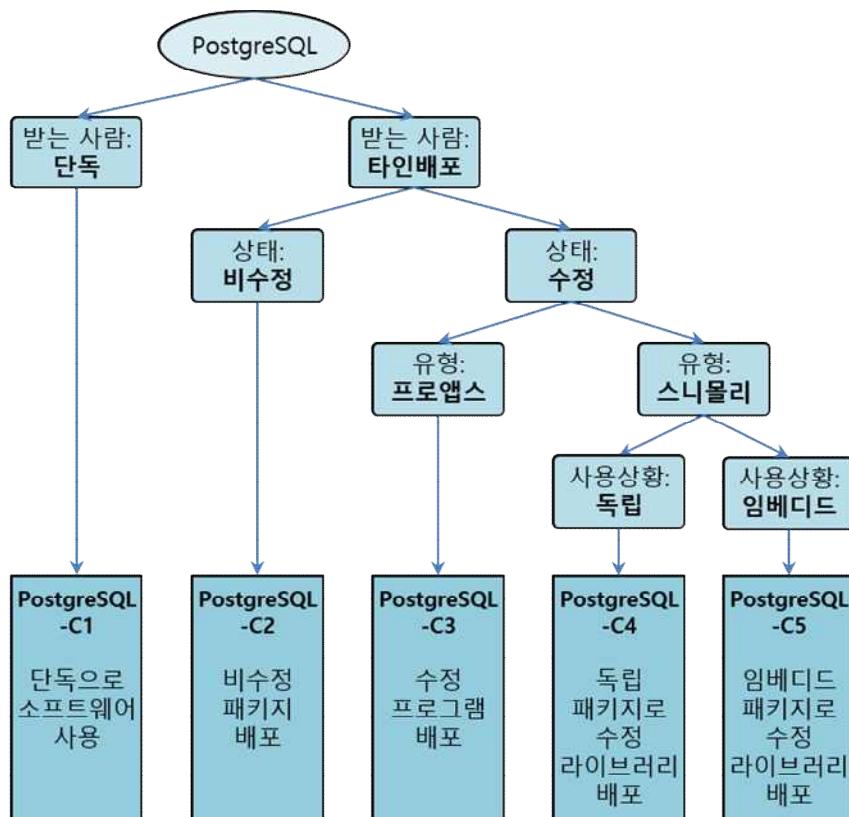
856) 위와 동일, wp. §3A, §3B, §3E. 참조

이러한 해석에 근거하여 다음과 같이 결정하였다.

- 비 수정 버전의 바이너리만을 포함하더라도 배포물에 라이선스 사본을 포함한다. 이를 수정하지 않았다면 라이선스를 추가할 경우 손해를 보지 않는다. 따라서 이는 MSL-PL 바이너리 조건을 충족하는 최선의 방법이다.
- 바이너리를 배포하는 경우에 모든 의무적 조건을 삭제한다. MS-PL의 특허 제한사항은 이미 OSLiC의 MS-PL 특허 부분에 적용되며,<sup>857)</sup> MS-PL 권한 관련 OSLiC 섹션에서 MS-PL<sup>858)</sup> 보증 없음 조항이 포함되는 반면에 상표 제한사항은 금지사항 부분에 확실히 추가되었다.
- 라이브러리에 의해서가 아니라 라이브러리를 사용하는 지배적인 저작물에 의해 일반적으로 저작권 디아일로그가 고안되므로 독립적으로 snimoli를 배포하는 경우에 임의로 업데이트한 저작권 디아일로그의 힌트를 삭제한다.

### 6.13 PostgreSQL 라이선스 소프트웨어

MIT 라이선스와 같이 PostgreSQL 라이선스는 매우 관대한 라이선스다. 따라서 PostgreSQL 상세 파인더를 다음과 간략하게 나타낼 수 있다.



857) → OSLiC, p. 53

858) → OSLiC, p. 39

### 6.13.1 PostgreSQL-C1: 단독으로 소프트웨어 사용

**의미:** PostgreSQL 라이선스가 적용된 소프트웨어를 단독으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>859)</sup>

**요구사항:** PostgreSQL 라이선스 조건을 충족하기 위한 작업은 다음과 같다.

- 제3자에게 소프트웨어를 배포하지 않고 기존의 저작권 고지 및 허가 고지를 수정하지 않는다면, PostgreSQL 라이선스가 적용된 소프트웨어를 어떠한 의무사항 실행 없이 사용할 수 있다.

**금지사항:** 명확한 사항 없음

### 6.13.2 PostgreSQL-C2: 수정하지 않은 소프트웨어 배포

**의미:** PostgreSQL 라이선스가 적용된 소프트웨어를 수정하지 않은 바이너리나 수정하지 않은 소스코드 파일로 제3자에게 배포하는 것을 의미한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그인을 독립 패키지로 배포하는 것과 같은 형태이다.

**적용범위:** OSUC-02S, OSUC-02B, OSUC-05S, OSUC-05B, OSUC-07S, OSUC-07B<sup>860)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 저작권 고지, 허가 고지, PostgreSQL 권리포기 사항을 포함한 PostgreSQL 라이선스를 받았던 형태로 패키지에 포함시킨다.
- [자발적:] 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.

**금지사항:** 명확한 사항 없음

### 6.13.3 PostgreSQL-C3: 수정한 프로그램 배포

**의미:** PostgreSQL 라이선스가 적용된 프로그램, 애플리케이션, 서버(proapse)를 수정한 뒤 수정된 버전을 바이너리나 소스코드 파일 형식으로 제3자에게 배포하는 의미이다.

**적용범위:** OSUC-04S, OSUC-04B<sup>861)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 저작권 고지, 허가 고지, PostgreSQL 권리포기 사항을 포함한 PostgreSQL 라이선스를

859) 자세한 내용 → OSLiC, pp.96-105 참조

860) 자세한 내용 → OSLiC, pp.96-102 참조

861) 자세한 내용 → OSLiC, pp.98-99 참조

받았던 형태로 패키지에 포함시킨다.

- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 자신의 저작물 또는 수정에 관한 정보를 기준 저작권 내용에 추가할 수 있다.
- [자발적:] 프로그램 실행과 함께 PostgreSQL 라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 이미 프로그램을 수정하였기 때문에 초기 저작권 고지에 그 설명이 결여되어 있다면 그런 힌트를 추가할 수 있다.

**금지사항:** 명확한 사항 없음

#### 6.13.4 PostgreSQL-C4: 수정한 라이브러리를 독립적으로 배포

**의미:** PostgreSQL로 라이선스 된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 제3자에게 바이너리나 소스코드 파일 형태로 배포한다. 이 경우 바이너리 및 소스코드 파일로 배포하는 것과 관계없이 스니펫, 모듈, 라이브러리, 플러그 포함한 더 큰 소프트웨어 유닛을 함께 배포한다.

**적용범위:** OSUC-08S, OSUC-08B<sup>862)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 저작권 고지, 허가 고지, PostgreSQL 권리포기 사항을 포함한 PostgreSQL 라이선스를 받았던 형태로 패키지에 포함시킨다.
- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.

**금지사항:** 명확한 사항 없음

#### 6.13.5 PostgreSQL-C5: 수정한 라이브러리를 임베디드 컴포넌트로 배포

**의미:** PostgreSQL로 라이선스 된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 제3자에게 바이너리나 소스코드 파일 형태로 배포한다.

이 경우 바이너리 및 소스코드 파일로 배포하는 것과 관계없이 스니펫, 모듈, 라이브러리, 플러그 포함한 더 큰 소프트웨어 유닛을 함께 배포한다.

**적용범위:** OSUC-10S, OSUC-10B<sup>863)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

---

862) 자세한 내용 → OSLiC, pp.103 참조

863) 자세한 내용 → OSLiC, pp.105-106 참조

- [필수적:] 저작권 고지, 허가 고지, PostgreSQL 권리포기 사항을 포함한 PostgreSQL 라이선스를 받았던 형태로 패키지에 포함시킨다.
- [자발적:] 배포를 하지 않더라도 소스코드 내 수정사항을 표시한다.
- [자발적:] 프로그램 실행과 함께 PostgreSQL 라이선스 하에 프로그램이 배포되고 있다고 표시하는 행위는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 임베디드 컴포넌트의 다운로드 페이지 및 홈페이지에 링크를 추가한다.
- [자발적:] 배포 문서 및 추가 자료에 초기 소프트웨어(프로젝트)와 그 홈페이지 링크를 포함시킨다.
- [자발적:] 초기 라이선싱 요소(원저작자의 저작권 고지, 허가 고지, PostgreSQL 권리포기 사항이 포함된 PostgreSQL 라이선스 텍스트)는 임베디드 라이브러리만을 명확히 참조하고 자신이 보유한 지배적인 저작물의 라이선싱을 침해하지 않도록 배포물을 구성한다. 라이브러리, 모듈, 스티커, 플러그인을 모든 라이선싱 구성요소를 포함한 디렉토리에 포함시키는 것을 권장한다.

**금지사항:** 명확한 사항 없음

#### 6.13.6 논의 및 설명

PostgreSQL 라이선스는 MIT 라이선스 구조를 따른다. 또한 (1) 저작권 고지, (2) 하려는 모든 행위가 거의 허용된다는 절의 내용을 포함하고, 그 다음에 (3) “모든 사본에” 저작권 고지와 허가 고지 및 권리포기 사항이 “표시”된다는 조건이 이어지며, 그리고 (4) 잘 알려진 권리포기 내용으로 종결된다.<sup>864)</sup> 그러나 MIT 라이선스와 같이 PostgreSQL 라이선스에서는 소스코드와 목적코드의 차이에 대해 언급하지 않는다. 따라서 MIT에 대한 분석을 PostgreSQL 라이선스에 적용할 수 있다.<sup>865)</sup>

#### 6.14 PHP-3.0 라이선스 소프트웨어

PHP-3.0 라이선스는 MIT 라이선스보다 많은 조건을 포함하고, “소스코드 재배포”<sup>866)</sup>와 “바이너리 형태로 재배포를” 구분 짓는다.<sup>867)</sup> PHP-3.0 라이선스는 재배포 또는 OSLiC에서 언급하려는 바대로 타인배포 사용사례에만 중점을 둔다. 따라서 PHP-3.0 상세 파인더는 다음과 같이 나타낼 수 있다.

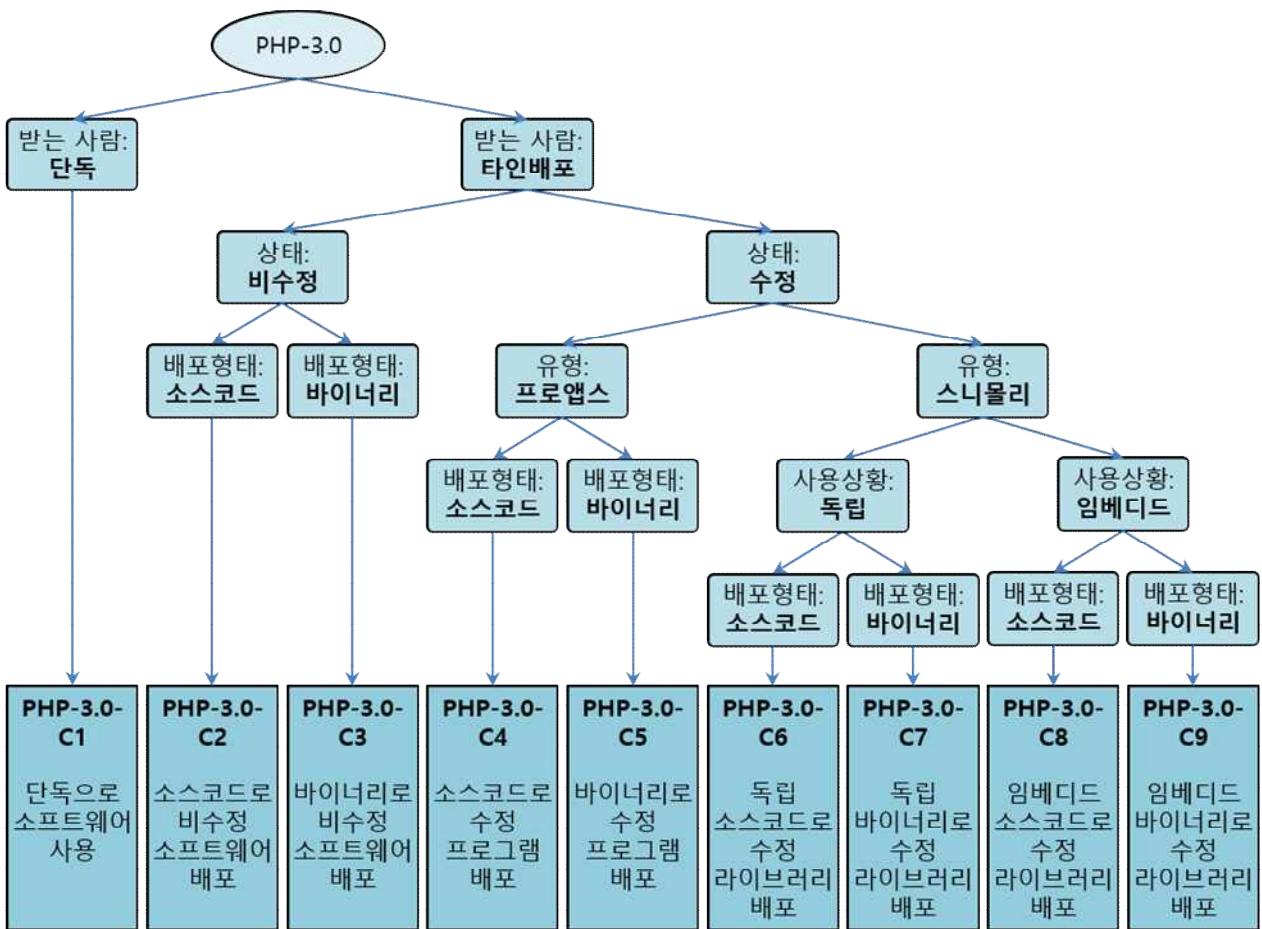
---

864) Open Source Initiative: PostgreSQL License, 2013, wp. 참조

865) → OSLiC, p.223

866) Open Source Initiative: PHP-3.0, 2013, wp 참조

867) 위와 동일.



#### 6.14.1 PHP-3.0-C1: 단독으로 소프트웨어 사용

**의미:** PHP-3.0 라이선스가 적용된 소프트웨어를 단독으로 사용하고 제3자에게 양도하지 않는다.

**적용범위:** OSUC-01, OSUC-03L, OSUC-03N, OSUC-06L, OSUC-06N, OSUC-09L, OSUC-09N<sup>868)</sup>

**요구사항:** 사용 사례와 관련하여 PHP-3.0 라이선스 조건을 충족하기 위한 작업은 없다.

- 제3자에게 소프트웨어를 제공하지 않는 한 PHP-3.0 라이선스가 적용된 소프트웨어를 어떠한 의무사항 실행 없이 사용할 수 있다.

**금지사항:** 'PHP'라는 이름으로 서비스를 보증하거나 홍보하는 행위

#### 6.14.2 PHP-3.0-C2: 수정하지 않은 소프트웨어를 소스코드 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 소프트웨어를 수정되지 않은 소스코드 파일이나 수정하지 않은 소스코드 패키지를 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그인을 독립되거나 임베디드 유닛으로 배포하는 형태와 동일한 형태이다.

868) 자세한 내용 → OSLiC, pp.96-105 참조

**적용범위:** OSUC-02S, OSUC-05S, OSUC-07S<sup>869)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기 사항)를 받았던 형태로 패키지에 포함한다.
- [필수적:] 배포 문서 및 추가 자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [자발적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

#### 6.14.3 PHP-3.0-C3: 수정하지 않은 소프트웨어를 바이너리 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 소프트웨어를 수정되지 않은 바이너리 파일이나 수정되지 않은 바이너리 패키지를 제3자에게 배포한다. 이 경우 프로그램, 애플리케이션, 서버, 스니펫, 모듈, 라이브러리, 플러그를 독립되거나 임베디드 유닛으로 배포하는 것은 문제되지 않는다.

**적용범위:** OSUC-02B, OSUC-05B, OSUC-07B<sup>870)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기)는 받았던 형태로 패키지를 통해 재 생성한다. 소스코드 패키지를 바이너리 파일로 컴파일하고 이 컴파일을 통해 라이선싱 파일을 생성 및 통합하지 않는다면, 저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 소스코드 패키지에 생성한 후 배포물에 수동으로 이 파일들을 삽입한다.<sup>871)872)</sup>
- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [자발적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

869) 자세한 내용 → OSLiC, pp.96-102 참조

870) 자세한 내용 → OSLiC, pp.96-102 참조

871) 비 수정 라이브러리를 배포하는 중이므로 애플리케이션의 저작권 화면을 통한 필수 작업이 무엇인지 추정할 수 있을 것이다.

872) 파일에 대한 양도를 올바르게 수행 → OSLiC, p.107

#### 6.14.4 PHP-3.0-C4: 수정한 프로그램을 소스코드 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 프로그램, 애플리케이션, 서버(proapse)의 수정된 버전을 소스코드 파일로 또는 소스코드 패키지로 제3자에게 배포한다.

**적용범위:** OSUC-04S<sup>873)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기 사항)를 받았던 형태로 패키지에 포함한다.
- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [자발적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] 프로그램 실행과 함께 저작권 고지에 대한 표시는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 이미 프로그램을 수정하였기 때문에 저작권 고지 원본에 그 설명이 결여되어 있다면 그런 힌트를 추가할 수 있다. 만약, 저작권 스크린에 고지가 확인되지 않는다면 저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기를 포함하는 PHP-3.0 라이선스 사본을 재 생성하도록 고려한다. (바이너리 배포 시 필요함)<sup>874)</sup>
- [자발적:] 소스코드에 수정한 부분을 표시한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

#### 6.14.5 PHP-3.0-C5: 수정한 프로그램을 바이너리 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 프로그램, 애플리케이션, 서버(proapse)의 수정된 버전을 바이너리 파일로 또는 바이너리 패키지로 제3자에게 배포한다.

**적용범위:** OSUC-04B<sup>875)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [필수적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기)는 받았던 형태로 패

873) 자세한 내용 → OSLiC, pp.98 참조

874) 컴파일된 버전의 배포자를 따르면 사용자의 저작물의 가치가 인정된다.

875) 자세한 내용 → OSLiC, pp.99 참조

키지를 통해 재 생성한다. 소스코드 패키지를 바이너리 파일로 컴파일하고 이 컴파일을 통해 라이선싱 파일을 생성 및 통합하지 않는다면, 저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 소스코드 패키지에 생성한 후 배포물에 수동으로 이 파일들을 삽입한다.

- [자발적:] 코드를 배포하지 않더라도 소스코드에 수정한 부분을 표시한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

#### 6.14.6 PHP-3.0-C6: 수정한 라이브러리를 독립적인 소스코드 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 소스코드 파일이나 소스코드 패키지 형태로 제3자에게 배포하는 형태이며, 다른 더 큰 소프트웨어 유닛에 포함하지 않는 형태이다.

**적용범위:** OSUC-08S<sup>876)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기 사항)를 받았던 형태로 패키지에 포함한다
- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [자발적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] 소스코드에 수정한 부분을 표시한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

#### 6.14.7 PHP-3.0-C7: 수정한 라이브러리를 독립적인 바이너리 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 바이너리 파일이나 바이너리 패키지 형태로 제3자에게 배포하는 형태이며, 다른 더 큰 소프트웨어 유닛에 포함하지 않는 형태이다.

**적용범위:** OSUC-08B<sup>877)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.

876) 자세한 내용 → OSLiC, pp.103 참조

877) 자세한 내용 → OSLiC, pp.103 참조

- [필수적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기)는 받았던 형태로 패키지를 통해 재 생성한다. 소스코드 패키지를 바이너리 파일로 컴파일하고 이 컴파일을 통해 라이선싱 파일을 생성 및 통합하지 않는다면, 저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 소스코드 패키지에 생성한 후 배포물에 수동으로 이 파일들을 삽입한다.
- [자발적:] 코드를 배포하지 않더라도 소스코드에 수정한 부분을 표시한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

#### 6.14.8 PHP-3.0-C8: 수정한 라이브러리를 임베디드 소스코드 형태로 배포

**의미:** PHP-3.0 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 소스코드 파일이나 소스코드 패키지 형태로 제3자에게 배포하는 형태이며, 스니펫, 모듈, 라이브러리, 플러그인을 임베디드 컴포넌트 형태로 포함한 더 큰 소프트웨어 유닛과 함께 배포하는 형태다.

**적용범위:** OSUC-10S<sup>878)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기 사항)를 받았던 형태로 패키지에 포함한다.
- [필수적:] 배포 문서 및 추가자료에 "이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>"라는 형태로 승인 문구를 포함한다.
- [필수적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] 프로그램 실행과 함께 저작권 고지에 대한 표시는 오픈소스 커뮤니티에서 바람직한 행위로 정의한다. 이미 프로그램을 수정하였기 때문에 저작권 고지 원본에 그 설명이 결여되어 있다면 그런 힌트를 추가할 수 있다. 만약, 저작권 스크린에 고지가 확인되지 않는다면 저작권 고지, PHP-3.0 조건들, PHP-3.0 권리포기를 포함하는 PHP-3.0 라이선스 사본을 재 생성하도록 고려한다. (바이너리 배포 시 필요함)<sup>879)</sup>
- [자발적:] 소스코드에 수정한 부분을 표시한다.
- [자발적:] 초기 라이선싱 요소(PHP-3.0 라이선스 텍스트, 원 저작권자의 저작권 고지, PHP-3.0 권리보기)는 임베디드 라이브러리만을 명확히 참조하고 자신이 보유한 지배적인 저작물의 라이선싱을 침해하지 않도록 배포물을 구성한다. 라이브러리, 모듈, 단편 코드, 플러그인을 모든 라이선싱 구성요소를 포함한 디렉토리에 포함시키는 것을 권장한다.

**금지사항:** 'PHP'라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다.

878) 자세한 내용 → OSLiC, pp.105 참조

879) 컴파일된 버전의 배포자를 따르면 사용자의 저작물의 가치가 인정된다.

#### 6.14.9 PHP-3.0-C9: 수정한 라이브러리를 임베디드 바이너리로 배포

**의미:** PHP-3.0 라이선스가 적용된 스니펫, 모듈, 라이브러리, 플러그인(snimoli)을 수정한 뒤 수정된 버전을 바이너리 파일이나 바이너리 패키지 형태로 제3자에게 배포하는 형태이며, 스니펫, 모듈, 라이브러리, 플러그인을 임베디드 컴포넌트 형태로 포함한 더 큰 소프트웨어 유닛과 함께 배포하는 형태다.

**적용범위:** OSUC-10B<sup>880)</sup>

**요구사항:** 라이선스 조건을 충족하기 위한 의무사항은 다음과 같다.

- [필수적:] 배포 문서 및 추가자료에 “이 제품은 다음 주소에서 자유로이 이용할 수 있는 PHP를 포함하다. <http://www.php.net/>”라는 형태로 승인 문구를 포함한다.
- [필수적:] 배포 문서 및 추가 자료에 또한 저작권 고지 원본, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 포함한다.
- [자발적:] PHP-3.0 라이선스(저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기)는 받았던 형태로 패키지를 통해 재 생성한다. 소스코드 패키지를 바이너리 파일로 컴파일하고 이 컴파일을 통해 라이선싱 파일을 생성 및 통합하지 않는다면, 저작권 고지, PHP-3.0 조건 및 PHP-3.0 권리포기 사항을 소스코드 패키지에 생성한 후 배포물에 수동으로 이 파일들을 삽입한다.
- [자발적:] 코드를 배포하지 않더라도 소스코드에 수정한 부분을 표시한다.
- [자발적:] 라이선싱 요소(PHP-3.0 라이선스 텍스트, 원 저작권자의 저작권 고지, PHP-3.0 권리보기)는 임베디드 라이브러리만을 명확히 참조하고 자신이 보유한 지배적인 저작물의 라이선싱을 침해하지 않도록 배포물을 구성한다. 라이브러리, 모듈, 단편 코드, 플러그인을 모든 라이선싱 구성요소를 포함한 디렉토리에 포함시키는 것을 권장한다.

**금지사항:** ‘PHP’라는 명칭을 이용하여 제품을 홍보하거나 지지하는 행위를 금지한다

#### 6.14.10 논의 및 설명

OLiC에서는 PHP-3.0 라이선스에 대해 다루면서도 바이너리 파일 배포와 연관되는 오픈소스 사용 사례에 대해서도 설명한다. PHP는 스크립트 언어이다. 따라서 소스코드 전달은 반드시 이행해야 하는 행위로 보인다. 그러나 PHP-3.0 라이선스는 컴파일러 패러다임 위에 구성된 다른 언어로 작성된 저작물에도 적용할 수 있다. 또는 PHP 컴파일러가 사용될 수 있다.

수정된 바이너리 배포의 경우 문서 내 PHP-3.0 라이선스를 복제하는 조건은 반드시 해야 할 당위적(must) 행위이지만 프로그램의 저작권 화면 내 복제는 의무적(should) 행위이다. 이는 “바이너리 형태로 재배포 시 문서 및 배포물과 함께 제공되는 자료에 위의 저작권 고지, 조건 리스트 및 권리포기 사항을 복제해야 한다”<sup>881)</sup>라고 요구하고 있는 PHP 라이선스의 바이너리 조건에 의해 직접적인 영향을 받는다. 그러나 당위적 행위와 의무적 행위 둘 다 이행하는 것이 가장 바람직하다.

880) 자세한 내용 → OSLiC, pp.106 참조

881) Open Source Initiative: PHP-3.0, 2013, wp. 참조

## 7 결론

우리는 지난 4년 동안 OSLiC(Open Source License Compendium)를 개발하였다. European Legal 및 FSFE(Free Software Foundation European)의 해 구성된 라이선싱 워크샵에 참여하고 있는 많은 오픈소스 전문가들과 함께 우리의 아이디어에 대해 토론한 것은 영광스럽고 기쁜 일이다. 독일, 아시아, 유럽에서 개최되고 있는 컨퍼런스에서 우리의 결과물을 발표하였다. 우리는 여기서 매우 고무적인 피드백을 얻었다. 우리가 처음 시작할 때 원했던 것을 우리는 알게 되었다. 오픈소스 소프트웨어 사용을 준수하기 위해 무엇을 이행해야 하는지 간단하고 믿을만한 방법을 제공함으로써, 오픈소스 사용에 대한 진입장벽을 줄일 수 있었다. 우리는 긴 길을 걸어온 것에 대해 자랑스럽게 여긴다. 그리고 공식적으로 OSI에 의해 OSLiC가 오픈소스 컴플라이언스를 관리하기 위한 하나의 툴로 선정된 것에 대해 자랑스럽게 여긴다.<sup>882)</sup>

그러나 우리는 또한 여러 피드백 의견을 조정하고 있다. 커뮤니티에 의해 우리의 초기 전제가 정당하게 받아들여지지 않았다. 우리는 소프트웨어 개발자 자신들은 OSLiC를 절대 사용하지 않을 것이라고 설명하였다. 이 책이 컴퓨터 언어의 상세내역을 포함하지 않는 한 그들은 리스트 및 테이블로 된 300페이지 이상의 책을 절대 읽지 않을 것이다. OSLiC는 오픈소스를 일반적으로 사용하는 유저에게는 너무 광범위하고 복잡할 것이다. 오픈소스 라이선스 전문가가 되기 위한 훌륭한 기초가 될 것이지만 손쉬운 도구는 아니다. 실제로 소프트웨어 개발자, 프로젝트 매니저, IT 매니저가 OSLiC를 직접 이해하고 사용할 수 있다고 가정한 것은 어리석다고 생각한다. OSLiC를 처음 읽게 되면 충격적인 현상을 받아들이게 될 것이고, 현재 우리 또한 그것을 알고 있다.

그렇지만, 우리가 이러한 환상의 매력에 빠진 것은 매우 중요했다. 이러한 오류가 없었으면 우리는 OSLiC의 개발을 시작하지 않았을 것이다. 그리하여 우리는 파인더 형식과 5개 질문 형식에 이슈들을 나열하기 위한 방법을 결코 찾지 못했을 것이다. 이러한 오류 없이 우리는 라이선스와 전문가를 인용하여 각 단일 주장을 정당화 하고 입증하는 작업을 실행하지 못했을 것이다. 우리가 받은 놀라운 피드백 없이 우리는 최고의 경험과 고무적인 경험을 얻지 못했을 것이다.

우리가 피드백을 받아 들였을 때 우리는 온라인 버전의 OSLiC를 개발할 것을 결정하였으며 OSCAd(Open Source Compliance Advisor)는 OS-CAd로 알려져 있다.<sup>883)</sup> 우리는 AGPL 라이선스를 적용하여 배포하였다. 그리고, Amadeus는 온라인 툴의 개발에 대한 권한을 넘겨받는 것을 결정하였다. 우리는 OSLiC에 추출 인터페이스를 추가하였으며, Amadeus는 OSCAd에 이를 재개발하여 삽입 인터페이스를 추가하였다.

이러한 추가로 우리는 하나의 특정 작업에 집중할 수 있었다. Amadeus는 온라인 툴<sup>884)</sup>의 유지 보수 및 서비스를 담당하였으며, 우리는 오픈소스 라이선스<sup>885)</sup> 기초 분석을 담당하게 되었다.

---

882) → <http://osi.xwiki.com/bin/Projects/Process+and+Compliance+Resources>

883) → <http://opensource.telekom.net/oscad/>

884) → <https://github.com/AmadeusITGroup/oscad>

885) → <https://github.com/dtag-dbu/oslic/>

따라서 우리는 중요한 자원 및 아이디어 공유의 장점을 얻게 되었다. 오픈소스를 활용하게 되면 작은 부분을 제공하지만 많은 부분을 얻을 수 있으며, 자신의 리소스를 절약할 수 있다.

따라서 우리는 사용자가 확인할 수 있는 OSLiC 버전 1.0을 출간하였으며, 오픈소스 커뮤니티에 고개 숙여 깊은 감사를 표한다.

## 8 부록

### 8.1 OSLiC 인용 스타일에 대한 몇 가지 추가 설명

우리는 이미 각주의 일반적인 방법에 대해 정의하였다.<sup>886)</sup> 다음은 참고문헌에 대한 간단한 설명이다. :

현대는 인문학을 변화시켰다. 이전에는 책이나 기사는 전달을 용이하게 하기 위해 반드시 출판되어야 했다. 우리의 성명서는 잘 준비된 정적인 작업에 의존했다. 심지어 요즘 대학 도서관들은 일부 원본으로 출판된 PDF파일로서 책이나 기사를 제공한다. 학자로서 당신은 출판된 버전과 PDF파일은 최소한 페이지 번호와 모양 면에서 동일하다는 것에 의존해야 한다. 하지만 당신은 그 동일성을 확인할 수 없다.

더욱이, '전자책'과 '전자기사'와 같은 경우 종종 도서관에서 PDF 파일 자체를 제공하지 않을 수 있지만 출판자의 다운로드 페이지 링크는 제공한다. 이전 학자로서 당신은 당신의 독자가 인용된 내용의 출처를 확인하기 원한다면 인용한 저작물을 검색할 수 있을 것이라고 믿을 수 있다. 과학적인 출처를 확보하는 것이 우리의 도서관의 하나의 임무이다. 하지만, 현재 독자들은 더 이상 책을 구매하지 않을 뿐만 아니라 대학교 인터넷 망을 통해 파일을 다운로드 할 권한을 갖는다. 이 경우 PDF 파일은 대학 도서관의 서버에 저장되지 않는다. 경우에 따라 각 학생들 혹은 독자들은 게시자에 의해 제공된 링크를 사용하여 자신의 파일을 다운로드 한다. 따라서 학자로서 당신은 링크를 제공한 게시자가 당신이 인용한 PDF파일을 변경하지 않을 것이라는 것을 믿어야 한다.

그러나 이러한 현상은 더욱 악화될 수 있다. 게시자들이 자신의 작업을 비밀리에 수정될 수 있지만 (심지어 그렇게 할 확률도 적다.), 웹 페이지가 자주 변경된다는 것은 명확한 특징이다. 그러므로 우리는 사라질 수 있는 진술과 문서에 근거하여 진지하게 논할 필요가 있는지, 그러한 불안정한 출처를 인용할 수 있는지, 등에 대해 스스로에게 질문해야 한다. 문제는 : 우리가 특히 인터넷 주제에 대해 글을 쓸 때 그리고 진정으로 신뢰할 만한 해설서를 쓰고 싶을지라도 그렇게 해야만 한다는 것이다.

그래서 우리는 무엇을 할 수 있을까? 첫째, 독자들이 우리가 인용하고 작성한 내용을 찾을 수 없을 경우 독자들이 우리의 원본 자료를 획득할 수 있도록 한다. 둘째, 우리는 우리가 읽은<sup>887)</sup> 모든 전자 문서<sup>888)</sup> 들을 보관한다. 셋째, 독자들에게 원본자료를 서로 다른 수준의 믿을만한 레벨로 공개한다. 따라서 우리는 우리의 독서 목록 데이터에 다음과 같은 부호를 사용한다.<sup>889)</sup>

- 인쇄/복사: 원본자료는 출판되어야 되며 우리는 출판된 사본을 확인하거나 공식적인 사본을 도서

886) → p. 14

887) 저작권 문제로 우리가 인용한 내용을 확인하기 위한 다운로드 링크를 제공 및 복사본의 제공은 허용되지 않는다.

888) 'e-book', 'e-article', 'e-paper' 의 (새로운) 일반화

889) 또 다른 힌트 : 요즘 과학 도서관 조차도 원본과 동일한 '전자사본'을 제공하지 않는다. 어떠한 경우에는 이전에 저널의 일부로 출간된 기사의 HTML 버전만을 확인할 수 있다. 이 경우 학자는 원본 자료의 페이지가 없는 출처를 사용해야한다. 현재 내부 페이지가 포함된 PDF 파일로만 제공되는 경우 동일한 결과가 발생될 수 있다. 이러한 종류의 기사를 인용할 경우 기사의 원래 행에 인용한 기사의 숫자를 정의하기 위해 가능한 내부 페이지 번호로 추가한다. 그러나 우리는 당연히 이러한 종류의 사본을 배포한 조직에 의해 제공된 독서목록 데이터를 따르려고 한다.

관에서 얻을 수 있다. 따라서 적어도 우리가 이용한 도서관(UB Frankfurt & ULB Darmstadt)에서 사본을 확인할 수 있을 것이다.

- BibWeb/[PDF/...]: 원본자료는 출판되어야 하지만, 우리는 대학 도서관(UB Frankfurt & ULB Darmstadt)의 네트워크 망에서 제공된 전자문서(PDF 혹은 다른 형식의 파일)만 읽었다.
- FreeWeb/[PDF/...]: 우리는 무료 웹 사이트에서 제공하는 전자문서를 읽었다. 이 경우 다운로드 되거나 텍스트를 확인할 때 URL<sup>890</sup>과 날짜를 추가한다.

## 8.2 널리 알려진 오픈소스의 근거 없는 믿음

인터넷 학생 관점으로 웹은 오픈소스 소프트웨어(라이선스)와 관련 강력한 루머를 제공하는 것을 고려해야 한다. 다음은 우리가 확인한 몇 가지의 오픈소스에 대한 근거 없는 믿음<sup>891</sup>이다.

**주의사항:** 이 세션은 반드시 검토 및 재출간되어야 한다. 이 세션의 내용은 개요수준의 내용이며, 인용을 금지하며 반드시 검증되어야 한다.

**오픈소스는 윤리적으로 세계를 개선하기 위해 노력한다.** – 아니다. 사람, 그룹, 목적을 배제에 대한 명백한 금지가 있다. 따라서 오픈소스 소프트웨어를 사용하는 누군가를 윤리적 또는 도덕적, 악의적인 이유로 배제할 수 없다.

**변경된 오픈소스는 반드시 재공개되어야 한다.** – 아니다. 이중의 의미로서, 사용자에 맞추어 코드를 수정하여 독점화 할 수 있도록 허용한 OS 라이선스가 있다. 그리고 명백하게 독점화를 허용하지 않는 LGPL와 GPL 조차도 일반적으로 수정된 코드를 반드시 재공개해야 한다고 요구하지는 않는다. 당신이 수정한 (L)GPL 라이선스가 적용된 애플리케이션을 다른 사람에게 바이너리로 제공하는 경우에만 수정된 코드를 반드시 제공해야 한다.

**수정된 오픈소스 소프트웨어는 반드시 전체 커뮤니티에 반환해야 한다.** – 아니다. 사용자에 맞추어 코드를 수정하여 독점화할 수 있도록 허용한 OS 라이선스가 있다. 심지어 수정된 코드 또한 제공해야 하는 것을 분명히 요구하고 있는 LGPL와 GPL도 수정된 바이너리를 누군가에게 제공한다면, 전 세계에 당신의 수정본을 배포하라고 요구하지 않는다. LGPL와 GPL은 당신이 바이너리를 제공하는 사람에게 코드를 제공해야 한다고 분명히 제시하고 있다. 만약 당신이 한 명 또는 한 그룹의 사람에게만 개선 사항을 제공한다면 당신은 해당 그룹의 모든 구성원들에게만 혹은 해당 수령자에게만 당신의 코드를 양도해야 한다.

---

890) 노트: 긴 URL은 라인에 맞추기 어렵기 때문에 텍스트를 모양을 변형시킬 수 있다. 따라서 이 작업을 실행하기 위해 LaTeX를 쉽게 만들어야 한다. URL을 때로는 분할하거나 빙칸을 삽입해야 한다. 따라서 URL을 검증하기 위해 때로는 빙칸을 제거해야 한다.

891) 과학적 합법적으로 논의하는 책은 적어도 “오픈소스 라이선스에 대한 신화”와 관련하여 적어도 한번은 다루어야 한다. cf Guibault, Lucie a. Ot van Daalen : Unravelling the Myth around Open Source Licenses. An Analysis from A Dutch and European Law Perspective: The Hague: T. M. C. Asser Press, 2006 (= IT & Law, [Vol./No.] 8), ISBN 978-90-6704-214-7, pp. 1ff, especially 209ff.

**배포된 오픈소스 소프트웨어는 영원히 공개되어야 한다.** – 아니다. 모든 향후 버전들은 반드시 오픈소스 라이선스가 적용되어 배포되어야 한다는 것은 틀린 말이다. 저작권자는 저작권을 영원히 소유하게 된다. 저작권자는 다음 소프트웨어 버전에 대한 라이선스를 변경할 수 있지만, 현재 및 이전 버전의 라이선스에는 적용되지 않는다. 이미 오픈소스로 공개된 소프트웨어는 오픈소스로 남겨져 있어야 한다.

**소프트웨어는 오픈소스 소프트웨어 또는 독점 소프트웨어가 될 수 있다.** – 아니다. 저작권자는 그들이 원하는 대로 다른 조건들을 코드에 부과하여 배포할 수 있다. 이는 라이선스 문제가 아니라 저작권의 문제이다.

**오픈소스의 반대는 상용 소프트웨어이다.** – 아니다. 첫째, 당신은 상업적 목적으로 오픈소스 소프트웨어를 사용할 수 있다. 오픈소스 소프트웨어에 포함되지 않은 유일한 점이 있는데 그것은 '오픈소스 소프트웨어'를 배포한다면 라이선스 비용을 청구할 수 없다는 것이다. 둘째, 오픈소스 소프트웨어도 상용소프트웨어도 아닌 다른 형태 혹은 공개 도메인 소프트웨어(Public Domain Software), 자유 소프트웨어와 같이 다른 많은 형태의 소프트웨어 분류가 있다. 오픈소스 소프트웨어의 반대를 구별하는 기준으로 수익에 관한 질문은 무의미하다. 더욱이, 오픈소스 소프트웨어와 반대되는 논리로서 상용 소프트웨어를 OSD와 맞지 않는 모든 종류의 소프트웨어는 독점소프트웨어이다라고 단정짓는 것은 부정적으로 논리이다 :

**오픈소스 소프트웨어는 수익 창출을 금지한다.** – 아니다. 당신은 당신이 원하는 비즈니스 모델을 개발 할 수 있지만 단 하나의 예외가 있다. '오픈소스 소프트웨어'를 배포한다면 라이선스 비용을 청구할 수 없다는 것이다. 이 제한은 라이선스에 명확하게 정의되어 있는 오픈소스 정의에 근거한다. 오픈소스 라이선스가 되기를 원하는 라이선스에서는 "여러 다른 소스를 이용하여 작성한 프로그램을 포함하는 결합 소프트웨어 배포의 구성요소로서 소프트웨어를 판매하거나 양도하는 것을 제한하지 않는다." 그리고 "이러한 상황 하에서의 라이선스는 그러한 판매를 위해 기타 수수료 혹은 로얄티를 요구해서는 안 된다"<sup>892)</sup>라고 하는 것을 분명히 제시하고 있다. 이러한 제약 조건을 "누구에게나 프로그램의 사용을 제한하면 안 된다."<sup>893)</sup> 또한, 컴파일된 형식뿐 만 아니라 소스코드로 배포를 허용해야 한다.<sup>894)</sup>라고 하는 오픈소스 라이선스가 요구하는 요구사항과 결합한다면 오픈소스 라이선스는 배포나 사용에 있어 비용을 요구하지 않는다고 결론을 내릴 수 있다. 그러나, 프로그램 인스톨을 위한 서비스, 특정 버전의 컴파일 및 수집, 환경에 대한 모니터는 비용을 요구할 수 있다.

역사적으로 이러한 근거 없는 믿음은 Debian에 의해 제기될 수 있다. GNU 프로젝트에는 커널을 포함하지 못한 반면 리눅스 커널은 이미 GNU 소프트웨어가 포함된 컬렉션의 일부로 배포되었다. 그리고 1983(?)년에 Ian Murdock은 RMS와 FSF의 지원을 받아 GNU 소프트웨어와 리눅스 커널이 포함된 무료 배포판(Debian)을 배포하였다. 그러나 Ian Murdock은 Debian은 돈을 벌고 싶어 하지 않는다고 하였다.

**오픈소스 소프트웨어의 수정은 반드시 표시해야 한다.** – 아니다. 이는 OSD에 의한 정의가 아니다. OSD는 라이선스에 수정 부분의 표시를 요구하는 것을 허용한다. 그러나 모든 라이선스가 오픈소스 라이선스가

---

892) Open Source Initiative: The Open Source Definition, 2012, §1 참조

893) 위와 동일, §6. 참조

894) 위와 동일, §2 참조

되기 위해 수정 표시를 요구하지 않는다.

**오픈소스 소프트웨어의 수정은 당신의 개인 데이터에 의해 표시되어야 한다.** – 아니다. 사용자가 수정 코드를 원본 코드와 구분할 수 있도록 수정사항을 표시하기만 하면 된다. 이것은 원래 작성자 코드의 무결성을 유지하는데 필요하다. 따라서 이 부분은 OSD를 구성하는 기준으로 요구되지 않는다. 어떤 라이선스는 당신의 이름을 추가적으로 요구할 수 있다. 하지만, 이는 오픈소스 소프트웨어의 일반적인 특징은 아니다. 적어도 우리가 논의한 라이선스는 당신의 이름을 삽입할 필요가 없다.

**오픈소스 정의(Open Source Definition)는 오픈소스 소프트웨어의 사용 조건을 결정한다.** – 아니다. 오픈소스 정의는 어떠한 라이선스가 오픈소스 라이선스인지만을 결정한다. OSD는 오픈소스 라이선스가 되기 위한 일련의 조건들의 집합이다. 그것은 다소 추상적인 규칙 집합으로 사용자와 자유와 책임을 결정한다. 그러나 OSD는 오픈소스 라이선스를 충족하기 위해 사용자가 수행해야 할 충분한 활동내역을 구성하지는 않는다. 오픈소스 라이선스는 OSD 기준을 구체적인 예를 들어 설명함에 있어 다를 수 있다. 라이선스를 충족하기 위해 무엇을 이행해야 하는지 알고 싶다면 사용하고 있는 소프트웨어의 진짜 라이선스 확인이 필요하다.

이 세션은 왜 우리는 *OSLiC*가 필요하고 어떻게 콘텐츠 및 양식을 필요에 따라 구성할지에 대한 우리 자신의 질문에 대해 초점을 맞춘 아웃라인이다.

### 8.2.1 이유

우리는 오픈소스 대한 다른 책이 필요한가? 당신은 오픈소스 소프트웨어와 관련하여 다른 책이 필요한가? 우리가 이미 알고 있는 것, 우리가 본능적으로 믿는 것, 그리고 우리가 들은 것들에 대한 관점에서 이 질문을 다루도록 하겠다. 예를 들어 당신은 다음 문장 중 하나 이상이 올바르다고 가정할 수 있다. 혹은 동료 또는 관리자와 유사한 인식을 경험했을 수 있다. 혹은 아래 사항들이 “오픈소스”를 설명하고 있다고 들어왔다.

- OSD는 오픈소스 소프트웨어를 사용하기 위한 규칙을 제공한다.
- 수정된 오픈소스 소프트웨어는 반드시 배포되어야 한다.
- 수정된 오픈소스 소프트웨어는 반드시 커뮤니티에 반환되어야 한다.
- 오픈소스 소프트웨어의 모든 버전은 반드시 공개된 채로 남아있어야 한다.
- 소프트웨어는 오픈소스 소프트웨어 및 독점 소프트웨어가 될 수 있다.
- 오픈소스 소프트웨어의 반대는 상용 소프트웨어이다.
- 오픈소스 소프트웨어는 수익 창출을 금지한다.
- 오픈소스 소프트웨어의 수정은 명시적으로 표시 되어야 한다.
- 오픈소스 소프트웨어의 수정자는 자신을 식별해야 한다.
- 오픈소스 바이너리를 배포할 때 소스코드를 다운받는 링크를 제공하는 것으로 충분하다.
- 오픈소스 소프트웨어의 목적은 세계를 윤리적으로 개선하는 것이다..
- 오픈소스 소프트웨어는 바이럴(viral)하며 확산속도가 빠르다.

이러한 정의가 친근한가? 불행하게도 우리가 무엇을 믿거나 바라든 이 개념은 정확하지 않다. 당연히 우리는 이 이슈에 대해 논의할 것이다. 잠시 동안 우리는 그들이 부정확하다고 가정할 것이다.<sup>895)</sup>

우리의 논증을 직접 검증하기를 원하는 사람들을 위해, 우리는 논증과 인용의 간략한 요약을 만들었다. 이 요약은 부록에서 제공한다. 다시한번, 우리는 오픈소스 소프트웨어 대한 다른 책이 필요한가? 적어도 독일 회사 도이치 텔레콤(Deutsche Telekom AG)에게는 필요하다. 대기업의 관점에서 주장하는 것은 일반적인 오해를 파악할 뿐 아니라 대기업의 고유한 요구 사항을 충족시켜야 한다. 실제로 회사의 규모가 크기 때문에 여러 문제가 발생한다.

대기업은 중소기업보다 다양한 환경에서 더 많은 오픈소스 소프트웨어를 사용한다. 중요한 질문은 모든 회사가 '우리가 존중해야 하는 오픈소스 소프트웨어의 모든 요구사항을 존중하고 있나?'라는 질문을 해야 한다는 것이다. 그러나 대기업은 이 질문에 대해 중소기업만큼 쉽게 대답할 수 없다. : 서로 다른 상황에서 여러 가지 다양한 오픈소스 배포가 이루어지기 때문에 작은 문제에서 발생할 수 있는 사례, 경우에 따른 관리가 필요에 부합하지 않는다. 이것은 시간과 돈을 낭비하게 만든다. 또한, 오픈소스 소프트웨어 라이선스 전문가로서 각 소프트웨어 팀에서 최소 한 명의 직원을 교육하는 것은 비용 효율성과 안정성 면에서 비현실적이다.

그럼에도 불구하고 대기업이라 할지라도 특히 도이치 텔레콤(Deutsche Telekom AG)은 오픈소스 소프트웨어의 규칙을 철저히 준수하길 원한다. 회사가 '우리는 우리가 존경해야 하는 문제가 될 수 있는 오픈소스 소프트웨어의 모든 규칙들을 정확히 준수한다고 확신하는가?'라는 질문에 대해 깨달았을 때 오픈소스 애호가로 알려진 일부 직원들에게 이 질문에 대한 답변을 얻기 위한 서비스와 프로세스를 수립하도록 직접 요청했다.

따라서 OSLiC(Open Source License Compendium)의 초기 저자인 우리가 도이치 텔레콤(Deutsche Telekom AG) 경영층으로부터 요청을 받은 것은 놀라운 일이 아니다. 공식적으로 오픈소스 관련 업무를 진행한 것에 대해 자랑스럽게 여긴다. 그러나 우리가 본 업무를 수행하면서 우리는 오픈소스에 대한 또 다른 책이 필요할지 스스로 자문해야 했다. 답은 '그렇다!'이다. 왜 그런지 이유는 다음과 같다.

첫째, 우리는 이미 지원하는 소프트웨어가 있다는 것을 알고 있다. 메타프로그램은 다른 애플리케이션의 코드를 수집하고 애플리케이션<sup>896)</sup>에 포함된 오픈소스 컴포넌트들의 리스트를 나열할 것이다. 그러나 이러한 지원 소프트웨어가 문제가 해결되어야 한다고 생각했던 방식과 항상 일치하지 않는다는 것을 이미 알고 있었다. 두 번째, 우리는 신뢰할 수 있는 가이드가 필요하다는 것을 상당히 빨리 인식했다. 우리는 개인적으로 우리는 개인적으로 우리 회사의 프로젝트에 대한 승인이 필요하다고 요청을 받았다. 그러나 심지어, Heise-Ticker가 모든 상황을 정확하고 완전히 설명했다 하더라도 '오, 몇일 전 Heise-Tiecker에서 이 내용을 읽었다.'라는 것을 가지고 그러한 요청에 답변을 줄 수 없었다. 우리 자신들은 우리가 이미 오픈소스 소프트웨어에 대해 많은 것을 알고 있는 것 보다 더욱 신뢰할 수 있어야 했다.<sup>897)</sup>

895) 우리의 논증을 직접 검증하기를 원하는 사람들을 위해 우리는 논증과 인용의 간략한 요약을 생성하였다. 이 요약은 부록에서 확인할 수 있다.

896) 일반적으로 블랙덕(<http://www.blackducksoftware.com/>)과 팔라미다(<http://www.palamida.com>)를 말할 수 있다.

897) 많은 다른 독일 오픈소스 애호가들처럼 Heise-Ticker를 주요 IT 정보 소스로 사용하는 것을 좋아한다는 점을 수 정하는 영광을 차지해야 했다. 불행히도 이러한 명성은 뉴스가 직접 인용될 수 있을정도로 높지 않았다.

그럼에도 불구하고 우리의 지식은 필요한 만큼 체계적이지 않았다. 오픈소스 라이선스의 기준을 충족시키기 위해 프로젝트 또는 제품개발 팀이 수행해야 했던 작업을 적절히 설명한 오픈소스 개요를 찾았다. 우리는 권고안을 기반으로 그 개요를 사용하기를 원했다.

우리는 매우 철저했지만 우리가 찾고자 하는 것을 찾지 못하였다. 우리의 '작은' 도서목록이 우리의 노력을 보여주고 있다. 우리가 발견한 것은 많은 원본 자료에 펼쳐있는 각각의 이슈들과 관련된 많은 정보들이었다. 우리는 특정 문헌에서도 질문에 대한 답을 찾지 못했다. 이슈들을 이해를 틀이기 위한 세가지 단계를 설명하면 다음과 같다.

오픈소스 라이선스 없이 오픈소스 운동은 없다. 그럼에도 불구하고 오픈소스 라이선스를 다룰 때 때로는 이를 무시한다. Apache Web Server를 예로 들어보자. 의심의 여지없이 특정 라이선스<sup>898)</sup>가 적용된 오픈소스 소프트웨어 중 가장 중요한 부분 중 하나이다.<sup>899)</sup> 또한, 상업 시장에서 오픈 소스 발전의 성공은 IBM이 *IBM WebSphere Application Server*의 자체 구성요소를 *Free Apache Web Server*로 전환하려는 IBM의 결정에 달려있다.<sup>900)</sup> 현재 많은 기업들이 *Apache http Server*를 사용하여 웹 서비스 프로바이더 역할을 하고 있다. 현재 *Apache http Server*는 다른 모든 *http Server* 비율보다 2배 더 많이 사용된다.<sup>901)</sup> 따라서 많은 비즈니스 모델이 Apache License에 의존한다. 또 다른 측면은 Apache의 설치, 설정, 유지보수에 대해 설명하는 유명한 *Apache Cookbook* 조차도 설치, 설정, 유지보수가 가능한 라이선스에 대해서는 언급하지 않았다<sup>902)</sup>. 색인에는 '라이선스'<sup>903)</sup>라는 단어가 나열되어 일지 않으며, '설치'<sup>904)</sup> 장이나 '기타'<sup>905)</sup>장에도 라이선스 질문이 진지하게 언급되어 있지 않다. 오픈소스 소프트웨어의 이점에 대해서는 단 한기지 힌트만 있다. 즉, 모든 사람들이 오픈소스 소프트웨어를 설치<sup>906)</sup>할 수 있다는 것이다. 이러한 문구를 기반으로 당신이 하고 있는 것을 하는 것이 허용 되었는지를 확신할 수 있는가?

당연히 *Apache Cookbook*은 변호사를 위한 책이 아니며 관리자 및 개발자를 위한 책이다. 그들은 합법

898) 예시. Apache Software Foundation : Apache License, 2.0, wp.

899) 아파치가 오픈소스 소프트웨어의 일부라는 것을 증명하기 위해 다음을 실행해야 한다. 첫째, 아파치는 ASF(Apache Software Foundation)에 의해 유지되는 메타 프로젝트와 같다. (<http://www.apache.org/>). 따라서 아파치 라이선스로 바로 뛰어 들 수 없다. 라이선스 링크가 일반적인 Apache License 하위 사이트로(예시. <http://www.apache.org/licenses/>) 인도하여도 반드시 프로젝트 사이트(<http://httpd.apache.org/>)에 접속해야 한다. "Apache Software Foundation 또는 그 프로젝트 또는 과제에서 제작 한 모든 소프트웨어는 아래 나열된 문서의 조항에 의거하여 라이선스가 부여된다." . 자신의 권리와 의무의 확인을 원한다면 Apache License Version 2.0으로 전환을 위해 제공된 링크를 사용할 수 있다. 그러나, 이는 어려운 작업이다. 라이선스를 충족시키기 위해 실행해야 할 간단한 목록은 존재하지 않는다. FAQ(<http://httpd.apache.org/docs/2.2/faq/>)에서도 확인할 수 없다. 현재는 FAQ는 wiki로 대체되고 있다. 오로지 서버는 '[...] 무제한의 라이선스와 함께"와 CD 코드에 삽입된다. (<http://wiki.apache.org/httpd/FAQ>). 따라서 ASF의 관점에서 라이선스는 스스로 모든 질문에 대해 답을 할 수 있다. [참조 다운로드: 2011-08-31]

900) 예시. Moody : Die Software-Rebellen, 2001, pp. ff.

901) 예시. Netcraft : 2011년 8월 웹 서버 설문조사: 2011

(URL: <http://news.netcraft.com/archives/2011/08/05/august-2011-web-server-survey-3.html>) - 참조 다운로드: 2011-08-31, wp.

902) 예시. Coar, Ken a. Rich Bowen: Apache Kochbuch: deutsche Übersetzung v. Jochen Wiedmann: Beijing [...]: O'Reilly, 2004, ISBN 3-89721-371-0, et passim.

903) 예시. cf. id., l.c., pp. 245ff, esp. p. 250

904) 예시. id., l.c., pp. 1ff.

905) 예시. id., l.c., pp. 219ff.

906) 예시. id., l.c., pp. 1: "... einer der Vorteile von open source software besteht darin, dass jedermann die Erlaubnis zur Erzeugung eines eigenen Installationskits hat".

성에 훙슬리기를 원치 않고, 가능한 한 빨리 Apache Web Server를 설치하고 일하기를 원한다. 실제로 이 Apache Cookbook은 훌륭한 지원을 제공한다. 하지만 당신이 하는 일을 실제로 하도록 허용되었는지 회사뿐만 아니라 스스로에게도 물어 보아야 한다. 당신은 Apache Cookbook에서 답을 찾을 수 있는가? 아니다. 당신은 라이선스 그 자체에서 찾을 수 있는가? 그렇다. 하지만 그것은 어렵다.<sup>907)</sup>

다시 말하자면: 당신은 Amazon이 현재 'apache server'라는 검색어에 대해 가장 많이 추천하는 다른 책에서 답을 찾을 수 있는가?<sup>908)</sup> 실제로 Sascha Kersken의 Apache 2.2 Handbook은 라이선스 장을 제공한다. 하지만 그 책은 겨우 두 페이지에 불과하다.<sup>909)</sup> 더군다나, 권리와 의무들은 단지 5개의 항목으로 요약되어 소프트웨어와 라이선스를 고객에게 양도해야 하는 시기와 개선 사항을 숨길 수 있는 시점을 설명해주지 않는다.<sup>910)</sup>

이것은 우리에게 필요한 모든 세부 사항을 설명하고 관련사항에 대한 빠른 접근을 제공하는 '일반 라이선스 요리책과 같은 것을 사용하지 못한다는 문제를 가져다준다.

물론 우리는 인터넷으로 탐색했다. 최소한 독일권에는 오픈소스 라이선스에 관한 훌륭한 사이트가 있다. 2000년에 (독일)변호사의 관점에서 'free software' 현상을 추적하기 위한 사립 연구소로 설립된 무료 및 오픈소스 소프트웨어의 Legal Aspects 연구소인<sup>911)</sup> ifross (*Institute for Legal Aspects of the Free and open source software* 정도로 번역 가능)<sup>912)</sup> 가 사이트를 제공하고 있다.

이 사이트는 다른 많은 측면 이외에도 매우 정교하게 작성된 FAQ<sup>913)</sup> 및 오픈소스 라이선스 및 기타 관련 라이선스의 수많은 목록들도 제공한다: 또한 분명히 copyleft-effect가 없거나(BSD), copyleft-effect가 엄격하거나(GPL) copyleft-effect가 제한적인(LGPL) 것들에 따라 오픈소스 라이선스를 분류하고 있다.<sup>914)</sup>

그러나 이 훌륭한 사이트 조차도 우리의 요구를 충족시키지는 못한다. 회사, 개발자 또는 프로젝트 관리자가 오픈소스 소프트웨어를 정기적으로 사용하도록 보장할 수 있는 맥락 중심의 과업목록을 제공하지 않기 때문이다.

따라서 우리는 가장 합법적인 참고문헌에 있는 표준 도서를 평가했다.<sup>915)</sup> Jaeger와 Metzger가

907) 또한 개발자와 유지관리자가 원래 라이선스를 읽으려고 원하는가? 우리는 정말로 그들이 사용된 모듈의 라이선스를 확인해야한다는 사실을 알길 원하는가?

908) Tested on <http://www.amazon.de/> at 2011-08-31.

909) Kersken, Sasche: Apache 2.2. Das umfassende Handbuch: 3rd, refreshed a. expanded edition: Bonn: Galileo Press, 2009, ISBN 978-8362-1325-7, pp. 111f 참조

910) 위와 동일, p. 112 참조

911) ifross : Ziele, Aufgaben, Geschichte: 2011 (URL: <http://www.ifross.org/node/16>) - reference download: 2011-09-05, wp. 참조

912) 원래는: "Institut für Rechtsfragen der Freien und open source software". Main entry point for its site is the URL <http://www.ifross.org/>.

913) ifross : FAQ: 2011 (URL: <http://www.ifross.org/faq-haeufig-gestellte-fragen>) - reference download: 2011-09-05, wp. 참조

914) 최소한 오픈소스를 다루는 독일 사법관련 문헌

915) Jaeger, Till a. Axel Metzger : Open Source Software. Rechtliche Rahmenbedingungen der Freien Software: 1st edition. Muñchen: Verlag C.H. Beck, 2002, ISBN 3406484026, pp. V - It can not be any surprise that both authors, Mr. Jaeger and Mr. Metzger are members of ifross (cf.

우려하는 책 – 대충 번역된 – 오픈소스 소프트웨어에 대한 사법 구조 요구사항<sup>916)</sup> 이 책의 초판조차도 이미 ‘저작권’장에서 명확한 구조를 가지고 있다.: 언급된 각 라이선스(또는 적어도 라이선스 클러스터마다)는 소프트웨어 사용자<sup>917)</sup>의 권한과 의무<sup>918)</sup>에 대한 하위목록을 제공했다. 화제가 된 오픈소스의 다른 많은 중요한 측면들도 논의되었다.<sup>919)</sup>

하지만 우리는 그 이상을 필요로 했다. 이 책의 우수함에도 불구하고 우리는 프로그래머에게 그 책을 당신이 접촉한 라이선스를 확인하고 관련 하위목록의 설명을 따르라고 추천하며 넘겨줄 수 없다는 확신이 있었다. 이 책에는 초판<sup>920)</sup>, 두 번째 판<sup>921)</sup> 뿐만 아니라 최근에 발행된 세 번째 판<sup>922)</sup>에도 단순히 확인할 수 있는 과업목록이 포함되어 있지 않았다. 그래서 만약에 누군가가 누군가에게 ‘이 것을 해라, 이 경우에는 그것을 해라, 결국은 이것을 또 하라’라고 말한 검증된 목록을 가지고 있지 않다면 회사, 개발자나 프로젝트 관리자가 어떻게 오픈소스 라이선스 요구사항을 충분히 충족할 수 있겠는가? 문현에서 필요한 단계를 밟춰해야 하는 오픈소스 라이선스 전문가가 된 이유는 무엇인가?

우리가 기존의 오픈소스 개요서를 검색하는 동안, 우리는 ‘오픈소스 소프트웨어 수립을 위한 개요서’<sup>923)</sup>라는 제목의 기사를 발견했다. 그것은 오픈소스 라이선스 조건하에 ‘소프트웨어 게재’를 위한 ‘실용적인 가이드북’과 ‘지원’을 목표로 한다.<sup>924)</sup> 더군다나, 이 기사의 끝 부분에서 저자들은 오픈소스 소프트웨어를 공개하기 위해, 오픈소스 소프트웨어를 게재하는 합법적이고 치밀한 과정을 얻기 위해 ‘가이드’를 섹션별로 수행해야 한다는 야심찬 계획을 세웠다.<sup>925)</sup>

이 기사의 저자들은 우리가 찾고 있는 것에 가까운 것을 설명한다. 실제로 이 기사에는 오픈소스 소프트웨어를 올바르게 처리하려는 경우 고려해야 할 중요한 측면이 나열되어 있다. : 독점 코드에 GPL코드를 포함시키거나 GPL 코드를 수정하는 경우 코드를 공개할 의무가 없음을 알린다. 이는 코드를 배포해야하는 다른 사람에게 일반인이 아닌 오직 그들에게만 바이너리를 넘겨줄 때에만 적용된다.<sup>926)</sup> 또한 기사는 소프트웨어를 정확히 설명 한다- 적어도 독일에서는-‘Nutzungsrechte’를 사용하는 권리를 다

---

http://www.ifross.org/personen/, wp). 참조

916) 위와 동일., pp. 30ff 참조.

917) pars pro toto: have a look at the chapter concerning the liability: cf. id., l.c., pp. 137ff.

918) For getting a good survey of the structure and the line of thought see the contents cf. id., l.c., pp. VIIIf.

919) pars pro toto: have a look at the chapter concerning the liability: cf. id., l.c., pp. 137ff.

920) cf. id., l.c., pp. VIIff.

921) cf. Jaeger, Till a. Axel Metzger : Open Source Software. Rechtliche Rahmenbedingungen der Freien Software; 2nd edition. Muñchen: Verlag C.H. Beck, 2006, ISBN 3406538037, pp. VIIff.

922) cf. Jaeger a. Metzger : Open Source Software. Rechtliche Rahmenbedingungen der Freien Software, 2011, pp. VIIff. Naturally we use this latest edition for adopting or discussing systematical aspects.

923) 대략 번역됨.

924) 참조. Bretschneider, Ulrich, Rainer Glaschick , a. Gernot Gräfe : Ratgeber für die Veröffentlichung von Open-Source-Software durch eine Hochschule: In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 166f (originally: ein “pragmatischer Ratgeber” zur “Veröffentlichung einer Software unter den Rahmenbedingungen einer Open-Source-Lizenz”).

925) 참조. id., l.c., pp. 186 (originally: ein “Ratgeber”, der es erlaubt “( . . . ) die zu berücksichtigende Aspekte (strukturiert abzuarbeiten) ( . . . )” und einen “rechtlich nicht angreifbaren Veröffentlichungsprozess” zu ermöglichen).

926) 참조. id., l.c., pp. 170 and 181.

른 사람에게 전가함으로써 오픈소스 소프트웨어로 인정될 수 있지만 저작권 자체 –the ‘Urheberpersönlichkeitrecht’–는 양도할 수 없으며 저자에게 속한다.

게다가, 다른 측면 이외에 기사는 독일에서 유효하지 않은 무보증 조항의 문제를 간략하고 깊이 있게 논의하므로 기부에<sup>927)</sup> 대한 책임규정이 자동으로 설명될 것이다. 이 기사는 마지막으로 실제로 Copyleft의 아이디어와 LGPL과 GPL의 차이점을 요약한다. 이 기사는 마지막으로 실제로 Copyleft의 아이디어와 LGPL과 GPL의 차이점을 요약한다.<sup>928)</sup>

하지만 약간의 격차가 남아 있다. 이 기사는 대학이나 회사가 오픈소스 소프트웨어를 기반으로 한 개발 산출물을 공개해야 하는 경우를 분석하지 않는다. 서로 다른 라이선스와 조건들을 구분하지 않고 있으며, 또한 (내부적으로)오픈소스 소프트웨어를 (재)사용하거나 배포하는 대학교나 회사가 접촉한 오픈소스 라이선스를 이행하기 위해 해야 하는 것에 대해서는 논의하지 않는다. 마지막으로, 이 기사에서는 약속대로 단계별 목록을 제공하지 않는다.

그러나 우리는 이 기사로 인해 두 가지 면에서 지지를 얻는다. 첫 번째, 그것은 몇 가지 주요 문제에 대한 잘 쓰여 진 요약이었다. 두 번째, 그것은 오픈소스 소프트웨어를 공개하는 법적으로 ‘철저한’ 프로세스를 수립할 수 있는 개요를 가질 필요성을 언급했다.<sup>930)</sup>

우리는 우리가 세운 가정 하에 정당화 된 것처럼 보였다. 그러나 우리가 찾고 있던 오픈소스 개요는 보다 실용적이고, 처리가 쉽고, 구별되며 정교해야 했다.

다시 말하자면: 우리는 오픈소스 소프트웨어에 관한 새로운 책이 필요했는가? 우리는 신뢰할 수 있는 통합된 오픈소스 개요를 찾아왔다. 그러나 우리는 별도의 조각적인 정보와 오늘날 알 수 있는 일부 풍문을 발견했다. 우리의 대답은 분명했다: 우리는 당연히 오픈소스에 관한 새로운 일반 서적이 필요하지 않았다. 그러나 책임이 있는 개발자, 프로젝트 관리자 또는 제품 개발자가 오픈소스 라이선스를 충족시키는 데에 필요한 설명이 부족했다. 우리는 오픈소스 라이선스 개요가 필요했다.

최고 수준의 오픈소스 라이선스 개요에는 ‘이행해야 할 라이선스-과업목록’을 처리하는 일련의 기능이 포함되어있다. 또한 이러한 목록들에 대해 이해하기 쉽고 사용자에게 친숙한 검색 옵션을 제공해야 한다. 어쨌든, 개발자와 프로젝트 관리자는 오픈소스 라이선스 전문가가 되기 위한 노력을 함께 해야 한다. 다른 사용자의 경우에도 왜 이렇게 해야만 하는지 이유를 분명하게 설명해야 한다. 이런 이유로 신뢰할 수 있는 오픈소스 라이선스 개요는 무엇을 해야 하는지 열거해야 할 뿐만 아니라 철저히 검증된 세부 정보와 명확하게 요약된 지침을 제공해야 한다.

비록 우리가 오픈소스 개요를 찾지는 못했지만 오픈소스 집단의 정신에 익숙해졌다. 그러므로 우리는 가장 간단한 규칙을 따랐다. : ‘당신이 놓친 것은 당신 스스로 개발해야 한다.’ 몇 가지 원칙은 우리가 목표를 달성하는 데 도움이 된다.

---

927) 참조. id., l.c., p. 173.

928) 참조. id., l.c., p. 177.

930) 참조. Bretschneider, Glaschick, a. Grafe: Ratgeber für die Veroentlichung von Open-Source-Software durch eine Hochschule, 2008a, p. 186.

**과업목록을 핵심으로 하여, 그것에 관한 토론:** 우리의 작업은 두 부분으로 분리되어야 한다. 그 핵심으로 우리는 과업목록 세트를 제공하기를 원했다. 각각의 목록들은 특정 오픈소스 라이선스와 관련이 있어야 하고 오픈소스 특정용도 사례별로 그룹핑 되어야 한다. 오픈소스 소프트웨어의 라이선스 및 규칙 핵심 해석에 영향을 미치는 모든 측면을 정밀하게 특성화 해야 한다. 그럼에도 불구하고 사용자는 세부 정보를 건너뛰고 필요한 부분으로 직접 이동할 수 있어야 한다.

**철저하게 특정 출처가 있는 인용문:** 사용자가 이 오픈소스 개요서를 다 읽을 필요는 없지만, 우리를 맹신해서도 안된다. 우리는 계속 개정되기를 원한다. 우리의 출처와 결론을 쉽게 검증할 수 있어야 하기 때문에, 우리는 새로운 '사실'을 발명하는 것이 아니라 정보를 수집하는 것이 우리의 과제라는 점에서 학술 인용을 사용하고 도서목록 데이터를 광범위하게 나열하기로 결정했다.

**인터넷 뿐만 아니라 책과 기사들:** 인터넷이 비록 다소 수정된 사본으로 가득 차 있다 할지라도 원본을 찾고 싶었다. 우리는 신뢰할 수 있는 사실과 설명을 얻길 원했다. 그래서 우리는 인터넷뿐만 아니라 기술이 뛰어난 출처-예를 들면 대학 도서관에서 제공하는 것들-도 평가하기로 결정했다.

**산림지를 청소하지도 말고 파괴하지도 말라:** 우리는 라이선스 및 법적인 측면을 다루어야 함에도 불구하고 자세한 논의에서 길을 잃고 싶지 않았다. 특정한 종류의 처리가 여전히 합법적이거나 이미 금지되어 있는지 여부를 알아내는 것이 우리의 임무가 되어서는 안 된다. 우리는 라이선스와 싸우고 싶지 않았다. 그것들의 범위를 늘리거나 경계를 시험하기를 원하지 않았다. 우리는 오픈소스 라이선스를 있는 그대로 받아들였다: 개발자로부터 개발자를 위해 작성된 규칙들. 그리고 이러한 라이선스 중 일부가 법률시스템과 관련하여 유효하지 않을지라도<sup>931)</sup>, 우리는 -최소한 일반 법률을 위반하지 않는 한 그것들을 지침으로 삼고 싶었다.<sup>932)</sup> 우리는 정말로 오픈소스 라이선스의 조금 확실하지 않은 숲을 가로지르는 하나의 입증된 방법을 찾고 싶었다. 라이선스의 일부 조항이 끝내 우리에게 강요할 수 없다 하더라도 우리는 그것들을 '자발적으로' 존중하기를 원했다. 오픈소스 개발자나 자유 소프트웨어 재단과의 라이선스 분쟁이 일어날 가능성을 배제하고 사용자를 지원하는 일련의 규칙을 제공하고자 했다.

**본문을 진지하게 받아들여라:** 다른 면에서 우리는 라이선스 본문을 있는 그대로 해석하고 싶었다. 만약 이러한 것들이 부족하다면,<sup>933)</sup> 우리는 오픈소스 생각의 정신으로 진행 사안을 해석할 것이다.

931) 실제로 GPL의 경우를 예를 들어 이런 식으로 논쟁할 수 있다: 비록 당신이 GPL을 각각 "기부" 형태의 계약인 "Schenkung"으로 가져가더라도, 그것은 각각 AGBs "Allgemeine Geschaftsbedingungen"의 형태로 제시되며 따라서 일반적인 AGB 규칙을 따라야 한다. 독일에서 '유감스럽게도' 이러한 일반적인 AGB규정은 각 유형의 보증을 제외할 수 없다. 만약 우리가 Oberhem을 따를 경우 GPL의 §11 및 §12는 이러한 일반적인 AGB규칙 때문에 독일에서 무효되어야만 한다. 더군다나 Oberhem은 §5조차도 -GPL의 규칙을 존중한다면 GPL소프트웨어를 사용하고 배포할 수 있는 권리를 얻을 수 있는 GPL의 중요한 조항- 또한 "unwirksam" 각각 무효로 보인다. 그러나 좋은 소식은 GPL이 무효조항을 포함하더라도 무효가 아니라는 것이다. Oberhem, Carolina: Vertrags- und Haftungsfragen beim Vertrieb von Open Source Software; Dissertation: Hamburg: Verlag Dr. Kovač, 2008 (= Recht der Neuen Medien, [Vol./No.] 50), ISBN 978-3-8300-4075-0, pp. 128, 133ff, 150ff, esp. 146, 159.

932) 그들이 분명히 하지 않는 것!

933) 라이선스의 계획적인 미결정은 오픈소스에서 각각 자유 소프트웨어 운동에 알려진 문제이다. RMS의 전기에 따르면 그의 주요 법률 고문인 Moglen은 "모든 법적 절차(...)에는 불확실성이 있다"고 언급해왔고 "모든 버그(...)를 제거하기 위해" 시도하는 것은 어리석은 것(...)처럼 보였다. 그렇지만 - 또한 Moglen resp. Williams -Richard Stallman의 목표는 "완전한 반대"였다: 그는 "본질적으로 불가능한 불확실성을 제거를 위해" 시도했다. 하지만-그리고 그것이 분석의 핵심이다-Moglen은 RMS챕터 때문에 Stallman을 따라야만 했다. 그리고 그는 그의 작업을 요

그러나 본문이 명확하고 분명한 경우, 우리는 그 의미가 잘 알려진 오픈소스 '사실'에 반대하는 경우일지라도 그 제안을 확실한 결정으로 받아들이고 싶었다.

**군중을 믿어라:** 우리는 우리 자신의 연구만을 기반으로 사용하고 싶지 않았다. 우리는 군중이 무작위로 선택된 일부 전문가보다 더 강하다는 것을 알고 있었다. 그래서 우리는 아직 끝나지 않은 조기 공개 0.2로 시작하는 저작물로 본문을 게재하기로 결정했다. 그리고 우리는 사회가 우리와 함께 그 개요를 완성하도록 초청하길 원한다. 우리는 오픈소스 개요를 GIT 또는 다른 버전의 제어 시스템에서 개발되고 관리될 수 있는 LaTeX 및 BibTeX 파일 세트로 정교하게 다룰 것이다. 마지막으로 우리는 다른 사람들이 우리의 것을 바로 잡아주고 도울 수 있는, 심지어 그들의 목적을 위한 결과를 얻을 수 있는 Creative Commons Attribution-Share Alike German 3.0 라이선스로 텍스트를 게시할 것이다.

그래서 우리는 실행했고, 결과는 다음과 같다. 우리의 라이선스 정책에 따라서 자유롭게 사용하길 바란다.

### 8.2.2 무엇(What)

다음은 어떻게 이 개론을 사용할지에 대한 간략한 설명이다 :

**동일한 아이디어, 다른 라이선스.** : Free Software 운동<sup>934)</sup>, 오픈소스 운동<sup>935)</sup>, GNU -Project<sup>936)</sup>에서 오픈소스 라이선스를 해석하기 위한 배경 지식을 찾을 수 있을 것이다. 우리는 오픈소스 라이선스를 결합하기 위한 각각의 방법을 논의하였다. 마침내 우리는 '개발자를 보호하는' 레이블을 바탕으로 자체 분류를 제시하였다. '라이선스 된 코드의 보호'와 상위 개발 보호'. 다른 라이선스들과 특별한 라이선스들을 분류하는 방법에 대해 친숙하거나, 당신이 라이선스 그룹에 대한 설명을 바탕으로 자신의 행동을 승인할 수 없다는 것을 아는 경우라면, 우리 생각의 이해와 우리의 분류법과 단어를 이해하기에 충분할 것이다.

**파생 저작물의 문제** : 이 챕터는 중요하다. 소프트웨어 개발자 정신으로 어떤 종류의 프로그래밍이 파생 작업을 불러일으키는지, 또는 그렇지 않은지 설명하였다. 우리가 이행해야 할 목록 대해 이 분석 내용을 참조할 것이다.

---

약해야만 했다. “(...)정밀한 결과(GPL,KR의), 간단한 결과(GPL의:KR)는 달성해야만 하는 것을 거의 달성하였다” 따라서 우리는 라이선스의 본문을 진지하게 받아들였다. 참조. Williams : Free as in Freedom. Richard Stallman's Crusade for Free Software, 2002, pp. 177

934) 적어도 이 장소에서 당신은 아마도 우리가 FLOSS, F/OSS, F/LOSS 와 같은 어표를 사용 할 것이라고 예상했을 것이다. Later on 이라는 개념은 해석하기에 따라 다르기 때문에 Free Software 사용에 어려움을 겪게 될 것이다. 나중에 '대체'로 디자인되거나 '분리자'로 실행하는 오픈소스 컨셉의 발명에 대해 확인할 예정이다. 언급된 어표들은 재수립을 위해 소개되거나, 적어도 일반적인 히스토리와 '양쪽' 운동의 공통 중심을 강조하기 위해 Libre는 자유라는 단어의 모호성을 해결할 것이다. 첫 번째 조사 참고. Free and open source software; n.l., 2011 (URL: [http://en.wikipedia.org/wiki/Free\\_and\\_open\\_source\\_software](http://en.wikipedia.org/wiki/Free_and_open_source_software))

참고 다운로드: 2011-09-08, wp.

935) 간단하고 유익한 소개를 위해 참고. Fogel : Producing Open Source Software, 2006, pp. 231ff esp. p. 232f.

936) OSD는 우리 분석의 범위를 지정하기 때문에 우리는 오픈소스 컨셉에 머무를 것이다. RMS의 공격으로부터 우리를 지키지 못할지라도 우리는 Stallmann와 FSF에게 깊은 경의를 표한다.

**다른 라이선스와 결합의 문제** : 이 챕터를 간과해서는 안 된다. 우리는 왜, 어떻게 다른 라이선스의 소프트웨어와 결합하는 것이 우리가 일반적으로 말하는 것보다 위험하지 않은지 설명하였다. 이 챕터의 결과는 우리가 이행해야 할 과업 목록에 영향에 주었다.

**오픈소스와 수익** : 우리는 수익이 문제가 되지 않는 방법에 대해 짧게 논의하였다. 제 2, 3자에게 오픈소스 소프트웨어 일부를 라이선싱하기 위한 비용을 요구하는 것이 금지되어 있다는 것을 알고 있거나, 몇몇 라이선스에 의해 금지되어 있다는 것을 알고 있다면 이 챕터를 읽는 것을 미룰 수 있다.

**암묵적인 특허 해제 문제** :- 여기서 우리는 소프트웨어 특허의 일부 견해와 일부 오픈소스 라이선스에 의해 어떻게 처리되는지 설명하였다. 어떠한 라이선스가 당신의 특허와 암묵적으로 문제가 발생되는지 알아야 한다. 그러나 소프트웨어 특허 개론을 작성하는 것이 우리의 목적은 아니다.

**분류의 원리로 오픈소스 사용 사례** :- 이 챕터는 매우 중요한 챕터이다. '그대로 사용(Use as it is)', '코드 수정(Modify the Code)', '재배포(With Redistribution)', '재배포하지 않음(Without Redistribution)', '독립된 초기 개발(Isolated Initial Development)', '온 탑 개발(On-Top-Development)')과 같은 카테고리를 설명하였다.: 우리는 '다른 라이선스와의 결합'과 '파생 저작물 생성'의 부작용과 관련하여 우리의 분류체계를 개발하고 논의하였다. 이 분류체계는 다음 챕터를 정의하였다.

**오픈소스 라이선스: 특별한 과업 목록을 찾아라** : '과업 목록을 위한 파인더'의 작성과 어떠한 관련된 양상들의 요약을 정의하였다. 우리의 설명을 읽는 것을 원치 않더라도 이 챕터는 자주 사용하게 될 것이다.

**오픈소스 라이선스 이행 : 분류된 과업 목록** : 이 챕터는 모두 분류된 과업 목록을 제공한다. 하위 챕터의 구조는 우리의 파인더의 구조와 우리의 분류 구조와 동일할 것이다.

**오픈소스 라이선스와 법적 환경** : 일반적인 방법으로 오픈소스 소프트웨어를 왜 사용해야 하는지에 대해 논의하였고, 이는 라이선스 질문뿐만 아니라 주변 법률 시스템과 관련된 내용도 논의하였다.

**첨부 : 몇몇 널리 알려진 오픈소스의 근거 없는 믿음** : 이 챕터의 시작 부분에 언급된 모든 문제가 왜 잘못되었는지 설명하기 위한 약속을 잘 지켰다. 원활 때마다 본 챕터를 특별한 소개 또는 상기시키기 위한 조언의 끝말로 활용할 수 있다.

## 간행물, 쇼트컷, 약어

- AGPL ..... GNU Affero General Public License  
ApL ..... Apache License  
BISE ..... Business & Information Systems Engineering [ISSN: 1867-0202]  
BSD ..... Berkeley Software Distribution (License)  
[n.abbr.] ..... Berkeley Technology Law Journal  
BWV ..... Berliner Wissenschafts-Verlag GmbH  
[n.abbr.] ..... Cultural Anthropology [ISSN: 1548-1360]  
CiHB ..... Computers in Human Behavior [ISSN: 0747-5632]  
CotACM ..... Communications of the ACM [ISSN: 0001-0782]  
CR ..... Computer und Recht. Zeitschrift f r die Praxis des Rechts der Informationstechnologien  
CRI ..... Computer Law Review international [ISSN: 1610-7608]  
[n.abbr.] ..... Computers & Education [ISSN: 0360-1315]  
[n.abbr.] ..... Cutter IT Journal [ISSN: 1048-5600]  
DDT ..... Drug Discovery Today [ISSN: 1359-6446]  
DSS ..... Decision Support Systems [ISSN: 0167-9236]  
[n.abbr.] ..... Ethics and Information Technology [ISSN: 1388-1957]  
E.C.L.R. ..... European Competition Law Review  
EER ..... European Economic Review [ISSN: 0014-2921]  
EPL ..... Eclipse Public License  
et seqq. ..... and the following ones  
EUPL ..... European Union Public License  
GPL ..... GNU General Public License  
[n.abbr.] ..... Information & Management [ISSN: 0378-7206]  
ibid ..... ibidem = latin for 'at the same place'  
ICC ..... Industrial and Corporate Change [ISSN: 0960-6491]  
id ..... idem = latin for 'the same', be it a man, woman or a group...  
IEaP ..... Information Economics and Policy [ISSN: 0167-6245]  
[n.abbr.] ..... IEEE Software [ISSN: 0740-7459]  
ifross ..... Institut f r Rechtsfragen der Freien und Open Source Software  
[n.abbr.] ..... International Information and Library Review [ISSN: 1057-2317]  
[n.abbr.] ..... International Journal of Medical Informatics [ISSN: 1386-5056]  
[n.abbr.] ..... interactions [ISSN: 1072-5520]  
ISJ ..... Information Systems Journal [ISSN: 1365-2575]  
ITRB ..... Der IT-Rechtsberater [ISSN: 1617-1527]  
JAIS ..... Journal of the Association for Information Systems [ISSN: 1536-9323]  
JCSC ..... Journal of Computing Sciences in [Small] Colleges [ISSN: 1937-4771]

JISE	Journal of Information Science and Engineering [ISSN: 1016-2364]
JLEO	Journal of Law, Economics, & Organization [ISSN: 1465-7341]
JMIR	Journal of Medical Information Research [ISSN: 1438-8871]
[n.abbr.]	Journal of Academic Librarianship [ISSN: 0099-1333]
[n.abbr.]	Journal of Comparative Economics [ISSN: 0147-5967]
[n.abbr.]	Journal of Systems and Software [ISSN: 0164-1212]
JSIS	Journal of Strategic Information Systems [ISSN: 0963-8687]
I.c.	loco citato = latin for 'in the place cited'
LGPL	GNU Lesser General Public License
LJ	Linux Journal [ISSN: 1075-3583]
MIT	Massachusetts Institute of Technology (License)
MPL	Mozilla Public License
Ms-PL	Microsoft Public License
n.abbr	no abbreviation (known)
[n.abbr.]	etWorker [ISSN: 1091-3556]
n.y	year not stated / no year
n.l.	location not stated / no location
np	no page numbering
n.st	not stated
[n.abbr.]	Organization Science [ISSN: 1047-7039]
PgL	Postgres License
PHP	PHP (License)
[n.abbr.]	Queue [ISSN: 1542-7730]
[n.abbr.]	R&D Management [ISSN: 1467-9310]
RP	Research Policy [ISSN: 0048-7333]
SIGCSE Bulletin	SIGCSE Bulletin [ISSN: 0097-8418]
SIGCAS	ACM SIGCAS Computers and Society [ISSN: 0095-2737]
SIGMIS Database	ACM SIGMIS - The Data Base for Advances in Information Systems [ISSN: 0095-0033]
SIGSOFT SEN	SIGSOFT Software Engineering Notes [ISSN: 0163-5948]
[n.abbr.]	Stanford Law Review [ISSN: 00389765]
[n.abbr.]	Software Quality Journal [ISSN: 0963-9314]
STHV	Science, Technology & Human Values [ISSN: 0162-2439]
ToIT	Transaction on Internet Technology [ISSN: 1533-5399]
ToSEM	Transactions on Software Engineering Methodology [ISSN: 1049-331X]
Ubiquity	Ubiquity - The ACM IT Magazine and Forum [ISSN: 1530-2180]
UB	'Universitätsbibliothek' = library of university X
ULB	'Universitäts- & Landesbibliothek' = library of university and state X

[n.abbr.] ..... University of Chicago Law Review  
[n.abbr.] ..... University of Illinois Law Review  
[n.abbr.] ..... University of Pittsburgh Law Review  
wp ..... webpage / webdocument without any internal (page)numbering  
ZGE / IPJ ..... Zeitschrift für geistiges Eigentum [ISSN: 1867-237x]

## 참고문헌

- Perspectives on free and open source software; o.O.??: ???, 2005
- Proceedings of the Linux Symposium; Ottawa, 2006
- Ågerfalk, Pär et al., editors: Open Source Software: New Horizons; 6th International IFIP WG 2.13 Conference on Open Source Systems, OSS 2010; (= IFIP Advances in Information and Communication Technology, [Vol./No.] 319) Berlin, Heidelberg u. New York: Springer, 2010, BibWeb/PDF, ISBN 978-3-642-13243-8
- Ahtainen, Aleksi , Sami Surakka, a. Mikko Rahikainen: Plaggie: GNU-licensed Source Code Plagiarism Detection Engine for Java Exercises; in: Proceedings of the 6th Baltic Sea Conference on Computing Education Research; New York, NY, USA: ACM, 2006 (= Baltic Sea '06) hURL: <http://doi.acm.org/10.1145/1315803.1315831> - reference download: 2011-12-29, BibWeb/PDF, pp. 141-142
- Airlie, D. M.: Open Source Graphic Drivers. They Don't Kill the Kittens; In Proceedings of the Linux Symposium, 2006, p. 19.26
- Alexi, O. a. J. Henkel: Promoting the Penguin: Who Is Advocating Open Source Software in Commercial Settings? München, 2007 hURL: <http://ssrn.com/abstract/>=988363
- Alexy, Oliver : Free Revealing. How Firms Can Profit From Being Open; Wiesbaden: Gabler, 2009 (= Gabler Edition Wissenschaft), Print and BibWeb/PDF, ISBN 978-3-8349-1475-0
- Allman, Eric: A Conversation with Chris DiBona; in: Queue, 1 July (2003), pp. 10-19 hURL: <http://doi.acm.org/10.1145/945074.945130> - reference download: 2011-12-29, BibWeb/PDF
- Allman, Eric a. Marshall Kirk McKusick: From the Editors: Open Source Revisited; in: Queue, 2 May (2004), pp. 8-9 hURL: <http://doi.acm.org/10.1145/1005062.1005072>
- Alspaugh, Thomas A., Hazeline U. Asuncion, a. Walt Scacchi : Analyzing software licenses in open architecture software systems; In Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2009, pp. 54-57 hURL: <http://dx.doi.org/10.1109/FLOSS.2009.5071361> - reference download: 2011-12-28, BibWeb/PDF

- Alspaugh, Thomas A., Walt Scacchi, a. Hazeline U. Asuncion: Software Licenses in Context: The Challenge of Heterogeneously-Licensed Systems; in: JAIS, 11 (2010), No. 11/12, pp. 730-755, BibWeb/PDF
- Amega-Selorm, Charles a. Johanna Awotwi: Free and Open Source Software (FOSS): It's Significance or Otherwise to the E-Governance Process in Ghana; in: Proceedings of the 4th International Conference on Theory and Practice of Electronic Governance; New York, NY, USA: ACM, 2010 (= ICEGOV '10) hURL: <http://doi.acm.org/10.1145/1930321.1930342> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0058-2, pp. 91-95
- anonymous: Mircosoft und Open Source: nichts als Ärger; 2000 hURL: <http://www.heise.de/newsticker/meldungen/8314>
- anonymous: Microsoft und die GPL: Freiheit, die ich meine ... 2002 hURL: <http://www.heise.de/newsticker/meldung/26355>
- anonymous: Mircosoft Shared Source Initiative Overview; 2004 hURL: <http://www.microsoft.com/resources/sharedsource/Intiative/Intiative.mspx>
- Anonymous: The Triumph of the Commons; in: The Economist (2005)
- Anvaari, Mohsen a. Slinger Jansen: Evaluating Architectural Openness in Mobile Software Platforms; in: Proceedings of the Fourth European Conference on Software Architecture: Companion Volume; New York, NY, USA: ACM, 2010 (= ECSA '10) hURL: <http://doi.acm.org/10.1145/1842752.1842775> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0179-4, pp. 85-92
- Apache Software Foundation: Apache License, Version 2.0; 2004, FreeWeb/Html hURL: <http://www.apache.org/licenses/LICENSE-2.0> - reference download: 2011-08-31
- Apache Software Foundation: Licenses; 2013 [n.y.], FreeWeb/Html hURL: <http://www.apache.org/licenses/i> - reference download: 2013-02-25
- Arlt , Brinkel, a. Volkmann; Spindler, Gerald, editor : 'BSD' - und 'Mozilla'-artige Lizenzen; In Spindler : Rechtsfragen bei Open Source Software, 2004, pp. 317-372, Print
- Arnö, Kaj : Dual Licensing - A Business Model from the Second Generation of Open-Source Companies; In Wynants a. Cornelius: How Open is the Future?, 2005, pp. 479-486

- Asay, Matt: A Funny Thing Happened on the Way to the Market: Linux, the General Public License, and a New Model for Software Innovation; Stanford CA, 2002, Web/Pdf hURL: <http://www.linuxdevices.com/files/misc/asay-paper.pdf>
- Asche, Michael et al., editors: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen; (= POWeR / Patent Oensive Westfalen Ruhr, [Vol./No.] 3) Münster, New York, München [... etc.]: Waxmann, 2008, Print, ISBN 978-3-8309-1845-5
- Ascher, David: Is Open Source Right for You? in: Queue, 2 May (2004), pp. 32-38 hURL: <http://doi.acm.org/10.1145/1005062.1005065> - reference download: 2011-12-28, Bib-Web/PDF
- Asiri, Sami: Open Source Software; in: SIGCAS, 33 March (2003), p. 2 hURL: <http://doi.acm.org/10.1145/966498.966501> - reference download: 2011-12-28, BibWeb/HTML
- Asundi, Jai: The Need for Eort Estimation Models for Open Source Software Projects; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 6:1-6:3 hURL: <http://doi.acm.org/10.1145/1082983.1083260> - reference download: 2011-12-29, BibWeb/PDF
- Axel Metzger, Till Jaeger und: Die neue Version 3 der GNU Genereal Public License; in: GRUR, o.A. (2008), No. 2, pp. 130-137, Copy
- Ayala, Claudia et al.: Challenges of the Open Source Component Marketplace in the Industry; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 213-224, BibWeb/PDF
- Azzi, R. Michael: CPR: How Jacobsen V. Katzer Resuscitated the Open Source Movement; in: University of Illinois Law Review, (2010), No. 4, pp. 1271-1302, BibWeb/PDF
- Babcock, Charles: Big Test For Open Source GPL; in: Informationweek, 17 December (2006), p. np., Copy
- Bach, Paula M. a. John M. Carroll : Characterizing the Dynamics of Open User Experience Design: The Cases of Firefox and OpenOce.org; in: JAIS, 11 (2010), No. 12, pp. 902-925, BibWeb/PDF
- Backu, Frieder: Open Source Software und Interoperabilität; in: ITRB (IT-Rechtsberater),

(2003), p. 180

- Baerwol, Matthias, Robert A. Gehring, a. Bernd Lutterbeck, editors: Open Source Jahrbuch 2005. Zwischen Softwareentwicklung und Gesellschaftsmodell; Berlin: Lehmanns Media, 2005 hURL: [http://www.opensourcejahrbuch.de/download/jb2005/OpenSourceJahrbuch2005\\_online.pdf](http://www.opensourcejahrbuch.de/download/jb2005/OpenSourceJahrbuch2005_online.pdf) - reference download: 2011-10-17, Print & Free-Web/PDF, ISBN 3-86541-059-6
- Bain, Malcolm et al.: Legal Aspects of the Information Society; Oberta de Catalunya: Free Technology Academy, 2010 hURL: <http://www.ftacademy.org/materials/fsm/6#1i> - reference download: 2012-101-20, FreeWeb/PDF
- Baird, Stacy Avery: The Heterogeneous World of Proprietary and Open-Source Software; in: Proceedings of the 2nd international conference on Theory and practice of electronic governance; New York, NY, USA: ACM, 2008 (= ICEGOV '08) hURL: <http://doi.acm.org/10.1145/1509096.1509143> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-386-0, pp. 232-238
- Bake, Pio: Open Source Software, Competition and Potential Entry; Working Paper; 2003, Web/Pdf hURL: <http://www.berlecon.de/tw/osscompetition.pdf>
- Bakker, Arno, Maarten Van Steen, a. Andrew S. Tanenbaum: A Wide-Area Distribution Network for Free Software; in: Transaction on Internet Technology, 6 August (2006), No. 3, pp. 259-281 hURL: <http://doi.acm.org/10.1145/1151087.1151089> - reference download: 2011-12-29, BibWeb/PDF
- Baldi, Stefan, Hauke Heier, a. Anett Mehler-Bicher: Open Courseware and Open Source Software; in: Communications of the ACM, 46 September (2003), No. 9, pp. 105-107 hURL: <http://doi.acm.org/10.1145/903893.903922> - reference download: 2011-12-29, BibWeb/PDF
- Barcellini, Flore, Francoise Detienne, a. Jean Marie Burkhardt: Cross-Participants: Fostering Design-Use Mediation in an Open Source Software Community; in: Proceedings of the 14th European Conference on Cognitive Ergonomics: Invent! Explore! New York, NY, USA: ACM, 2007 (= ECCE '07) hURL: <http://doi.acm.org/10.1145/1362550.1362564>, ISBN 978-1-84799-849-1, pp. 57-64
- Barcellini, Flore et al.: Thematic Coherence and Quotation Practices in OSS Design-Oriented Online Discussions; in: Proceedings of the 2005 International ACM SIGGROUP Conference

on Supporting Group Work; New York, NY, USA: ACM, 2005 (= GROUP '05)  
hURL: <http://doi.acm.org/10.1145/1099203.1099237>i - reference download: 2012-02-01,  
BibWeb/PDF, ISBN 1-59593-223-2, pp. 177-186

- Baytersdorfer, Mitch: Managing a Project With Open Source Components; in: *interactions*, 14 November/December (2007), pp. 33-34 hURL: <http://doi.acm.org/10.1145/1300655.1300677>i - reference download: 2011-12-29, BibWeb/PDF
- Baytiyeh, Hoda a. Jay Pfaman: Open source software: A community of altruists; in: *Computers in Human Behavior*, 26 (2010), p. 1345-1354, BibWeb/PDF
- Beard, Ashley a. Hyunju Kim: A Survey On Open Source Software Licenses; in: *JCSC*, 22 (2007), No. 4, pp. 205-211 hURL: <http://dl.acm.org/citation.cfm?id=1229637.1229673>i
- Behlendorf, Brian: Open Source as a Business Strategy; In Dibona, Ockman, a. Stone: *Open Sources*, 1999, pp. 149-170
- Behlendorf, Brian: How Open Source Can Still Save the World; conference contribution; In
- Boldyre et al.: *Open Source Ecosystems*, 2009, p. 2, BibWeb/PDF
- Beigbeder, Michel, Wray Buntine, a. Wai Gen Yee: Open Source Search and Research; in: *Proceedings of the 2006 International Workshop on Research Issues in Digital Libraries*; New York, NY, USA: ACM, 2007 (= IWRIDL '06) hURL: <http://doi.acm.org/10.1145/1364742.1364748>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-608-4, pp. 5:1-5:7
- Berglund, Erik a. Michael Priestley: Open-Source Documentation: In Search of User-Driven, Just-in-Time Writing; in: *Proceedings of the 19th annual international conference on Computer documentation*; New York, NY, USA: ACM, 2001 (= SIGDOC '01) hURL: <http://doi.acm.org/10.1145/501516.501543>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-295-6, pp. 132-141
- Bergquist, Magnus a. Jan Ljungberg: The power of gifts: organizing social relationships in open source communities; in: *Information Systems Journal*, 11 (2001), pp. 305-320
- Berlecon[-]Research: Basics of Open Source Software Markets and Business Models; FLOSS Final Report - Part 3; 2002, Web/Pdf hURL: [http://www.berlecon.de/studien/downloads/200207FLOSS\\_Basics.pdf](http://www.berlecon.de/studien/downloads/200207FLOSS_Basics.pdf)

- Berlecon[-]Research: Firms Open Source Activities: Motivations and Policy Implications; FLOSS Final Report - Part 2; 2002, Web/Pdf hURL: [http://www.berlecon.de/studien/downloads/200207FLOSS\\_Activities.pdf](http://www.berlecon.de/studien/downloads/200207FLOSS_Activities.pdf)
- Berry, David M.: Copy, Rip, Burn; The Politics of Copyleft and Open Source; London: Pluto Press, 2008, Print, ISBN 978-0-7453-2414-2
- Bessen, James: What Good is Free Software? In Hahn: Government Policy toward Open Source Software, 2002, pp. 12-34
- Bessen, James: Holdup and licensing of cumulative innovations with private information; in: Economics Letters, 82 (2004), No. 3, pp. 321-326
- Bessen, James a. Robert M. Hunt: An empirical look at software patents; in: Federal Reserve Bank of Philadelphia, Working Paper 03-17 (2004)
- Bezroukov, Nikoplai: BSD vs. GPL: Part 2: The Dynamic Properties of BSD and GPL Licenses in the Context of the Program Life Cycle; 2003 hURL: [http://www.softpanorama.org/Copyright/License-classification/social\\_dynamics\\_of\\_BSD\\_and\\_GPL.shtml](http://www.softpanorama.org/Copyright/License-classification/social_dynamics_of_BSD_and_GPL.shtml)
- Bhattacharya, Jaijit a. Sourabh Suman: Analysis of popular open source licenses and their applicability to e-governance; in: Proceedings of the 1st international conference on Theory and practice of electronic governance; New York, NY, USA: ACM, 2007 (= ICEGOV '07) hURL: <http://doi.acm.org/10.1145/1328057.1328110> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-59593-822-0, pp. 254-257
- Bianco, Vieri del et al.: The QualiSPo approach to OSS product quality evaluation; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 23-28 hURL: <http://doi.acm.org/10.1145/1833272.1833277> - reference download: 2011-12-29, BibWeb/PDF
- Bieman, James: Editorial: Free/open source software, silver bullets, and mythical months; in: Software Quality Journal, 14 (2006), pp. 289-290 hURL: <http://dx.doi.org/10.1007/s11219-006-0036-3> - reference download: 2012-202-03, BibWeb/PDF
- Bird, Christian, Alex Gourley, a. Prem Devanbu: Detecting Patch Submission and Acceptance in OSS Projects; in: Proceedings of the Fourth International Workshop on Mining

Software Repositories; Washington, DC, USA: IEEE Computer Society, 2007 (= MSR '07) hURL: <http://dx.doi.org/10.1109/MSR.2007.6i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 0-7695-2950-X, pp. 26:1-26:4

- Bird, Christian et al.: Open Borders? Immigration in Open Source Projects; in: Proceedings of the Fourth International Workshop on Mining Software Repositories; Washington, DC, USA: IEEE Computer Society, 2007 (= MSR '07) hURL: <http://dx.doi.org/10.1109/MSR.2007.23i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 0-7695-2950-X, pp. 6:1-6:8
- Bird, Christian et al.: Latent Social Structure in Open Source Projects; in: Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of Software Engineering; New York, NY, USA: ACM, 2008 (= SIGSOFT '08/FSE-16) hURL: <http://doi.acm.org/10.1145/1453101.1453107i> - reference download: 2012-01-02, BibWeb/PDF, ISBN 978-1-59593-995-1, pp. 24-35
- Bitzer, Jürgen, Wolfram Schrettl , a. Philipp J.H. Schröder: Intrinsic motivation in open source software development; in: Journal of Comparative Economics, 35 (2007), p. 160-169, BibWeb/PDF
- Björgvinsson, Tryggvi a. Helgi Thorbergsson: Software Development for Governmental Use Utilizing Free and Open Source Software; in: Proceedings of the 1st International Conference on Theory and Practice of Electronic Governance; New York, NY, USA: ACM, 2007 (= ICEGOV '07) hURL: <http://doi.acm.org/10.1145/1328057.1328087i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-59593-822-0, pp. 133-140
- Black Duck: Top 20 Open Source Licenses; 2014 [n.y], FreeWeb/HTML hURL: <http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses> - reference download: 2014-02-11
- Böckler, Lina: Mit Freier Software gegen den Wettbewerb; in: Katharina Vera Boesche, editor : Variationen im Recht: Beiträge zum Arbeits-, Immaterialgüter-, Infrastruktur-, Lauterkeits-, Unternehmens-, Wettbewerbs- und Zivilrecht; Festbeigabe für Fanz Jürgen Säcker zum 65. Geburstag; Berlin: BWV, 2006, ISBN 3-8305-1234-1, pp. 69 - 76, BibWeb/Copy
- Boehm, Barry: A View of 20th and 21st Century Software Engineering; in: Proceedings of the 28th International Conference on Software Engineering; New York, NY, USA: ACM, 2006 (= ICSE '06) hURL: <http://doi.acm.org/10.1145/1134285.1134288i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-375-1, pp. 12-29

- Boldrin, Michele a. David K. Levine: The Case Against Intellectual Property; in: American Economic Review, 92 (2002), No. 2, pp. 209-212
- Boldrito, Remo Suppi a. Josep Jorba Esteve: GNU/Linux Advanced Administration; coordinated by Josep Jorba Esteve; Barcelona: Free Technology Academy, 2007 hURL: <http://www.ftacademy.org/i> - reference download: 2012-01-20, FreeWeb/PDF
- Boldyre, Cornelia et al., editors: Open Source Ecosystems: Diverse Communities Interaction; 5th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2009 Skövde, Sweden, June 3-6, 2009; Berlin, Heidelberg and New York: Springer, 2009, BibWeb/PDF, ISBN 978-3-642-02031-5
- Bolzern, Mark: A New Project or a GNU Project? in: Linux Journal, 13 May (1995), p. 7:1 hURL: <http://dl.acm.org/citation.cfm?id=324822.324829i> - reference download: 2011-12-29, BibWeb/HTML
- Bonaccorsi, A., M. Merito ans L. Piscitello, a. C. Rossi: The 'Open Innovation' Paradigm. How Firms Do Business out of Open Source Software; Copenhagen, 2006, Paper presented at the DRUID Summer Conference
- Bonaccorsi, A. a. C. Rossi: Contributing to the Common Pool Resources in Open Source Software. A Comparison between Individuals and Firms. Pisa, 2003, Sant' Anna School of Advanced Studies; Working Paper
- Bonaccorsi, A. a. C. Rossi : Why Open Source Software Can Succeed; in: Research Policy, 32 (2003), No. 7, pp. 1243-1258
- Bonaccorsi, A. a. C. Rossi: Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement. From Community to Business; Pisa, 2004, Sant' Anna School of Advanced Studies; Working Paper hURL: <http://opensource.mit.org/i>
- Bonaccorsi, Andrea et al.: Business Firms' Engagement in Community Projects. Empirical Evidence and Further Developments of the Research; in: Proceedings of the First International Workshop on Emerging Trends in FLOSS Research and Development; Washington, DC, USA: IEEE Computer Society, 2007 (= FLOSS '07) hURL: <http://dx.doi.org/10.1109/FLOSS.2007.3i>, ISBN 0-7695-2961-5, pp. 13-
- Booth, David R.: Peer Production and Software. What Mozilla Has To Teach Government; Cambridge (Massachusetts) and London (England): MIT Press, 2010 (= The John D. and

Catherine T. MacArthur Foundation Reports on Digital Media and Learning), Print, ISBN 978-0-262-51461-3

- Bresson, Jean, Carlos Agon, a. Gerard Assayag: OpenMusic; Visual Programming Environment for Music Composition, Analysis and Research; in: Proceedings of the 19th ACM International Conference on Multimedia; New York, NY, USA: ACM, 2011 (= MM '11) hURL: <http://doi.acm.org/10.1145/2072298.2072434>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0616-4, pp. 743-746
- Bretschneider, Ulrich, Rainer Glaschick, a. Gernot Gräfe: Ratgeber für die Veröffentlichung von Open-Source-Software durch eine Hochschule; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 167-188, Print
- Brodie, Mark et al.: Support services: persuading employees and customers to do what is in the community's best interest; in: Proceedings of the 2nd international conference on Persuasive technology; Berlin, Heidelberg: Springer-Verlag, 2007 (= PERSUASIVE'07) hURL: <http://dl.acm.org/citation.cfm?id=1780402.1780424>i, ISBN 3-540-77005-4, 978-3-540-77005-3, pp. 121-124
- Brown, Peter: EOF: The Free Software Foundation at 20; in: Linux Journal, 137 September (2005), p. 15:1 hURL: <http://dl.acm.org/citation.cfm?id=1084783.1084798>i - reference download: 2011-12-29, BibWeb/HTML
- Brügge, Bernd et al.: Open-Source Software. Eine ökonomische und technische Analyse; Berlin and Heidelberg: Springer, 2004, Print, ISBN 3-540-20366-4
- Buchtala, Rouven: Determinanten der Open Source Software-Lizenzwahl. Eine spieltheoretische Analyse; Frankfurt am Main, Berlin, Bern [... etc.]: Peter Lang, 2007 (= Informationsmanagement und strategische Unternehmensführung), [Vol./No.] 12), Print, ISBN 978-3-631-57114-9
- Burgess, Guy: Open Source: The Aero General Public License; in: Magazine of the Society for Computers and Law, 19 (2008), No. 4, pp. 42-43, Copy
- Bygott, David: David Bygott's Gnu Book. A light-hearted look at the Gnu, or Wildebeest; first published 1998; Southerton, Harare: Robert Woollacott, 1992, Print, ISBN 0-7974-1082-1
- Bärwol, M.: Tight Prior Open Source Equilibrium: The Rise of Open Source as a Source of Economic Welfare; in: First Monday, 11 (2006) hURL: <http://firstmonday.org/issues/>

- Camp, L. Jean: DRM: Doesn't Really Mean Digital Copyright Management; in: Proceedings of the 9th ACM Conference on Computer and Communications Security; New York, NY, USA: ACM, 2002 (= CCS '02) hURL: <http://doi.acm.org/10.1145/586110.586122i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-612-9, pp. 78-87
- Campbell-Kelly, Martin: Historical Reflections: Will the Future of Software be Open Source? in: Communications of the ACM, 51 October (2008), No. 10, pp. 21-23 hURL: <http://doi.acm.org/10.1145/1400181.1400189i> - reference download: 2011-12-29, BibWeb/PDF
- Capiluppi, Andrea, Andres Baravalle, a. Nick W. Heap: From "Community" to "Commercial" FLOSS - the Case of Moodle; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 11-16 hURL: <http://doi.acm.org/10.1145/1833272.1833275i> - reference download: 2012-02-01, BibWeb/PDF
- Capiluppi, Andrea a. Thomas Knowles: Software Engineering in Practice: Design and Architecture of FLOSS Systems; 5th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2009 Skövde, Sweden, June 3-6, 2009; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 34-46, BibWeb/PDF
- Chawner, Brenda: F/OSS in the Library World: An Exploration; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 3:1-3:4 hURL: <http://doi.acm.org/10.1145/1082983.1083262i> - reference download: 2011-12-29, BibWeb/PDF
- Cheliotis, Giorgos: From open source to open content: Organization, licensing and decision processes in open cultural production; in: Decision Support Systems, 47 (2009), No. 3, pp. 229-244 hURL: <http://www.sciencedirect.com/science/article/pii/S0167923609000578i> - reference download: 2012-02-01, BibWeb/PDF
- Chen, Shun-ling: Free/Open Source Software. Licensing; Shri Pratap Udyog, Srinivas Puri, New Delhi: Elsevier India, 2006 hURL: <http://www.iosn.net/licensing/foss-licensing-primer/foss-licensing-final.pdfi> - reference download: 2013-02-02, FreeWeb/PDF, ISBN 978-81-312-0422-1
- Chen, Zhixiong a. Delia Marx: Experiences with Eclipse IDE in Programming Courses; in: JCSC, 21 December (2005), No. 2, pp. 104-112 hURL: <http://dl.acm.org/citation.cfm?id=1089053.1089068i> - reference download: 2011-12-29, BibWeb/PDF

- Cheung, Giord et al.: Designing for Discovery: Opening the Hood for Open-Source End User Tinkering; in: Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems; New York, NY, USA: ACM, 2009 (= CHI EA '09) hURL: <http://doi.acm.org/10.1145/1520340.1520660> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-247-4, pp. 4321-4326
- Chindalia, Sanjanaa: Open source software: The future ahead; in: JOURNAL OF INTELLECTUAL PROPERTY RIGHTS, 13 (2008), pp. 218-224
- Chopra, S. a. S. Dexter: Free software and the political philosophy of the cyborg world; in: SIGCAS, 37 November (2007), No. 2, pp. 41-52 hURL: <http://doi.acm.org/10.1145/1327325.1327328> - reference download: 2011-12-29, BibWeb/PDF
- Chopra, S. a. S. Dexter: Free software, economic 'realities', and information justice; in: SIGCAS, 39 December (2009), No. 3, pp. 12-26 hURL: <http://doi.acm.org/10.1145/1713066.1713067> - reference download: 2011-12-29, BibWeb/PDF
- Chopra, Samir a. Scott Dexter: The freedoms of software and its ethical uses; in: Ethics and Information Technology, 11 (2009), pp. 287-297 hURL: <http://dx.doi.org/10.1007/s10676-009-9191-0>, BibWeb/PDF
- Christ, Fabian a. Stefan Sauer: OSS - Open-Source-Stacks; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 133-154, Print
- Ciolilli, Andrea: The Economics of Open Source Hijacking and Declining Quality of Digital Information Resources: A Case for Copyleft; 2004 hURL: <http://opensource.mit.edu/papers/ciffolili.pdf>
- Coar, Ken a. Rich Bowen: Apache Kochbuch; deutsche Übersetzung v. Jochen Wiedmann; Beijing [...]: O'Reilly, 2004, Print, ISBN 3-89721-371-0
- Colazo, Jorge a. Yulin Fang: Impact of License Choice on Open Source Software Development Activity; in: Journal of the American Society for Information Science and Technology, 60 (2009), No. 5, pp. 997-1011, BibWeb/PDF
- Coleman, Gabriella: CODE IS SPEECH: Legal Tinkering, Expertise, and Protest among Free and Open Source Software Developers; in: Cultural Anthropology, 24 (2009), No. 3,

pp. 420-454 hURL: <http://dx.doi.org/10.1111/j.1548-1360.2009.01036.xi> - reference  
download: 2012-02-03, BibWeb/PDF, ISSN 1548-1360

- Comino, Stefano a. Fabio M. Manenti: Dual licensing in open source software markets; in: *Information Economics and Policy*, 23 (2011), No. 3-4, pp. 234-242 hURL: <http://www.sciencedirect.com/science/article/pii/S016762451100028Xi> - reference download: 2012-02-01, BibWeb/PDF
- Comino, Stefano, Fabio M. Manenti, a. Maria Laura Parisi: From planning to mature: On the success of open source projects; in: *Research Policy*, 36 (2007), No. 10, pp. 1575-1586 hURL: <http://www.sciencedirect.com/science/article/pii/S004873307001709i> - reference download: 2012-02-01, BibWeb/PDF
- copyleft.org: What is copyleft.org; n.l, 2014, FreeWeb/HTML hURL: <http://copyleft.org/i> - reference download: 2014-12-15
- Costa-Soria, Cristobal a. Jennifer Perez : Teaching Software Architectures and Aspect-Oriented Software Development using Open-Source Projects; in: Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education; New York, NY, USA: ACM, 2009 (= ITiCSE '09) hURL: <http://doi.acm.org/10.1145/1562877.1563027i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-381-5, p. 385
- Crowston, Kevin et al.: Eective Work Practices for Software Engineering: Free/Libre Open Source Software Development; in: Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research; New York, NY, USA: ACM, 2004 (= WISER '04) hURL: <http://doi.acm.org/10.1145/1029997.1030003i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-988-8, pp. 18-26
- Crowston, Kevin a. James Howison: The social structure of Free and Open Source software development; in: *First Monday* 10 (2005), No. 2
- Crowston, Kevin et al.: Self-organization of teams for free/libre open source software development; in: *Information and Software Technology*, 49 (2007), No. 6, pp. 564 - 575 hURL: <http://www.sciencedirect.com/science/article/pii/S0950584907000080i>, <ce:title>Qualitative Software Engineering Research</ce:title>, ISSN 0950-5849
- Cuellar, Luis E.: Open Source License Alternatives for Software Applications; Is it a solution to stop software piracy?; in: Proceedings of the 43rd Annual Southeast Regional Conference; Volume 2, New York, NY, USA: ACM, 2005 hURL: <http://doi.acm.org/10.1145/1167253>.

1167314i - reference download: 2011-12-28, BibWeb/PDF, ISBN 1-59593-059-0, pp. 269-274

- Currión, Paul, Chamindra de Silva, a. Bartel Van de Walle: Open Source Software For Disaster Management; in: Communications of the ACM, 50 March (2007), No. 3, pp. 61-65 hURL: <http://doi.acm.org/10.1145/1226736.1226768>i - reference download: 2011-12-29, BibWeb/PDF
- Cusumano, Michael A.: Reflections on Free and Open Software; in: Communications of the ACM, 47 October (2004), No. 10, pp. 25-27 hURL: <http://doi.acm.org/10.1145/1022594>. 1022615i - reference download: 2012-202-03, BibWeb/PDF
- Dahlander, L.: Appropriation and Appropriability in Open Source Software; in: International Journal of Innovation Management, 9 (2005), No. 3, pp. 259-285
- Dahlander, Linus: Penguin in a new suit: a tale of how de novo entrants emerged to harness free and open source software communities; in: Industrial and Corporate Change, 16 (2007), No. 5, pp. 913-943 hURL: <http://icc.oxfordjournals.org/content/16/5/913>. abstracti - reference download: 2012-202-03, BibWeb/PDF
- Dalle, Jean-Michel et al.: Advancing Economic Research on the Free and Open Source Software Mode of Production; In Wynants a. Cornelius: How Open is the Future?, 2005, pp. 395-426
- David, Paul A. a. Francesco Rullani : Dynamics of innovation in an "open source" collaboration environment: lurking, laboring, and launching FLOSS projects on SourceForge; in: Industrial and Corporate Change, 17 (2008), No. 4, pp. 647-710, BibWeb/PDF
- Davies, Julius: Measuring Subversions: Security and Legal Risk in Reused Software Artifacts; in: Proceedings of the 33rd International Conference on Software Engineering; New York, NY, USA: ACM, 2011 (= ICSE '11) hURL: <http://doi.acm.org/10.1145/1985793.1986025>i - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-4503-0445-0, pp. 1149-1151
- Davis, Donald a. Iat Jabeen: Learning in the GNU/Linux Community; in: Proceedings of the 2011 Conference on Information Technology Education; New York, NY, USA: ACM, 2011 (= SIGITE '11) hURL: <http://doi.acm.org/10.1145/2047594.2047600>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-1017-8, pp. 21-26
- Davis, Mike et al.: Linux and Open Source in the Academic Enterprise; in: Proceedings of the 28th annual ACM SIGUCCS conference on User services: Building the future; New York, NY, USA: ACM, 2000 (= SIGUCCS '00) hURL: <http://doi.acm.org/10.1145/354908>.

354923i - reference download: 2011-12-28, BibWeb/PDF, ISBN 1-58113-229-8, pp. 65-69  
de Laat, Paul B.: Copyright or copyleft? An analysis of property regimes for software development; in: Research Policy, 34 (2005), pp. 1511-1532, BibWeb/PDF

- De Nicolo, Christopher: Open Source Software - Rechtliche Aspekte nach deutschem und italienischem Recht. Eine rechtsvergleichende Studie. Dissertation; Regensburg: Universität Regensburg, 2010, Print
- Debian: The Debian Free Software Guidelines (DFSG); 2013 [n.y.], FreeWeb/HTML hURL: [http://www.debian.org/social\\_contract#guidelines](http://www.debian.org/social_contract#guidelines) - reference download: 2013-01-22
- Deike, Thies: Open Source Software: IPR-Fragen und Einordnung ins deutsche Rechtssystem; in: CR [Computer und Recht], (2003), pp. 9
- Deitcher, Avi : The challenges of open source in the enterprise; in: Linux Journal, 195 July (2010), pp. Article No. 3 hURL: <http://dl.acm.org/citation.cfm?id=1883478.1883481> i - reference download: 2011-12-28, BibWeb/HTML
- Dempsey, Bert J. et al.: A quantitative profile of a community of Open Source Linux developers; North Carolina: University of North Carolina at Chapel Hill, School of Information and Library Science, 1999 (= (= [University of North Carolina] Technical Report TR 1999-05))
- Dempsey, Bert J. et al.: Who Is an Open Source Software Developer? in: Communications of the ACM, 45 (2002), No. 2, pp. 67-72 hURL: <http://doi.acm.org/10.1145/503124.503125> i - reference download: 2011-12-28, BibWeb/PDF
- Deodhar, Swanand J., K. B. C. Saxena, a. Mikko Ruohonen: Firm-Oriented Success Factors of an Open Source Software (OSS) Product; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 1-4 hURL: <http://doi.acm.org/10.1145/1833272.1833273> i - reference download: 2011-12-29, BibWeb/PDF
- Determann, Lothar: Softwarekombinationen unter der GPL; in: GRUR Int. (slg: Gewerblicher Rechtsschutz und Urheberrecht, Internationaler Teil, 2006), (2006), pp. 645 - 653
- Di Penta, Massimiliano et al.: An Exploratory Study of the Evolution of Software Licensing; in: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering; Volume 1, New York, NY, USA: ACM, 2010 hURL: <http://doi.acm.org/10.1145/1806799.1806824> i - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-719-6, pp. 145-

- DiBona, Chris, Cooper C, a. D. Stone, editors: Open Sources 2.0: The Continuing Evolution; Sebastopol CA: O'Reilly, 2005
- DiBona, Chris, Sam Ockman, a. Mark Stone, editors: Open Sources. Voices from the Open Source Revolution; Beijing u.a.: O'Reilly, 1999
- Diedrich, Oliver : Die Geschichte von Linux; 2011, FreeWeb/PDF hURL: <http://www.heise.de/open/artikel/Die-Geschichte-von-Linux-1329997.html> - reference download: 20110826
- Dionisio, John David N. et al.: An Open Source Software Culture in the Undergraduate Computer Science Curriculum; in: SIGCSE Bulletin, 39 June (2007), No. 2, pp. 70-74 hURL: <http://doi.acm.org/10.1145/1272848.1272881> - reference download: 2011-12-29, BibWeb/PDF
- Djordjevic, Valle et al., editors: Urheberrecht im Alltag. Kopieren, bearbeiten, selber machen; Bonn: Bundeszentrale für politische Bildung, 2008, Print, ISBN 978-3-89331-812-4
- Dobb, Dr.: It All About The License; in: Informationweek, n.V. (2009), No. 1253, p. 46, Copy
- Doernhoefer, Mark: Surng the Net for Software Enginerring Notes; in: SIGSOFT Software Engineering Notes, 35 (2010), No. 4, pp. 8-16, BibWeb/PDF
- Donnelly, Francis P.: Evaluating open source GIS for libraries; in: Library Hi Tech, 28 (2010), No. 1, pp. 131-151 hURL: <http://www.emeraldinsight.com/0737-8831.html> - reference download: 2012-02-13, BibWeb/PDF
- Donoro, Brian: The Politics of "Free": Open Source Software in Government; in: JCSC, 19 (2004), No. 5, pp. 279-280 hURL: <http://dl.acm.org/citation.cfm?id=1060081.1060117> - reference download: 2011-12-29, BibWeb/PDF
- Dorman, David: The Case for Open Source Software in the Library Market; in: Ubiquity, January (2004), p. 4:1 hURL: <http://doi.acm.org/10.1145/985600.985601> - reference download: 2011-12-29, BibWeb/HTML
- Dougherty, William C. a. Audrey Schadt : Linux Is for Everyone; Librarians Included! in: The Journal of Academic Librarianship, 36 (2010), No. 2, pp. 173-175 hURL: <http://>

//www.sciencedirect.com/science/article/pii/S0099133310000108i - reference download:  
2012-02-09, BibWeb/PDF

- Douglas, David: A bundle of software rights and duties; in: Ethics and Information Technology, 13 (2011), pp. 185-197 hURL: <http://dx.doi.org/10.1007/s10676-010-9229-3> - reference download: 2012.02.09, BibWeb/PDF
- Drossou, Olga, Stefan Kreml, a. Andreas Poltermann: Der Kampf um die Innovationsfreiheit: Der Big Bang des Wissens und seine Sprengkraft. Plädoyer für einen oenen Umgang mit Wissen im Interesse der Innovationskraft von Wirtschaft und Gesellschaft; Editorial; In
- Drossou, Kreml, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 1-10, Print
- Drossou, Olga, Stefan Kreml, a. Andreas Poltmann, editors: Die wunderbare Wissensvermehrung. Wie Open Innovation unsere Welt revolutioniert; (= Telepolis) Hannover: Heise, 2006, Print, ISBN 3-936931-38-0
- Eckl, Julian: Die politische ökonomie der Wissenschaftsgesellschaft. Geistige Eigentumsrechte und die Frage des Zugangs zu Ideen; Marburg: Tectum Verlag, 2004, Print, ISBN 3-8288-8735-X
- Eclipse Foundation: Eclipse Public License, Version 1.0; 2005 [n.y. of the page itself], FreeWeb/HTML hURL: <http://www.eclipse.org/org/documents/epl-v10.php> - reference download: 2013-02-20
- Eclipse Foundation: CPL to EPL Conversion; 2013 [n.y. of the page itself], FreeWeb/HTML hURL: <http://www.eclipse.org/legal/cpl2epl/i> - reference download: 2013-02-20
- Economides, Nicholas a. Evangelos Katsamakas: Two-Sided Competition of Proprietary vs. Open Source Technology Platforms and the Implications for the Software Industry; in: Management Science, 52 (2006), No. 7, pp. 1057-1071
- Elkemann-Reusch, Ilva: Die erzwungene Gegengabe; in: ZGE / IPJ, 2 (2010), pp. 413-452, Copy
- Elliott, Margaret, Mark S. Ackerman, a. Walt Scacchi: Knowledge Work Artifacts: Kernel Cousins for Free/Open Source Software Development; in: Proceedings of the 2007 international ACM conference on Supporting group work; New York, NY, USA: ACM, 2007 (= GROUP '07) hURL: <http://doi.acm.org/10.1145/1316624.1316650> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-59593-845-9, pp. 177-186

- Elliott, Margaret S. a. Walt Scacchi: Free Software Developers as an Occupational Community: Resolving Conflicts and Fostering Collaboration; in: Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work; New York, NY, USA: ACM, 2003 (= GROUP '03) hURL: <http://doi.acm.org/10.1145/958160.958164> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-693-5, pp. 21-30
- Ellis, Jason a. Jean-Paul Van Belle: Open Source Software Adoption by South African MSEs: Barriers and Enablers; in: Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association; New York, NY, USA: ACM, 2009 (= SACL A '09) hURL: <http://doi.acm.org/10.1145/1562741.1562746> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-683-0, pp. 41-49
- Engelfriet, Arnoud: Tools of the Trade[:] Choosing an Open Source License; in: IEEE Software, 27 (2010), No. 1, pp. 48-49, Copy
- Epplin, J.: Using GPL Software in Embedded Applications; hURL: <http://www.linuxdevices.com/articles/AT916119242.html>
- Ernst, Stefan: Die Verfügbarkeit des Sourcecodes; in: MultiMedia und Recht, 4 (2001), pp. 208-213
- Euler, Ellen: Creative Commons: Mehr Innovation durch die Önung des Urheberechts? In Drossou, Krempel, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 147-158, Print
- European Community a. European commission Joinup: European Union Public Licence v. 1.1. 2007, FreeWeb/HTML hURL: <http://joinup.ec.europa.eu/system/files/EN/EUPL%20v.1.1%20-%20Licence.pdf> - reference download: 2013-02-08
- European Community a. European commission Joinup: New FSF statements on the EUPL are a step in the right direction; 2013 [n.y], FreeWeb/HTML hURL: <https://joinup.ec.europa.eu/community/eupl/news/new-fsf-statements-eupl-are-step-right-direction> - reference download: 2013-03-05
- Europäische Gemeinschaft a. European commission Joinup: Open-Source-Lizenz für die Europäische Union; 2007, FreeWeb/HTML hURL: <http://joinup.ec.europa.eu/system/files/DE/EUPL%20v.1.1%20-%20Lizenz.pdf> - reference download: 2013-02-08

- Evans, David S. a. Bernard J. Reddy: Government Preferences for Promoting Open-Source Software: A Solution in Search of a Problem; in: 9 Mich Telecomm. Tech. L. Rev. 313 (2003), pp. 313-394
- Ezeala, Adanna, Hyunju Kim, a. Loretta A. Moore: Open Source Software Development: Expectations and Experience from a Small Development Project; in: Proceedings of the 46th Annual Southeast Regional Conference on XX; New York, NY, USA: ACM, 2008 (= ACM-SE 46) hURL: <http://doi.acm.org/10.1145/1593105.1593168i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-105-7, pp. 243-246
- Fantl, Stephen: Copyleft or Copyright. Into the new paradigm; in: MacTech Magazine, 16 (2000), No. 10, pp. 98-100, Copy
- Fehr, Ernst a. Simon Gächter: Fairness and Retaliation: The Economics of Reciprocity; in: Journal of Economic Perspectives, 14 (2000), pp. 159-181
- Feig, Michael: Einführung in GNU; München and Wien: Carl Hanser Verlag, 1996 (= Unix easy), Print, ISBN 3-446-18311-6
- Feller, Joseph: Meeting challenges and surviving success: the 2nd workshop on open source software engineering; evaluate the complete workshop results; in: Proceedings of the 24th International Conference on Software Engineering; New York, NY, USA: ACM, 2002 (= ICSE '02) hURL: <http://doi.acm.org/10.1145/581339.581436i>, ISBN 1-58113-472-X, pp. 669-670
- Feller, Joseph a. Brian Fitzgerald: A Framework Analysis of the Open Source Software Development Paradigm; in: Proceedings of the twenty first international conference on Information systems; Atlanta, GA, USA: Association for Information Systems, 2000 (= ICIS '00) hURL: <http://dl.acm.org/citation.cfm?id=359640.359723i>, BibWeb/PDF, pp. 58-69
- Feller, Joseph et al.: Collaboration, Conflict and Control: The 4th Workshop on Open Source Software Engineering; evaluate worksup results; in: Proceedings of the 26th International Conference on Software Engineering; Washington, DC, USA: IEEE Computer Society, 2004 (= ICSE '04) hURL: <http://dl.acm.org/citation.cfm?id=998675.999508i>, ISBN 0-7695-2163-0, pp. 764-765
- Feller, Joseph et al.: Collaboration, conflict and control: report on the 4th workshop on open source software engineering; in: SIGSOFT Softw. Eng. Notes, 30 May (2005), pp. 1-2 hURL:

- Feller, Joseph et al.: Taking stock of the bazaar: the third workshop on open source software engineering; in: SIGSOFT Softw. Eng. Notes, 28 November (2003), pp. 5-5 hURL: <http://doi.acm.org/10.1145/966221.966227>, ISSN 0163-5948
- Feller, Joseph, Brian Fitzgerald, a. Andre van der Hoek: 1st workshop on open source software engineering; evaluate workshop results; in: Proceedings of the 23rd International Conference on Software Engineering; Washington, DC, USA: IEEE Computer Society, 2001 (= ICSE '01) hURL: <http://dl.acm.org/citation.cfm?id=381473.381660>, ISBN 0-7695-1050-7, pp. 780-781
- Feller, Joseph, Brian Fitzgerald, a. Andre van der Hoek: Making sense of the bazaar: 1st workshop on open source software engineering; in: SIGSOFT Softw. Eng. Notes, 26 November (2001), pp. 51-52 hURL: <http://doi.acm.org/10.1145/505532.505543>, ISSN 0163-5948
- Feller, Joseph et al.: Open source application spaces: the 5th workshop on open source software engineering; evaluatre workshop results; in: Proceedings of the 27th international conference on Software engineering; New York, NY, USA: ACM, 2005 (= ICSE '05) hURL: <http://doi.acm.org/10.1145/1062455.1062619>, ISBN 1-58113-963-2, pp. 694-694
- Feller, Jospeph a. Brian Fitzgerald: Understanding Open Source Software Development; Reading, Mass., London?: Addison-Wesley, 2002
- Fielding, Roy T.: Shared leadership in the Apache Project; in: Communications of the ACM, 42 (1999), No. 4, pp. 42-43
- Fielding, Roy T.: Software Architecture in an Open Source World; in: Proceedings of the 27th International Conference on Software Engineering; New York, NY, USA: ACM, 2005 (= ICSE '05) hURL: <http://doi.acm.org/10.1145/1062455.1062474> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-963-2, pp. 43-43
- Fink, Martin: The business and economics of Linux and open source; Upper Saddle River, N.J.: Prentice Hal, 2003
- Fitzgerald, Brian: The Transformation of Open Source Software; in: MIS Quarterly, 30 (2006), No. 3, pp. 587-598
- Fogel, Karl: Producing Open Source Software; How to Run a Successful Free Software Project;

Beijing, Cambridge, Köln [...]: O'Reilly, 2006, Print, ISBN 978-0-596-00759-1

- Fosfuri, Andrea, Marco S. Giarratana, a. Alessandra Luzzi: The Penguin Has Entered the Building: The Commercialization of Open Source Software Products; in: *OrganizationScience*, 19 March-April (2008), No. 2, p. 292-305, BibWeb/PDF
- Fox, Laurie a. Shawn Plummer: Opening the Lines of Communications with Open Source Software; in: Proceedings of the 34th Annual ACM SIGUCCS Fall Conference; New York, NY, USA: ACM, 2006 (= SIGUCCS '06) hURL: <http://doi.acm.org/10.1145/1181216.1181242>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-438-3, pp. 114-117
- Franck, Egon a. Carola Jungwirth: Die Governance von Open Source Projekten; in: *Zeitschrift für Betriebswirtschaft*, 73 (2003), No. 5 (Ergänzungsheft), pp. 1 -21
- Franky, Marita Consuelo: Agile Management and Development of Software Projects based on Collaborative Environments; in: *SIGSOFT Software Engineering Notes*, 36 May (2011), No. 3, pp. 1-6 hURL: <http://doi.acm.org/10.1145/1968587.1968605>i - reference download: 2011-12-29, BibWeb/PDF
- Free Software Foundation: GNU General Public License, version 2; 1991 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://www.gnu.org/licenses/gpl-2.0.html>i - reference download: 2013-02-05
- Free Software Foundation: GNU Library General Public License [version 2.0]; 1991 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://www.gnu.org/licenses/old-licenses/lgpl-2.0.html>i - reference download: 2013-03-25
- Free Software Foundation: GNU Lesser General Public License [Version 2.1]; 1999 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://www.gnu.org/licenses/lgpl-2.1.html>i - reference download: 2013-03-06
- Free Software Foundation: GNU General Public License [version 3]; 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://www.gnu.org/licenses/gpl.html>i - reference download: 2013-03-06
- Free Software Foundation: GNU Lesser General Public License [version 3]; 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://www.gnu.org/copyleft/lesser.html>i - reference download: 2013-03-06

- Free Software Foundation: GNU Operating System[:] Licenses; 2011, FreeWeb/HTML hURL: <http://www.gnu.org/licenses/i> - reference download: 2013-03-25
- Free Software Foundation: Various Licenses and Comments about Them; 2013 [n.y.], FreeWeb/HTML hURL: <http://www.gnu.org/licenses/license-list.html> - reference download: 2013-02-08
- Free Software Foundation: What is free software? The Free Software Denition; 2015 [n.y.], FreeWeb/HTML hURL: <https://www.gnu.org/philosophy/free-sw.en.html> - reference download: 2015-02-20
- Friedman, Batya et al.: Development of a Privacy Addendum for Open Source Licenses: Value Sensitive Design in Industry; in: Paul Dourish a. Adrian Friday, editors: UbiComp 2006: Ubiquitous Computing; [Proceedings of the] 8th International Conference, UbiComp 2006 Orange County, CA, USA, September 17-21, 2006; Berlin, Heidelberg, a. New York: Springer, 2006 (= Lecture Notes in Computer Science, [Vol./No.] 4206), BibWeb/PDF, ISBN 978-3-540-39634-5, pp. 194-211
- Fujita, Kunihiko a. Yasuyuki Tsukada: An Analysis of Interoperability between Licenses; in: Proceedings of the Tenth Annual ACM Workshop on Digital Rights Management; New York, NY, USA: ACM, 2010 (= DRM '10) hURL: <http://doi.acm.org/10.1145/1866870.1866884> i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0091-9, pp. 61-72
- Funk, Axel a. Georg Zeitfang: Die GNU General Public License, Version3; in: CR [Computer und Recht], (2007), pp. 617 - 624
- Funk, Axel a. Gregor Zeitfang: Die GNU General Public License, Version 3; in: Computer und Recht, 23 (2007), No. 10, pp. 617-624, BibWeb/Copy
- Fuse Source Team, The: How to Use Open Source Integration Software Safely in the Enterprise. Analysis of potential risks and how to protect your IT environment; October 2010, FreeWeb/PDF [ File received by a promotion campaign of itwhitepapers.com. The article refers to <http://www.fusesource.com/>. But we didn't retrieve the paper there. ] hURL: [http://www.itwhitepapers.com/?option=com\\_category&report&task=viewabstract&pathway=no&autodn=1&title=14770&crv=0&src=5053&ctg=410&cmp=3732&yld=0&pi=1628115](http://www.itwhitepapers.com/?option=com_category&report&task=viewabstract&pathway=no&autodn=1&title=14770&crv=0&src=5053&ctg=410&cmp=3732&yld=0&pi=1628115) i - reference download: 2011-08-24
- Gallini, Nancy a. Suzanne Scotchmer: Intellectual Property: When Is It the Best Incentive

System? in: NBER Innovation Policy & the Economy, 2 (2002), No. 1, pp. 51-77

- Gambardella, Alfonso a. Bronwyn H. Hall : Proprietary versus public domain licensing of software and research products; in: RP, 35 (2006), No. 6, pp. 875-892 hURL: <http://www.sciencedirect.com/science/article/pii/S004873306000643i> - reference download: 2012-02-09, BibWeb/PDF
- Gandal, Neil a. Chaim Fershtman: Open Source Projectsd: Output per Contributor and Restrictive Licensing; o.O.: ???, 2004 (= (= CEPR Working Paper 2650))
- Gassmann, Oliver a. Martin A. Bader: Patentmanagement: Innovationen erfolgreich nutzen und schützen; Berlin: ???, 2005
- Geese, Elmar: Innovation und freie Software; In Drossou, Krempel, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 77-84, Print
- Gehring, Robert A. a. Bernd Lutterbeck, editors: Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell; Berlin: Lehmanns Media, 2004 hURL: <http://www.opensourcejahrbuch.de/download/jb2004/OpenSourceJahrbuch2004.pdfi> - reference download: 2011-08-29, Print & FreeWeb/PDF, ISBN 3-936427-78-X
- Gerber, Aurona, Onkgopotse Molefe, a. Alta van der Merwe: Documenting Open Source Migration Processes for Re-use; in: Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists; New York, NY, USA: ACM, 2010 (= SAICSIT '10) hURL: <http://doi.acm.org/10.1145/1899503.1899512i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-950-3, pp. 75-85
- Gerlach, Carsten: Praxisprobleme der Open-Source-Lizenzierung; in: CR [Computer und Recht], (2006), pp. 649 - 654
- German, Daniel M.: Using software distributions to understand the relationship among free and open source software projects; in: Proceedings of the Fourth International Workshop on Mining Software Repositories; Washington, DC, USA: IEEE Computer Society, 2007 (= MSR '07) hURL: <http://dl.acm.org/citation.cfm?id=1268983.1269038i>, BibWeb/PDF, ISBN 0-7695-2950-X, p. 24
- German, Daniel M. a. Jesus M. Gonzalez-Barahona: An Empirical Study of the Reuse of Software Licensed under the GNU General Public License; conference contribution; In

- German, Daniel M. a. Ahmed E. Hassan: License Integration Patterns: Addressing License Mismatches in Component-Based Development; in: Proceedings of the 31st International Conference on Software Engineering; Washington, DC, USA: IEEE Computer Society, 2009 (= ICSE '09) hURL: <http://dx.doi.org/10.1109/ICSE.2009.5070520i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-4244-3453-4, pp. 188-198
- German, Daniel M., Yuki Manabe, a. Katsuro Inoue: A Sentence-Matching Method for Automatic License Identification of Source Code Files; in: Proceedings of the IEEE/ACM International Conference on Automated Software Engineering; New York, NY, USA: ACM, 2010 (= ASE '10) hURL: <http://doi.acm.org/10.1145/1858996.1859088i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0116-9, pp. 437-446
- German, Daniel M., Jens H. Webber, a. Massimiliano Di Penta: Lawful Software Engineering; in: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research; New York, NY, USA: ACM, 2010 (= FoSER '10) hURL: <http://doi.acm.org/10.1145/1882362.1882390i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0427-6, pp. 129-132
- Gilroy, Bernard Michael a. Tobias Volpert: Die Funktionen eines Patentsystems und ihre Bedeutung für Unternehmensausgründungen aus Hochschulen; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 21-39, Print
- Gobeille, Robert: The FOSSology Project; in: Proceedings of the 2008 international working conference on Mining software repositories; New York, NY, USA: ACM, 2008 (= MSR '08) hURL: <http://doi.acm.org/10.1145/1370750.1370763i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-024-1, pp. 47-50
- Godfrey, Michael a. Qiang Tu: Growth, Evolution, and Structural Change in Open Source Software; in: Proceedings of the 4th International Workshop on Principles of Software Evolution; New York, NY, USA: ACM, 2001 (= IWPSE '01) hURL: <http://doi.acm.org/10.1145/602461.602482i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 1-58113-508-4, pp. 103-106
- Goeminne, Mathieu a. Tom Mens: A Framework for Analysing and Visualising Open Source Software Ecosystems; in: Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE); New

York, NY, USA: ACM, 2010 (= IWPSE-EVOL '10) hURL: <http://doi.acm.org/10.1145/1862372.1862384> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0128-2, pp. 42-47

- Goldmann, R. a. R.P. Gabriel: Innovation Happens Elsewhere. Open Source as Business Strategy; San Francisco: Elsevier, 2005
- Gomulkiewicz, Robert W.: De-Bugging Open Source Software Licensing; in: University of Pittsburgh Law Review, 64 (2002), pp. 75-99, BibWeb/PDF
- Gonzalez-Barahona, Jesus M., Joaquín Seoane Pascual, a. Gregorio Robles: Introduction to Free Software; coordinated by Jordi Mas Hernandez and David Megías Jimenez; Oberta de Catalunya: Free Technology Academy, 2009 hURL: <http://www.ftacademy.org/materials/fsm/1#1> - reference download: 2012-01-20, FreeWeb/PDF
- Goode, Sigi : Something for nothing: management rejection of open source software in Australia's top rms; in: Information & Management, 669-681 (2005), p. 42, BibWeb/PDF
- Gordon, Thomas F.: Analyzing Open Source License Compatibility Issues with Carneades; in: Proceedings of the 13th International Conference on Artificial Intelligence and Law; New York, NY, USA: ACM, 2011 (= ICAIL '11) hURL: <http://doi.acm.org/10.1145/2018358.2018364> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0755-0, pp. 51-55
- Grassmuck, Volker : Open Source - Betriebssysteme für eine freiheitliche Gesellschaft; hURL: <http://www.waste.informatik.hu-berlin.de/Grassmuck/Texts/OSS-Tutting-5-00.html>
- Grassmuck, Volker : Freie Software. Geschichte, Dynamiken und gesellschaftliche Bezüge; Berlin, 2000
- Grassmuck, Volker : Lizenzmodelle; 2000 hURL: <http://www.mikro.org/Events/OS/text/lizenzen.html>
- Grassmuck, Volker: Freie Software. Zwischen Privat- und Gemeineigentum; Themen und Materialien; Bonn: Bundeszentrale für politische Bildung, 2002, Print, ISBN 3-89331-432-6
- Green, Collin et al.: Leveraging Open-Source Software in the Design and Development Process; in: Proceedings of the 27th International Conference Extended Abstracts on Human Factors

in Computing Systems; New York, NY, USA: ACM, 2009 (= CHI EA '09) hURL: <http://doi.acm.org/10.1145/1520340.1520433>i - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-247-4, pp. 3061-3074

- Green, Eric Lee: Economics of Open Source Software; 1998 hURL: <http://badtux.org/home/eric/editorial/economics.php>
- Green, Lisa a. Heather Meeker: Open Software Licenses: Part II; in: Intellectual Property Strategist 10 (1999)
- Greve, Georg C. F.: Geschichte und Philosophie des GNU Projekts; hURL: <http://www.gnu.org/philosophy/greeve-clown.html>
- Grodzinsky, F. S. a. M. C. Bottis: Private Use as Fair Use: Is it Fair? in: SIGCAS, 37 November (2007), No. 2, pp. 11-24 hURL: <http://doi.acm.org/10.1145/1327325.1327326>i - reference download: 2011-12-29, BibWeb/PDF
- Gräfe, Gernot: Open-Source-Software und Open-Source-Portale - Potentiale für die Softwareentwicklung in Hochschulen und den Ergebnistransfer in die Praxis; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 55-72, Print
- Grützmacher, Malte: Open-Source-Software - die GNU General Public License / Lizenzbestimmungen im Umfeld des neuen Schuld- und Urhebervertragsrechts; in: ITRB (ITRechtsberater), (2002), pp. 84
- Grützmacher, Malte: Open Source Software - BSD Copyright and Apache Software. Copyright statt Copyleft; in: ITRB, o.A. (2006), No. 5, pp. 1008-112, Copy
- Guibault, Lucie a. Ot van Daalen: Unravelling the Myth around Open Source Licenses. An Analysis from A Dutch and European Law Perspective; The Hague: T. M. C. Asser Press, 2006 (= IT & Law, [Vol./No.] 8), Print, ISBN 978-90-6704-214-7
- Gull, Daniel: Valuation of Discount Options in Software License Agreements; in: BISE, 4 (2011), pp. 221-230 hURL: <http://dx.doi.org/10.1007/s12599-011-0170-8>i - reference download: 2012-02-09, BibWeb/PDF
- Gurbani, Vijay K., Anita Garvert, a. James D. Herbsleb: Managing a Corporate Open Source Software Asset; in: Communications of the ACM, 53 February (2010), No. 2, pp. 155-159

hURL: <http://doi.acm.org/10.1145/1646353.1646392> - reference download: 2011-12-29,  
BibWeb/PDF

- Gutsche, Jörg: Ökonomische Analyse einer Software; Mannheim: Universität Mannheim, 2006, BibWeb/PDF
- Gutwin, Carl, Reagan Penner, a. Kevin Schneider: Group Awareness in Distributed Software Development; in: Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work; New York, NY, USA: ACM, 2004 (= CSCW '04) hURL: <http://doi.acm.org/10.1145/1031607.1031621> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-810-5, pp. 72-81
- Haase, H.: Die Patentierbarkeit von Computersoftware; Hamburg, 2003  
Haddad, I.: Adopting an Open Source Approach to Software Development, Distribution, and Licensing; in: Enterprise Open Source Magazine, (2007) hURL: <http://opensource.sys-con.com/read/318776.html>
- Haddad, Ibrahim: Open-Source Compliance; in: Linux Journal, 185 September (2009), p. 5:1 hURL: <http://dl.acm.org/citation.cfm?id=1610564.1610569> - reference download: 2011-12-29, BibWeb/HTML
- Hahn, Robert, editor : Government Policy toward Open Source Software; AEI-Brooking Joint Centre (Org.) 2002
- Hamerly, Jim, Tom Paquin, a. Susan Walton: Freeing the Source: The Story of Mozilla; In DiBona, Ockman, a. Stone: Open Sources, 1999, pp. 197-206
- Hammouda, Imed et al.: Open source legality patterns: architectural design decisions motivated by legal concerns; in: Proceedings of the 14th International Academic MindTrek Conference: Envisioning Future Media Environments; New York, NY, USA: ACM, 2010 (= MindTrek '10) hURL: <http://doi.acm.org/10.1145/1930488.1930533> - reference download: 2012-01-06, BibWeb/PDF, ISBN 978-1-4503-0011-7, pp. 207-214
- Hardaway, Donald E.: Sharing Research in the 21st Century: Borrowing a Page from Open Source Software; in: Communications of the ACM, 48 August (2005), No. 8, pp. 125-128 hURL: <http://doi.acm.org/10.1145/1076211.1076216> - reference download: 2011-12-29, BibWeb/PDF
- Hars, Alexander a. Shaosong Ou: Working for free? Motivations for participating in open

source projects; in: International Journal of Electronic Commerce, 6 (2002), No. 3, pp. 25-39

- Hauge, Oyvind et al.: An Empirical Study on Selection of Open Source Software - Preliminary results; In Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2009, pp. 42-47 hURL: <http://dx.doi.org/10.1109/FLOSS.2009.5071359>i - reference download: 2011-12-29, BibWeb/PDF
- Hauge, Oyvind a. Sven Ziemer: Providing Commercial Open Source Software: Lessons Learned; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 70-82, BibWeb/PDF
- Hawkins, R. E.: The Economics of Open Source Software in a Competitive Firm: Why Give It Away For Free? in: Netnomics, 6 (2004), pp. 103-117
- Heap, Nicholas: OSI-Referenzmodell ohne Geheimnis; translated by G & U, Flensburg; Hannover: Heise, 1994, Print, ISBN 3-88229-045-5
- Hecker, F.: Setting up Shop: The Business of Open Source Software; in: IEEE Software, Jan/Feb (1999), pp. 46-61
- Hean, Ira V.: Copyleft: Licensing Collaborative Works in the Digital Age; in: Stanford Law Review, 1997 (49), pp. 1487-1521 hURL: <http://www.jstor.org/stable/1229351>i - reference download: 2012-02-09, BibWeb/PDF
- Heise online: Deutsches Gericht bestätigt Wirksamkeit der GPL; 2004 hURL: <http://www.heise.de/newsticker/meldung/49377>i
- Heise online: Der Streit um Softwarepatente; 2007 hURL: <http://www.heise.de/ct/hintergrund/meldung/61230>i
- Hemel, Armijn et al.: Finding Software License Violations Through Binary Code Clone Detection; in: Proceedings of the 8th Working Conference on Mining Software Repositories; New York, NY, USA: ACM, 2011 (= MSR '11) hURL: <http://doi.acm.org/10.1145/1985441.1985453>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0574-7, pp. 63-72
- Henkel, Joachim: Software development in embedded Linux; In Uhr, Esswein, a. Schoop: Wirtschaftsinformatik 2003 / Band II, 2003, pp. 81-99

- Henkel, Joachim: The Jukebox Mode of Innovation; A Model of Commercial Open Source Development; 2004 hURL: <http://opensource.mit.edu/papers/henkel.pdf>
- Henkel, Joachim: Open source software from commercial rms; Tools, complements, and collective invention; in: Zeitschrift für Betriebswirtschaft, 4/2004 (2004), No. 4 Ergänzungsheft, pp. 1-23
- Henkel, Joachim: Patterns of Free Revealing; Balancing Code Sharing and Protection in Commercial Open Source Development; 2004 hURL: <http://opensource.mit.edu/papers/henkel2.pdf>
- Henkel, Joachim: Oene Innovationsprozesse. Die kommerzielle Entwicklung von Open-Source-Software; Wiesbaden: Deutscher Universitäts-Verlag, 2007 (= Gabler Edition Wissenschaft), BibWeb/PDF, ISBN 978-3-8350-0978-3
- Herraiz, Israel et al.: The Processes of Joining in Global Distributed Software Projects; in: Proceedings of the 2006 International Workshop on Global Software Development for the Practitioner; New York, NY, USA: ACM, 2006 (= GSD '06) hURL: <http://doi.acm.org/10.1145/1138506.1138513> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-404-9, pp. 27-33
- Herraiz, Israel, Gregorio Robles, a. Jesus M. Gonzalez-Barahona: Towards Predictor Models for large Libre Software Projects; in: Proceedings of the 2005 Workshop on Predictor Models in Software Engineering; New York, NY, USA: ACM, 2005 (= PROMISE '05) hURL: <http://doi.acm.org/10.1145/1082983.1083168> - reference download: 2011-12-29, BibWeb/PDF, ISBN -159593-125-2, pp. 1-6
- Herraiz, Israel , Gregorio Robles, a. Jesus M. Gonzalez-Barahona: Research Friendly Software Repositories; in: Proceedings of the Joint International and Annual ERCIM Workshops on Principles of Software Evolution (IWPSE) and Software Evolution (Evol) Workshops; New York, NY, USA: ACM, 2009 (= IWPSE-Evol '09) hURL: <http://doi.acm.org/10.1145/1595808.1595814> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-678-6, pp. 19-24
- Hertel, Guido, Sven Niedner, a. Stefanie Herrmann: Motivation of Software Developers in Open Source Projects: An Internet-based Survey of Contributors to the Linux Kernel; in: Research Policy, 32 (2003), pp. 1159-1177
- Hill, Benjamin Mako: Samir Chopra, Scott D. Dexter, Decoding Liberation: The Promise

of Free and Open Source Software; in: *Minds Mach.* 18 June (2008), pp. 297-299 hURL: <http://dl.acm.org/citation.cfm?id=1375424.1375428i>, ISSN 0924-6495

- Hislop, Gregory W. et al.: Using Open Source Software to Engage Students in Computer Science Education; in: Proceedings of the 40th ACM technical symposium on Computer science education; New York, NY, USA: ACM, 2009 (= SIGCSE '09) hURL: <http://doi.acm.org/10.1145/1508865.1508915i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-183-5, pp. 134-135
- Hissam, Scott A. et al., editors: Open Source Systems: Grounding Research; 7th IFIP WG 2.13 International Conference, OSS 2011; (= IFIP Advances in Information and Communication Technology, [Vol./No.] 365) Heidelberg, Dordrecht, London u. NewYork: Springer, 2011, BibWeb/PDF, ISBN 978-3-642-24418-6
- Hoeren, Thomas: Anmerkungen zum Urteil vom 19.5.2004 des LG München I zur Wirksamkeit einer GPL-Lizenz; in: *CR [Computer und Recht]*, (2004), pp. 776-778
- Hoeren, Thomas: Internetrecht; April 2011, Web/PDF? hURL: <http://www.uni-muenster.de/Jura.itm/hoeren/lehre/materialieni>
- Hofmann, Susanne, Sven Pfeier, a. Urs Walter: Open Source School. Neue Synergien zwischen Schule und Kiez in Gropiusstadt. Architektur als sozialer Katalysator; Berlin, 2010, BibWeb/PDF hURL: [http://opus.kobv.de/tuberlin/volltexte/2010/2841/pdf/9783798322738\\_content.pdfi](http://opus.kobv.de/tuberlin/volltexte/2010/2841/pdf/9783798322738_content.pdfi) - reference download: 2011-07-30
- Horne, Natasha: Open Source Software Licensing: Using Copyright Law to Encourage Free Use; in: *Georgia State University Law Review*, (2001), pp. 863-891
- Horns, A.: Der Patentschutz für software bezogene Erndungen im Verhätnis zur 'Open Source'-Software; in: *Zeitschrift für Rechtsinformatik*, (2000) hURL: <http://www.jurpc.de/aufsatz/2000223.html>
- Howland, John E.: Software Freedom, Open Software and the Undergraduate Computer Science Curriculum; in: *JCSC*, 15 March (2000), No. 3, pp. 293-301 hURL: <http://dl.acm.org/citation.cfm?id=1852563.1852604i> - reference download: 2011-12-29, BibWeb/PDF
- Howland, John E.: Managing Computer Science Laboratories Using Open Software; in: *JCSC*, 16 March (2001), No. 3, pp. 117-126 hURL: <http://dl.acm.org/citation.cfm?id=374685.374726i> - reference download: 2011-12-29, BibWeb/PDF

- Hubbard, Jordan: Open Source to the Core; in: Queue, 2 (2004), pp. 24-31 hURL: <http://doi.acm.org/10.1145/1005062.1005064i>, BibWeb/PDF  
ifross: FAQ; 2011, FreeWeb/HTML hURL: <http://www.ifross.org/>
- faq-haeufig-gestellte-frageni - reference download: 2011-09-05
- ifross: Ziele, Aufgaben, Geschichte; 2011, FreeWeb/HTML hURL: <http://www.ifross.org/>  
node/16i - reference download: 2011-09-05
- ifross: License Center; 2011 [n.y.], FreeWeb/HTML hURL: [http://www.ifross.org/ifross\\_html/lizenzcenter-en.html](http://www.ifross.org/ifross_html/lizenzcenter-en.html) - reference download: 2013-02-26
- ifross et al.: Die GPL kommentiert und erklärt; Beijing, Cambridge, Farnham [etc ..]: O'Reilly, 2005, Print, ISBN 3-89721-389-3
- llardi, Terry J.: Common OSS License Problems; n.l., 2010, FreeWeb/PDF hURL: [http://www2.aipla.org/html/spring/2010/papers/llardi\\_Paper.pdf](http://www2.aipla.org/html/spring/2010/papers/llardi_Paper.pdf) - reference download: 2014-12-16
- Imhorst, Christian: Die Anarchie der Hacker. Richard Stallman und die Freie-Software-Bewegung; Marburg: Tectum Verlag, 2004, Print, ISBN 3-8288-8769-4
- Izurieta, Clemente a. James Bieman: The Evolution of FreeBSD and Linux; in: Proceedings of the 2006 ACM/IEEE international Symposium on Empirical Software Engineering; New York, NY, USA: ACM, 2006 (= ISESE '06) hURL: <http://doi.acm.org/10.1145/1159733.1159765i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 1-59593-218-6, pp. 204-211
- Jackson, Darla W.: Thinking about Technology . . . Watson, Answer Me This: Will You Make Librarians Obsolete or Can I Use Free and Open Source Software and Cloud Computing to Ensure a Bright Future? in: Law Library Journal, 103 (2011), No. 3, pp. 497-504, BibWeb/PDF
- Jacobs, Stephen, Clif Kussmaul, a. Mihaela Sabin: Free and Open Source Software in Computing Education; in: Proceedings of the 2011 Conference on Information Technology Education; New York, NY, USA: ACM, 2011 (= SIGITE '11) hURL: <http://doi.acm.org/10.1145/2047594.2047606i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-1017-8, pp. 41-42

- Jaeger, Till : Copyright oder Copyleft; in: Computerwoche Spezial, 27 (2000), No. 4, p. 36  
hURL: [http://www.ifross.de/ifross\\_html/art6.html](http://www.ifross.de/ifross_html/art6.html)
- Jaeger, Till : GPL und Haftung: Ohne Verantwortung? in: Linux-Magazin, (2000), No. 5, pp. 134
- Jaeger, Till : Die GPL kommentiert und erklärt; hrsg; v. ifross; Köln, 2005
- Jaeger, Till a. Axel Metzger: Open Source Software und deutsches Urheberrecht; in: GRUR Int. (1999), pp. 839
- Jaeger, Till a. Axel Metzger: Open Source Software. Rechtliche Rahmenbedingungen der Freien Software; 1st edition. München: Verlag C.H. Beck, 2002, Print, ISBN 3406484026
- Jaeger, Till a. Axel Metzger: Open Content-Lizenzen nach deutschem Recht; in: MultiMedia und Recht, (2003), pp. 431
- Jaeger, Till a. Axel Metzger: Open Source Software. Rechtliche Rahmenbedingungen der Freien Software; 2nd edition. München: Verlag C.H. Beck, 2006, Print, ISBN 3406538037
- Jaeger, Till a. Axel Metzger: Die neue Version 3 der GNU general Public License; in: GRUR (Gerwerblicher Rechtsschutz und Urheberrecht), (2008), pp. 130-137
- Jaeger, Till a. Axel Metzger: Open Source Software. Rechtliche Rahmenbedingungen der Freien Software; 3rd edition. München: Verlag C.H. Beck, 2011, Print
- Jaeger, Till a. Carsten Schulz : Gutachten zu ausgewählten rechtlichen Aspekten der Open Source Software - im Rahmen des Projektes 'NOW - Nutzung des Open Source-Konzepts in Wirtschaft und Industrie'; Feb 2005
- Janamanchi, Balaji et al.: The State and Prole of Open Source Software Projects in health and medical informatics; in: International Journal of Medical Informatics, 78 (2009), No. 7, pp. 457-472 hURL: <http://www.sciencedirect.com/science/article/pii/S1386505609000318i> - reference download: 2012-02-09, BibWeb/PDF
- Jansson, Kurt, Patrick Danowski, a. Jakob Voss: Wikipedia: Kreative Anarchie für den freien Informations- und Wissensaustausch; In Drossou, Krempel, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 159-167, Print

- Jendroska, Dirk: Arbeitsgestaltung in der Softwareentwicklung: Ein empirischer Vergleich subjektiver Arbeitsmerkmale in proprietären und Open Source Softwareprojekten; Dissertation; Münster: Philosophischen Fakultät der Westfälischen Wilhelms Universität zu Münster, 2010, BibWeb/PDF
- Johnson, Justin P.: Open Source Software: Private Provision of a Public Good; in: Journal of Economics & Management Strategy, 11 (2002), No. 4, pp. 637-663
- Johnson, Michael K.: Licenses and Copyright; in: Linux Journal, 29 September (1996), p. 3:1 hURL: <http://dl.acm.org/citation.cfm?id=326350.326353>i - reference download: 2011-12-28, BibWeb/HTML
- Johnson-Eilola, Johndan: Open Source Basics: Definitions, Models, and Questions; in: Proceedings of the 20th Annual International Conference on Computer Documentation; New York, NY, USA: ACM, 2002 (= SIGDOC '02) hURL: <http://doi.acm.org/10.1145/584955.584967>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-543-2, pp. 79-83
- Johri, Aditya, Oded Nov, a. Raktim Mitra: "Cool" or "Monster"? Company Takeovers and Their Effect on Open Source Community Participation; in: Proceedings of the 2011 iConference; New York, NY, USA: ACM, 2011 (= iConference '11) hURL: <http://doi.acm.org/10.1145/1940761.1940806>i, ISBN 978-1-4503-0121-3, pp. 327-331
- Jones, Paul: Open (Source)ing the Doors for Contributor-run Digital Libraries; in: Communications of the ACM, 44 May (2001), No. 1, pp. 45-46 hURL: <http://doi.acm.org/10.1145/374308.374337>i - reference download: 2011-12-49, BibWeb/PDF
- Karel, Michael J.: Commercializing Open Source Software; in: Queue, 1 July/August (2003), pp. 46-55 hURL: <http://doi.acm.org/10.1145/945074.945125>i - reference download: 2011-12-28, BibWeb/PDF
- Karus, Siim a. Harald Gall : A Study of Language Usage Evolution in Open Source Software; in: Proceedings of the 8th Working Conference on Mining Software Repositories; New York, NY, USA: ACM, 2011 (= MSR '11) hURL: <http://doi.acm.org/10.1145/1985441.1985447>i - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-4503-0574-7, pp. 13-22
- Kelty, Christopher M: Free Software/Free Science; in: First Monday, 6 (2001), No. 12, p. o.A.
- Kelty, Christopher M.: Two Bits: The cultural Significance of Free Software; ???, 2008

- Kennedy, D. M.: A primer on open source licensing legal issues: copyright, copyleft and copyfuture; 2001 hURL: <http://www.denniskennedy.com/opensourcedmk.pdf>
- Kern, W. a. F. Rammig: Eine Einführung zum Open Source Konzept aus Sicht der wirtschaftlichen und rechtlichen Aspekte; Paderborn; in: C-LAB Report, 2 (2003), vielleicht Buch?
- Kersken, Sasche: Apache 2.2. Das umfassende Handbuch; 3rd, refreshed a. expanded edition; Bonn: Galileo Press, 2009, Print, ISBN 978-8362-1325-7
- Keuel, Warren: License Overload; in: Software Development, 14 (2006), No. 2, p. 56, Copy
- Kerler, Steen a. Paul Alpar: Customization of Open Source Software in Companies; 5th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2009 Skövde, Sweden, June 3-6, 2009; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 129-142, BibWeb/PDF
- Kienle, Holger M. et al.: Intellectual Property Aspects of Web Publishing; in: Proceedings of the 22nd annual international conference on Design of communication: The engineering of quality documentation; New York, NY, USA: ACM, 2004 (= SIGDOC '04) hURL: <http://doi.acm.org/10.1145/1026533.1026569> - reference download: 2011-12-28, BibWeb/PDF, ISBN 1-58113-809-1, pp. 136-144
- Kilamo, Terhi: The Community Game: Learning Open Source Development Through Participatory Exercise; in: Proceedings of the 14th International Academic MindTrek Conference: Envisioning Future Media Environments; New York, NY, USA: ACM, 2010 (= MindTrek '10) hURL: <http://doi.acm.org/10.1145/1930488.1930500> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0011-7, pp. 55-60
- Kirschner, Bryan: Building a Balanced Scorecard for Open Source Policy and Strategy: A Case Study of the Microsoft Experience; in: Proceedings of the 2nd International Conference on Theory and Practice of Electronic Governance; New York, NY, USA: ACM, 2008 (= ICEGOV '08) hURL: <http://doi.acm.org/10.1145/1509096.1509142> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-386-0, pp. 226-231
- Kitcat, Jason: Source Availability and E-voting: An Advocate Recants; in: Communications of the ACM, 47 October (2004), No. 10, pp. 65-67 hURL: <http://doi.acm.org/10.1145/1022594.1022625> - reference download: 2011-12-29, BibWeb/PDF
- Klemm, Martin: GPL Version 3.0; in: Innovation und internationale Rechtspraxis, o.A. (2009),

pp. 363-381, Copy

- Koch, Frank A.: Urheber- und kartellrechtliche Aspekte der Nutzung von Open-Source-Software (I); in: CR [Computer und Recht], (2000), pp. 273
- Koch, Frank A.: Urheber- und kartellrechtliche Aspekte der Nutzung von Open-Source-Software (II); in: CR [Computer und Recht], (2000), p. 333
- Koch, Frank A.: Probleme beim Wechsel zur neuen Version 3 der General Public License (Teil 1); in: ITRB (IT-Rechtsberater), (2007), pp. 261-263
- Koch, Frank A.: Probleme beim Wechsel zur neuen Version 3 der General Public License (Teil 2); in: ITRB (IT-Rechtsberater), (2007), pp. 285-288
- Koeglin, Olaf : Entfesseltes Wissen - Creative Commons und der Versuch, das GPL-Prinzip für jede Schöpfung anzuwenden; in: Linux-Magazin, (2003), No. 10, p. 70
- Koeglin, Olaf : Die Nutzung von Open Source Software unter neuen GPL Versionen nach der 'any later Version'-Klausel; in: CR [Computer und Recht], (2008), pp. 137-143
- Koeglin, Olaf a. Axel Metzger: Urheber- und Lizenzrecht im Bereich von Open-Source-Software; in Open Source Jahrbuch 2004; 2004, pp. 293
- Koenig, J.: Seven Open Source Business Strategies for mCompetitive Advantage; in: IT Manager's Journal, (2004) hURL: <http://management.itmanagersjournal.com/article?sid=04/05/10/2052216i>
- Koglin, Olaf : Opensourcerecht. Die urheber- und schuldrechtlichen Beziehungen zwischen Lizenzgeber und Lizenznehmer bei Open Source Software am Beispiel der General Public License (GPL); Frankfurt am Main: Peter Lang, 2007 (= Schriften zum Wirtschafts- und Medienrecht, Steuerrecht und Zivilprozeßrecht, [Vol./No.] 31), Print, ISBN 978-3-631-56308-3
- Kogut, B. a. A. Metiu: Open Source Software Development and Distributed Innovation; in: Oxford Review of Economic Policy, 17 (2001), pp. 248-264
- Koponen, Timo a. Virpi Hotti: Open Source Software Maintenance Process Framework; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 4:1-4:5 hURL: <http://doi.acm.org/10.1145/1082983.1083265i> - reference download: 2011-12-28,

## BibWeb/PDF

- Kreutzer, Till : Anmerkungen zum Urteil vom 19.5.2004 des LG München I zur Wirksamkeit einer GPL-Lizenz; in: MultiMedia und Recht, (2004), pp. 695-698
- Kreutzer, Till : Software und Spiele kopieren[:] Das Lizenzmodell entscheidet; In Djordjevic et al.: Urheberrecht im Alltag, 2008, pp. 29-33, Print
- Kreutzer, Till : Software veröffentlichen[:] Wem gehören die Rechte? In Djordjevic et al.: Urheberrecht im Alltag, 2008, pp. 163-167, Print
- Kreutzer, Till : Softwarelizenzen - Beispiele[:] Und welche Lizenz nehme ich jetzt? In Djordjevic et al.: Urheberrecht im Alltag, 2008, pp. 176-179, Print
- Krishnamurthy, Sandeep a. Arvind K. Tripathi: Monetary donations to an open source software platform; in: Research Policy, 38 (2009), pp. 404-414
- Krogstie, Birgit R.: Power Through Brokering: Open Source Community Participation in Software Engineering Student Projects; in: Proceedings of the 30th International Conference on Software Engineering; New York, NY, USA: ACM, 2008 (= ICSE '08) hURL: <http://doi.acm.org/10.1145/1368088.1368201> i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-079-1, pp. 791-800
- Kugler, Petra: Coordinating Innovation: Evidence from Open Source Development; Dissertation; St. Gallen: University of. St. Gallen, 2005, Print
- Kuhlen, Rainer: Open Innovation: Teil einer nachhaltigen Wissensökonomie; In Drossou, Krempl, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 12-23, Print
- Kuhn, Bradley M. et al.: Copyleft and the GNU General Public License: A Comprehensive Tutorial and Guide; n.l, 2014, FreeWeb/PDF hURL: <http://copyleft.org/guide/comprehensive-gpl-guide.pdf> i - reference download: 2014-12-15
- Kumar, Vineet, Brett R. Gordon, a. Kannan Srinivasan: Competitive Strategy for Open Source Software; in: MARKETING SCIENCE, 30 (2011), No. 6, pp. 1066-1078, BibWeb/PDF
- Käs, Simone: Rethinking industry practice. The emergence of openness in the embedded component industry; München: Pro BUSINESS, 2008, Print, ISBN 978-3-86805-256-5

- Lacy, Sarah: Open Source: Now It's an Ecosystem; 2005 hURL: [http://www.businessweek.com/technology/content/oct2005/tc2005103\\_0519\\_tc\\_218.htm](http://www.businessweek.com/technology/content/oct2005/tc2005103_0519_tc_218.htm)
- Lakhani, Karim R. a. Eric Hippel : How Open Source software works: "Free" user-to-user assistance; 2002, MIT Sloan School of Management Working Paper
- Lakhani, Karim R. a. Robert G. Wolf : Why Hackers Do What They Do: Understanding Motivation Eort in Free/Open Source Software Projects; (= MIT Sloan School of Management Working, Paper 4425-03), 2003
- Laroque, Chrstoph, Andre Döring, a. Thorsten Timm: 'Give or Let Buy': Kritische überlegungen eines Software-Ingeneurs zur Veröentlichung von Software als Open-Source-Projekte; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 155-166, Print
- Lavazza, Luigi et al.: Predicting OSS Trustworthiness on the Basis of Elementary Code Ssessment; in: Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement; New York, NY, USA: ACM, 2010 (= ESEM '10) hURL: <http://doi.acm.org/10.1145/1852786.1852834> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0039-1, pp. 36:1-36:4
- Lawrie, Tony a. Cristina Gacek: Issues of Dependability in Open Source Software Development; in: SIGSOFT Software Engineering Notes, 27 May (2002), No. 3, pp. 34-37 hURL: <http://doi.acm.org/10.1145/638574.638584>
- Lee, Samuel, Nina Moisa, a. Marco Weiss. An Economic Analysis: Open Source as a Signalling Device; Frankfurt a.M.: Goethe-University Frankfur/Main, 2003 (= Working Paper Series: Finance and Accounting, [Vol./No.] 102), BibWeb/PDF
- Lee, Samuel , Nina Moisa, a. Marco Weiss: Conditions for Open Source as a Signalling Device; Frankfurt a.M.: Goethe-University Frankfur/Main, 2004 (= Working Paper Series: Finance and Accounting), BibWeb/PDF
- Lejeune, Mathias: Rechtsprobleme bei der Lizenzierung von Open Source Software nach der GNU GPL; in: ITRB (IT-Rechtsberater), 1 (2003), pp. 10-12
- Lelli, Francesco a. Mehdi Jazayeri: Community Support for Software Development in Small Groups: the Initial Steps; in: Proceedings of the 2nd international workshop on Social software engineering and applications; New York, NY, USA: ACM, 2009 (= SoSEA '09)

hURL: <http://doi.acm.org/10.1145/1595836.1595840> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-682-3, pp. 15-22

- Lemley, Mark A. a. Ziv Shar: Who Chooses Open-Source Software? in: University of Chicago Law Review, 78 (2011), pp. 139-163, BibWeb/PDF
- Lenarcic, John a. Eric C. Mousset: The Open Source Singularity: A Postmodernist View; in: Selected papers from conference on Computers and philosophy - Volume 37; Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2003 (= CRPIT '03) hURL: <http://dl.acm.org/citation.cfm?id=1082145.1082157> - reference download: 2011-12-28, BibWeb/PDF, ISBN 1-920-68219-8, pp. 73-77
- Lerner, Josh a. Jean Tirole: The open source movement: Key research questions; in: European Economic Review, 45 (2001), pp. 819-826, BibWeb/PDF
- Lerner, Josh a. Jean Tirole: The Scope of Open Source Licensing; in: JLEO, 21 (2005), No. 1, pp. 20-56, BibWeb/PDF
- Lerner, Joshua a. Jean Tirole: Some simple economics of Open Source; in: Journal of Industrial Economics, 50 (2002), No. 2, pp. 197-234
- Levy, S.: Hackers; USA: Penguin, 2001
- Li, Yan, Chuan Hoo Tan, a. Hock Hai Teo: Firm-Specificity and Organizational Learningrelated Scale on Investment in Internal Human Capital for Open Source Software Adoption; in: Proceedings of the 2008 ACM SIGMIS CPR Conference on Computer Personnel Doctoral Consortium and Research; New York, NY, USA: ACM, 2008 (= SIGMIS CPR '08) hURL: <http://doi.acm.org/10.1145/1355238.1355244> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-069-2, pp. 22-29
- Li, Yan et al.: Motivating Open Source Software Developers: Influence of Transformational and Transactional Leadership; in: Proceedings of the 2006 ACM SIGMIS CPR Conference on Computer Personnel Research: Forty Four Years of Computer Personnel Research: Achievements, Challenges & the Future; New York, NY, USA: ACM, 2006 (= SIGMIS CPR '06) hURL: <http://doi.acm.org/10.1145/1125170.1125182> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-349-2, pp. 34-43
- Li, Yan et al.: Open Source Software Adoption: Motivations of Adopters and Amotivations of Non-adopters; in: SIGMIS Database, 42 May (2011), No. 2, pp. 76-94 hURL: <http://doi>.

- Li, Yung-Ming, Jhih-Hua Jhang-Li, a. Yen-Chun Liu: Optimal Strategies of IT Consulting Firms: The Impact of License Fee and Open Source; in: Proceedings of the 10th International Conference on Electronic Commerce; New York, NY, USA: ACM, 2008 (= ICEC '08) hURL: <http://doi.acm.org/10.1145/1409540.1409594i> - reference download: 2011-12-29, ISBN 978-1-60558-075-3, pp. 40:1-40:7
- Lin, Yi-Hsuan et al.: Open Source Licenses and the Creative Commons Framework: License Selection and Comparison; in: JISE, 22 (2006), pp. 1-17, BibWeb/PDF
- Lin, Yu-Wei a. Enrico Zini: Free/libre open source software implementation in schools: Evidence from the eld and implications for the future; in: Computers & Education, 50 (2008), No. 3, pp. 1092-1102 hURL: <http://www.sciencedirect.com/science/article/pii/S0360131506001722i>, BibWeb/PDF, ISSN 0360-1315
- Lindman, J., M. Rossi, a. A. Puustell : Matching Open Source Software Licenses with Corresponding Business Models; in: Software, IEEE, 28 july-aug. (2011), No. 4, pp. 31 -35, ISSN 0740-7459
- Lindman, Juho, Juha-Pekka Juutilainen, a. Matti Rossi: Beyond the Business Model: Incentives for Organizations to Publish Software Source Code; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 47-56, BibWeb/PDF
- Lovett, Jayne: Open Source - A Practical Solution; in: Proceedings of the 35th Annual ACM SIGUCCS Fall Conference; New York, NY, USA: ACM, 2007 (= SIGUCCS '07) hURL: <http://doi.acm.org/10.1145/1294046.1294099i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-59593-634-9, pp. 221-223
- Lundell, Björn, Brian Lings, a. Edvin Lindqvist: Open source in Swedish companies: where are we? in: Information Systems Journal, 20 (2010), p. 519-535, BibWeb/PDF
- Lutterbeck, Bernd, Matthias Baerwol , a. Robert A. Gehring, editors: Open Source Jahrbuch 2006. Zwischen Softwareentwicklung und Gesellschaftsmodell; Berlin: Lehmanns Media, 2006 hURL: [http://www.opensourcejahrbuch.de/download/jb2006/OpenSourceJahrbuch2006\\_online.pdfi](http://www.opensourcejahrbuch.de/download/jb2006/OpenSourceJahrbuch2006_online.pdfi) - reference download: 2011-10-17, Print & Free-Web/PDF, ISBN 3-86541-135-5
- Lutterbeck, Bernd, Matthias Baerwol , a. Robert A. Gehring, editors: Open Source

Jahrbuch 2007. Zwischen Softwareentwicklung und Gesellschaftsmodell; Berlin: Lehmanns Media, 2007 hURL: [http://www.opensourcejahrbuch.de/download/jb2007/OpenSourceJahrbuch2007\\_online.pdf](http://www.opensourcejahrbuch.de/download/jb2007/OpenSourceJahrbuch2007_online.pdf) - reference download: 2011-10-17, Print & Free-Web/PDF, ISBN 978-3-86541-191-4

- Lutterbeck, Bernd, Matthias Baerwol , a. Robert A. Gehring, editors: Open Source Jahrbuch 2008. Zwischen Softwareentwicklung und Gesellschaftsmodell; Berlin: Lehmanns Media, 2008 hURL: <http://www.opensourcejahrbuch.de/download/jb2008/osjb08.pdf> - reference download: 2011-10-17, Print & FreeWeb/PDF, ISBN 978-3-86541-271-3
- Lutterbeck, Bernd a. Robert A. Gehring Matthias Bärwol , editors: Open Source Jahrbuch; Berlin: Lehmanns Media, 2005 hURL: <http://...mitaufnehmen..i>
- Maaßen, C a. E. Schern: Softwarepatente; in: Das Wirtschaftsstudium, 33 (2004), No. 10, pp. 1026-1028
- Maaßen, C. a. E. Schern: Software-Lizenzierung; in: Das Wirtschaftsstudium, 34 (2005), No. 2, pp. 185-188
- Maaßen, Christian: Zur Bedeutung des Urheber- und Patanterechts in der quelloenen Softwareentwicklung; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 41-54, Print
- MacCormack, A., Rischack J , a. C. Y. Baldwin: Exploring the Structure of Complex Software Design: An Empirical Study of Open Source and Proprietary Code; in: Management Science, 52 (2006), pp. 1015-1030
- Mahler, Marcus: Open Source Software: The Success of an Alterntaive Intellectual Property Incentive Paradigm; in: Fordham Intellectual Property, Media & Entertainmeint Law Journal, 21 (2000), pp. 619-646
- Maldonado, Edgar: The Process of Introducing FLOSS in the Public Administration: The Case of Venezuela; in: JAIS, 11 (2010), No. 11, pp. 756-783, BibWeb/PDF
- Manabe, Yuki, Yasuhiro Hayase, a. Katuro Inoue: Evolutional Analysis of Licenses in FOSS; in: Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE); New York, NY, USA: ACM, 2010 (= IWPSE-EVOL '10) hURL: <http://doi.acm.org/10.1145/1862372.1862391> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-4503-0128-2, pp. 83-87

- Mancinelli, Fabio et al.: Managing the Complexity of Large Free and Open Source Package-Based Software Distributions; in: Proceedings of the 21st IEEE/ACM International Conference on Automated Software Engineering; Washington, DC, USA: IEEE Computer Society, 2006 hURL: <http://dl.acm.org/citation.cfm?id=1169218.1169319i>, ISBN 0-7695-2579-2, pp. 199-208
- Mann, Florian et al.: Open Access Publishing In Science; in: Communications of the ACM, 52 March (2009), pp. 135-139 hURL: <http://doi.acm.org/10.1145/1467247.1467279i>
- Mannaert, Herwig a. Kris Ven: The Use of Open Source Software Platforms by Independent Software Vendors: Issues and Opportunities; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 7:1-7:4 hURL: <http://doi.acm.org/10.1145/1082983.1083266i> - reference download: 2011-12.29, BibWeb/PDF
- Marly, Jochen: Praxishandbuch Softwarerecht; 5th edition. München: Beck, 2009
- Marmorstein, Robert: Open Source Contribution As An Eective Software Engineering Class Project; in: Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education; New York, NY, USA: ACM, 2011 (= ITiCSE '11) hURL: <http://doi.acm.org/10.1145/1999747.1999823i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0697-3, pp. 268-272
- Martins Melo, Felipe a. Pereira, Jr.: A Component-Based Open-Source Framework for General-Purpose Recommender Systems; in: Proceedings of the 14th international ACM Sigsoft symposium on Component based software engineering; New York, NY, USA: ACM, 2011 (= CBSE '11) hURL: <http://doi.acm.org/10.1145/2000229.2000239i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-4503-0723-9, pp. 67-72
- McAllister, Neil: Licence to Prot [. Hybrid Open Source Licensing]; in: New Architect and Web Techniques ([www.newarchitectmag.com](http://www.newarchitectmag.com)), 8 (2003), p. np., Copy
- McGowan, D.: Legal Implications of Open Source Software; in: University Illinois Law Review, (2001), pp. 241-304
- McGowan, David: The Tory Anarchism of F/OSS Licensing; in: University of Chicago Law Review, 78 (2011), pp. 207-223, BibWeb/PDF
- McInerney, Paul-Brian: Technology Movements and the Politics of Free/Open Source Software;

in: Science, Technology & Human Values, 34 (2009), No. 2, pp. 206-233 hURL: <http://sth.sagepub.com/content/34/2/206.abstract>, BibWeb/PDF

- Megias, David et al.: Free Technology Academy: a European initiative for distance education about Free Software and Open Standards; in: Proceedings of the 14th annual ACM SIGCSE conference on Innovation and technology in computer science education; New York, NY, USA: ACM, 2009 (= ITiCSE '09) hURL: <http://doi.acm.org/10.1145/1562877.1562904i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-381-5, pp. 70-74
- Meretz, Stefan: Linux & Co : freie Software - Ideen für eine andere Gesellschaft; Neu-Ulm: ???, 2000
- Metzger, Axel: Frei ab 18 Jahre; in: Linux-Magazin, (2000), No. 11, pp. 52
- Metzger, Axel: Anmerkungen zum Urteil vom 19.5.2004 des LG München I zur Wirksamkeit einer GPL-Lizenz; in: CR [Computer und Recht], (2004), pp. 778-780
- Michaelson, Jay: There's no such thing as a Free (Software) Lunch; in: Queue, 2 May (2004), pp. 40-47 hURL: <http://doi.acm.org/10.1145/1005062.1005066i> - reference download: 2011-12-29, BibWeb/PDF
- Microsoft: Einige Fragen zur GNU General Public License (GPL), die sich jedes Unternehmen stellen sollte; in: ??? (2001)
- Mitre: Use of Free and Open Source Software (FOSS) in the U.S. Department of Defense; hURL: <http://www.egovos.org/pdf/dodfoss.pdf>
- MLA: MLA Handbook for Writers of Research Papers; 7th edition. New York: The Modern Language Association of America, 2009, Print, ISBN 978-1-60329-024-1
- Mockus, Audris, Roy T. Fielding, a. James Herbsleb: A Case Study of Open Source Software Development: the Apache Server; in: Proceedings of the 22nd international conference on Software engineering; New York, NY, USA: ACM, 2000 (= ICSE '00) hURL: <http://doi.acm.org/10.1145/337180.337209i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-206-9, pp. 263-272
- Mockus, Audris, Roy T. Fielding, a. James D. Herbsleb: Two Case Studies of Open Source Software Development: Apache and Mozilla; in: Transactions on Software Engineering Methodology, 11 July (2002), No. 3, pp. 309-346 hURL: <http://doi.acm.org/10.1145/>

- Moglen, Even a. Mishi Choudhary: Software Freedom Law Center Guide to GPL Compliance, 2nd Edition; 2014, FreeWeb/HTML hURL: [https://www.softwarefreedom.org/resources/2014/SFLC-Guide\\_to\\_GPL\\_Compliance\\_2d\\_ed.html](https://www.softwarefreedom.org/resources/2014/SFLC-Guide_to_GPL_Compliance_2d_ed.html) - reference download: 2014-12-15
- Moglen, Peter: Anarchism triumphant: Free Software and the Death of Copyright; in: First Monday, 48 (1999), p. o.A.
- Monden, A. et al.: Guilty or Not Guilty: Using Clone Metrics to Determine Open Source Licensing Violations; in: Software, IEEE, 28 march-april (2011), No. 2, pp. 42 -47, ISSN 0740-7459
- Montante, Robert: A Survey of Portable Software; in: JCSC, 24 January (2009), No. 3, pp. 19-24 hURL: <http://dl.acm.org/citation.cfm?id=1409873.1409879> i - reference download: 2011-12-29, BibWeb/PDF
- Moody, Glyn: Die Software-Rebellen. Die Erfolgsstory von Linus Torvalds und Linux; transl. from the American [edition, 2000] by Annemarie Pumpering; Landsberg am Lech: verlag moderne industrie, 2001, Print, ISBN 3-478-38730-2
- Moody, Glyn: Rebel Code: Linux And The Open Source Revolution; [New York]: Basic Books, 2002, Print, ISBN 978-0738206707
- Moody, Glyn: Interview with Eric Raymond; in: Linux Journal, 165 January (2008), p. 5:1 hURL: <http://dl.acm.org/citation.cfm?id=1344189.1344194> i - reference download: 2011-12-29, BibWeb/HTML
- Morasca, Sandro, Davide Taibi , a. Davide Tosi: Towards Certifying the Testing Process of Open-Source Software: New Challenges or Old Methodologies? In Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2009, pp. 25-30 hURL: <http://dx.doi.org/10.1109/FLOSS.2009.5071356> i - reference download: 2011-12-29, BibWeb/PDF
- Morasca, Sandro, Davide Taibi , a. Davide Tosi: Towards certifying the testing process of Open-Source Software: New challenges or old methodologies? in: Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development; Washington, DC, USA: IEEE Computer Society, 2009 (= FLOSS '09) hURL:

<http://dx.doi.org/10.1109/FLOSS.2009.5071356i>, ISBN 978-1-4244-3720-7, pp. 25-30

- Morelli, Ralph a. Trishan de Lanerolle: Foss 101: Engaging Introductory Students in the Open Source Movement; in: Proceedings of the 40th ACM technical symposium on Computer science education; New York, NY, USA: ACM, 2009 (= SIGCSE '09) hURL: <http://doi.acm.org/10.1145/1508865.1508977i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-183-5, pp. 311-315
- Morgan, Lorraine a. Patrick Finnegan: Open Innovation in Secondary Software Firms: An Exploration of Managers' Perceptions of open Source Software; in: SIGMIS Database, 41 February (2010), No. 1, pp. 76-95 hURL: <http://doi.acm.org/10.1145/1719051.1719056i>, BibWeb/PDF
- Mozilla Foundation: Mozilla Public License 2.0 (MPL-2.0); 2012, FreeWeb/HTML hURL: <http://www.mozilla.org/MPL/2.0/i> - reference download: 2013-03-05
- Mozilla Foundation: About MPL 2.0: Revision Process and Changes FAQ; 2013 [n.y.], FreeWeb/HTML hURL: <http://www.mozilla.org/MPL/1.1/i> - reference download: 2013-03-05
- Mozilla Foundation: Mozilla Public License Version 1.1; 2013 [n.y.], FreeWeb/HTML hURL: <http://www.mozilla.org/MPL/1.1/i> - reference download: 2013-03-05
- Mtsweni, Jabu a. Elmarie Biermann: An investigation into the implementation of open source software within the SA government: an emerging expansion model; in: Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology; New York, NY, USA: ACM, 2008 (= SAICSIT '08) hURL: <http://doi.acm.org/10.1145/1456659.1456677i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-286-3, pp. 148-158
- Müller Molina, Arnoldo Jose a. Takeshi Shinohara: On Approximate Matching of Programs for Protecting Libre Software; in: Proceedings of the 2006 Conference of the Center for Advanced Studies on Collaborative research; New York, NY, USA: ACM, 2006 (= CASCON '06) hURL: <http://doi.acm.org/10.1145/1188966.1188994i> - reference download: 2011-12-29, BibWeb/PDF, pp. 1-14
- Mundhenke, Jens: Wettbewerbswirkungen von Open-Source-Software und oenen Standards auf Softwaremärkten; Berlin, Heidelberg, and New York: Springer, 2007 (= Kiel Studies,

- Munga, Neeshal, Thomas Fogwill , a. Quentin Williams: The Adoption of Open Source Software in Business Models: A Red Hat and IBM Case Study; in: Proceedings of the 2009 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists; New York, NY, USA: ACM, 2009 (= SAICSIT '09) hURL: <http://doi.acm.org/10.1145/1632149.1632165i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-60558-643-4, pp. 112-121
- Mustaqim, Moyen Mohammad: A Systems Thinking Model for Open Source Software Development in Social Media; in: Proceedings of the International Workshop on Modeling Social Media; New York, NY, USA: ACM, 2010 (= MSM '10) hURL: <http://doi.acm.org/10.1145/1835980.1835987i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0229-6, pp. 7:1-7:2
- Mustonen, Mikko: Copyleft - the economics of Linux and other open source software; in: Information Economics and Policy, 15 (2003), No. 1, pp. 99-121 hURL: <http://www.sciencedirect.com/science/article/pii/S0167624502000902i> - reference download: 2012-02-09, BibWeb/PDF & Copy
- Mustonen, Mikko: Essays on the Economics of Information and Communication Technologies: Copyleft, Networks and Compatibility; Ph.D thesis, University of Helsinki, Department of Economics, Faculty of Social Sciences, 2003
- Mustonen, Mikko: Why do rms support the development of substitute copyleft programs? Volume 15 of Mustonen: Copyleft - the economics of Linux and other open source software, 2003hURL: <http://www.sciencedirect.com/science/article/pii/S0167624502000902i> - reference download: 2012-02-09, pp. ??-??, BibWeb/PDF & Copy
- Mustonen, Mikko: When Does a Firm Support Substitute Open Source Programming? in: Journal of Economics & Management Strategy, 14 (2005), No. 1, pp. 121-138
- Müller, Martin: Open Source - kurz & gut; Köln, 1999 hURL: [http://www.oreilly.de/german/freebooks/os\\_tb/toc.html](http://www.oreilly.de/german/freebooks/os_tb/toc.html)
- Müller-Seitz, Gordon a. Guido Reger: Is open source software living up to its promises? Insights for open innovation management from two open source software-inspired projects; in: R&D Management, 39 (2009), No. 4, pp. 372-381 hURL: <http://dx.doi.org/10.1111/j.1467-9310.2009.00565.xi> - reference download: 2012-02-09, BibWeb/PDF

- Nadah, Nadia, Melanie Dulong de Rosnay, a. Bruno Bachimont: Licensing Digital Content With A Generic Ontology: Escaping From The Jungle of Rights Expression Languages; in: Proceedings of the 11th International Conference on Artificial Intelligence and Law; New York, NY, USA: ACM, 2007 (= ICAIL '07) hURL: <http://doi.acm.org/10.1145/1276318>. 1276330i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-59593-680-6, pp. 65-69
- Nagy, Del, Areej M. Yassin, a. Anol Bhattacherjee: Organizational adoption of open source software: barriers and remedies; in: Communications of the ACM, 53 (2010), pp. 148-151 hURL: <http://doi.acm.org/10.1145/1666420.1666457i>, BibWeb/PDF
- Nakakoji, Kumiyo et al.: Evolution Patterns of Open-Source Software Systems and Communities; in: Proceedings of the International Workshop on Principles of Software Evolution; New York, NY, USA: ACM, 2002 (= IWPSE '02) hURL: <http://doi.acm.org/10.1145/512035>. 512055i - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-545-9, pp. 76-85
- Netcraft : August 2011 Web Server Survey; 2011, FreeWeb/Html hURL: <http://news.netcraft.com/archives/2011/08/05/august-2011-web-server-survey-3.html> - reference download: 2011-08-31
- Neumann, Peter G.: Inside Risks: Robust Open-Source Software; in: Communications of the ACM, 42 February (1999), No. 2, p. 128 hURL: <http://doi.acm.org/10.1145/293411>. 293491i, BibWeb/PDF
- Nilendu, P. a. T. R. Madanmohan: Competing on Open Source: Strategies and Practise; 2002 hURL: <http://opensource.mit.edu/papers/madanmohan.pdf>
- Noll, John a. Wei-Ming Liu: Requirements Elicitation in Open Source Software Development: A Case Study; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 35-40 hURL: <http://doi.acm.org/10.1145/1833272>. 1833279i - reference download: 2011-12-28, BibWeb/PDF
- Nordquist, Pete, Anna Petersen, a. Angelina Todorova: License Tracing in Free, Open, and Proprietary Software; in: JCSC, 19 December (2003), No. 2, pp. 101-112 hURL: <http://dl.acm.org/citation.cfm?id=948785.948802i> - reference download: 2011-12-28, BibWeb/PDF

- Nov, Oded a. George Kuk: Open source content contributors' response to free-riding: The effect of personality and context; in: Computers in Human Behavior, 24 (2008), pp. 2848-2861, BibWeb/PDF
- Oberhem, Carolina: Vertrags- und Haftungsfragen beim Vertrieb von Open Source Software; Dissertation; Hamburg: Verlag Dr. Kovac, 2008 (= Recht der Neuen Medien, [Vol./No.] 50), Print, ISBN 978-3-8300-4075-0
- Oezbek, Christopher, Lutz Prechelt, a. Florian Thiel: The Onion has Cancer: Some Social Network Analysis Visualizations of Open Source Project Communication; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 5-10 hURL: <http://doi.acm.org/10.1145/1833272.1833274>i - reference download: 2012-02-01, BibWeb/PDF
- O'Hara, Keith J. a. Jennifer S. Kay: Open source software and computer science education; in: J. Comput. Small Coll. 18 February (2003), pp. 1-7 hURL: <http://dl.acm.org/citation.cfm?id=771712.771716>i, ISSN 1937-4771
- Omsels, Hermann-Josef : Open Source und das deutsche Vertrags- und Urheberrecht; in: Christian Schertz a. Herman-Josef Omsels, editors: Festschrift für Paul W. Hertin zum 60. Geburtstag; 2000
- Open Source Development Labs: OSDL announces patent common project; 2005 hURL: [http://www.osdl.org/newsroom/press\\_releases/2005/2005\\_08\\_09\\_beaverton.html](http://www.osdl.org/newsroom/press_releases/2005/2005_08_09_beaverton.html)
- Open Source Initiative: GNU General Public License, version 2 (GPL-2.0). Version 2, June 1991; 1991 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/GPL-2.0>i - reference download: 2013-02-05
- Open Source Initiative: The GNU Lesser General Public License, version 2.1 (LGPL-2.1); 1999 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/LGPL-2.1>i - reference download: 2013-03-06
- Open Source Initiative: Common Development and Distribution License (CDDL-1.0); 2004 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/CDDL-1.0>i - reference download: 2013-04-19
- Open Source Initiative: Apache License, Version 2.0; 2004 [n.y. of the page itself],

FreeWeb/HTML hURL: <http://opensource.org/licenses/Apache-2.0i> - reference download:

2013-02-07

- Open Source Initiative: Eclipse Public License, Version 1.0; 2005 [n.y. of the page itself],

FreeWeb/HTML hURL: <http://opensource.org/licenses/EPL-1.0i> - reference download:

2013-02-20

- Open Source Initiative: European Union Public License, version 1.1 (EUPL-1.1; 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/EUPL-1.1i> - reference download: 2013-03-04

- Open Source Initiative: GNU Aero General Public License, Version 3 (AGPL-3.0); 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/AGPL-3.0i> - reference download: 2013-04-05

- Open Source Initiative: GNU General Public License, version 3 (GPL-3.0); 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/GPL-3.0i> - reference download: 2013-03-05

- Open Source Initiative: The GNU Lesser General Public License, version 3.0 (LGPL-3.0); 2007 [n.y. of the html page itself], FreeWeb/HTML hURL: <http://opensource.org/licenses/LGPL-3.0i> - reference download: 2013-03-06

- Open Source Initiative: The BSD 2-Clause License; 2012 [n.y.], FreeWeb/HTML hURL: <http://www.opensource.org/licenses/BSD-2-Clausei> - reference download: 2012-07-03

- Open Source Initiative: The BSD 3-Clause License; 2012 [n.y.], FreeWeb/HTML hURL: <http://www.opensource.org/licenses/BSD-3-Clausei> - reference download: 2012-07-04

- Open Source Initiative: The MIT License; 2012 [n.y.], FreeWeb/HTML hURL: <http://opensource.org/licenses/mit-license.php> - reference download: 2012-08-24

- Open Source Initiative: The Open Source Denition; 2012 [n.y.], FreeWeb hURL: <http://www.opensource.org/docs/osdi> - reference download: 2012-06-21

- Open Source Initiative: The Open Source Initiative; 2012 [n.y.], FreeWeb hURL: <http://www.opensource.org/about/i> - reference download: 2013-01-22

- Open Source Initiative: The Open Source Licenses, alphabetically sorted; 2012 [n.y.], FreeWeb

hURL: <http://opensource.org/licenses/alphabetical> - reference download: 2013-01-22

- Open Source Initiative: The [OSI] Licence Review Process; 2012 [n.y.], FreeWeb hURL: <http://www.opensource.org/approval> - reference download: 2013-01-22
- Open Source Initiative: OSI Mailing List. License-discuss. Draft of new OSI licenses landing page; 2012 [n.y.], FreeWeb/HTML hURL: <http://projects.opensource.org/pipermail/license-discuss/2012-April/000332.html> - reference download: 2013-01-29
- Open Source Initiative: Microsoft Public License (MS-PL); 2013 [n.y.], FreeWeb/HTML hURL: <http://opensource.org/licenses/MS-PL> - reference download: 2013-02-26
- Open Source Initiative: Mozilla Public License 2.0 (MPL-2.0); 2013 [n.y.], FreeWeb/HTML hURL: <http://opensource.org/licenses/MPL-2.0> - reference download: 2013-02-07
- Open Source Initiative: Open Source Licenses by Category; 2013 [n.y.], FreeWeb hURL: <http://opensource.org/licenses/category> - reference download: 2013-01-29
- Open Source Initiative: The PHP License 3.0 (PHP-3.0); 2013 [n.y.], FreeWeb/HTML hURL: <http://opensource.org/licenses/PHP-3.0> - reference download: 2013-02-27
- Open Source Initiative: The PostgreSQL Licence (PostgreSQL); 2013 [n.y.], FreeWeb/HTML hURL: <http://opensource.org/licenses/PostgreSQL> - reference download: 2013-02-27
- Oreg, Shaul a. Oded Nov: Exploring motivations for contributing to open source initiatives: The roles of contribution context and personal values; in: Computers in Human Behavior, 24 (2008), No. 5, pp. 2055-2073 hURL: <http://www.sciencedirect.com/science/article/pii/S0747563207001537> - reference download: 2012-02-01, BibWeb/PDF
- O'Reilly, Tim: Lessons from Open-Source Software Development; in: Communications of the ACM, 42 (1999), No. 4, pp. 32-37 hURL: <http://doi.acm.org/10.1145/299157.299164> - reference download: 2011-12-28, BibWeb/PDF
- O'Reilly, Tim, editor : Open Source: kurz und gut; 1999
- Orsila, Heikki et al.: Trust Issues in Open Source Software Development; in: Proceedings of the Warm Up Workshop for ACM/IEEE ICSE 2010; New York, NY, USA: ACM, 2009 (= WUP '09) hURL: <http://doi.acm.org/10.1145/1527033.1527037> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-565-9, pp. 9-12

- Osterloh, M., S. Rota, a. M. von Wartburg: Open Source - New Rules in Software Development; Zürich, 2001, working paper hURL: <http://www.iou.unizh.ch/orga/downloads/OpenSourceAoM.pdf>
- Osterloh, Margit a. Sandra Rota: - Just another case of collective invention? in: Research Policy, 36 (2007), No. 2, pp. 157-171 hURL: <http://www.sciencedirect.com/science/article/pii/S004873306001983i> - reference download: 2012-02-01, BibWeb/PDF
- Osterloh, Margit , Sandra Rota, a. Bernhard Kuster: Open Source Software Production: Climbing on the Shoulders of Giants; 2002 hURL: <http://opensource.mit.edu/papers/osterlohratakuster.pdf>
- Osterloh, Margit , Sandra Rota, a. Roger Lüthi: 'Collective Invention' als neues Innovationsmodell; In Drossou, Krempel, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 65-76, Print
- O'Sullivan, Maureen: EOF[:] Free Software Licenses; in: Linux Journal, 122 June (2004), pp. Article No. 11 hURL: <http://dl.acm.org/citation.cfm?id=993247.993258i> - reference download: 2011-12-28, BibWeb/HTML
- O'Mahony, Siobhan: Guarding the commons: how community managed software projects protect their work; in: RP, 32 (2003), No. 7, pp. 1179-1198 hURL: <http://www.sciencedirect.com/science/article/pii/S004873303000489i> - reference download: 2012-02-09, BibWeb/PDF
- Patterson, Chip: Copyright Misuse and Modified Copyleft: New Solutions to the Challenges of Internet; in: Michigan Law Review, 98 (2000), No. 5, pp. 1351-1383, BibWeb/PDF
- Pelizza, Annalisa: Openness as an Asset: A Classification System for Online Communities Based on Actor-Network Theory; in: Proceedings of the 6th International Symposium on Wikis and Open Collaboration; New York, NY, USA: ACM, 2010 (= WikiSym '10) hURL: <http://doi.acm.org/10.1145/1832772.1832784i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0056-8, pp. 8:1-8:10
- Perens, Bruce: The Open Source Definition; In DiBona, Ockman, a. Stone: Open Sources, 1999, pp. 171-188
- Perens, Bruce: Combining GPL and Proprietary Software; in: Datamation, 9 (2009), pp. wp. [3 pages] hURL: <http://www.datamation.com/osrc/article.php/3801396/>

Bruce-Perens-Combining-GPL-and-Proprietary-Software.htm - reference download:

2012-03-09, FreeWeb/HTML

- Perr, Jon, Melissa M. Appleyard, a. Patrick Sullivan: Open for business: emerging business models in open source software; in: INTERNATIONAL JOURNAL OF TECHNOLOGY MANAGEMENT, 52 (2010), pp. 432-456
- Peters, Stormy: Open Source Is Changing the Way Work Gets Done; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, p. 1, BibWeb/PDF
- Petreley, Nicholas: /var/opinion: The GPLv2 vs. GPLv3 Debate; in: Linux Journal, 153 January (2007), p. 17 hURL: <http://dl.acm.org/citation.cfm?id=1194955.1194972> - reference download: 2011-12-28, BibWeb/HTML
- Phillips, Douglas E.: The Software License Unveiled. How Legislation by License Controls Software Access; Oxford, New York, Auckland [etc. ...]: Oxford University Press, 2009, ISBN 978-0-19-534187-4
- Piller, Frank T.: User Innovation: der Kunde kann's besser; In Drossou, Kreml, a. Poltmann: Die wunderbare Wissensvermehrung, 2006, pp. 85-97, Print
- Piller, Harald: Von Open Source zu Open Innovation; in: Harvard Business Manager, 25 (2003), No. 12, p. 114
- Pisano, G.: Protecting from Innovation and the Intellectual Property Revolution; in: Research Policy, 35 (2006), No. 8, pp. 1122-1130
- Plaß, Gunda: Open Contents im deutschen Urheberrecht; in: GRUR, (2002), pp. 670
- Polanski, Arnold: Is the General Public Licence a Rational Choice? in: Journal of Industrial Economics, 55 (2007), pp. 691-714, BibWeb/PDF
- Prechelt, Lutz : Some Non-Usage Data for a Distributed Editor: the Saros Outreach; in: Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering; New York, NY, USA: ACM, 2011 (= CHASE '11) hURL: <http://doi.acm.org/10.1145/1984642.1984651> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-4503-0576-1, p. 48
- Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source

Software Research and Development; (= FLOSS '09) Washington, DC, USA: IEEE Computer Society, 2009 hURL: <http://dl.acm.org/citation.cfm?id=1572192> - reference download: 2012-01-25, BibWeb/PDF, ISBN 978-1-4244-3720-7

- Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; (= FLOSS '10) New York, NY, USA: ACM, 2010 hURL: <http://dl.acm.org/citation.cfm?id=1833272> - reference download: 2012-01-25, BibWeb/PDF, ISBN 978-1-60558-978-7
- Proceedings of the Fifth Workshop on Open Source Software Engineering; (= 5-WOSSE) New York, NY, USA: ACM, 2005 hURL: <http://doi.acm.org/10.1145/1082983.1083260> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-59593-127-9
- Qureshi, Israr a. Yulin Fang: Socialization in Open Source Software Projects: A Growth Mixture Modeling Approach; in: Organizational Research Methods, 14 (2011), No. 1, pp. 208-238, BibWeb/PDF
- Raq, Muhammad: LIS community's perceptions towards open source software adoption in libraries; in: International Information & Library Review (2009) 41, 137e145, 41 (2009), pp. 137-145, BibWeb/PDF
- Raja, Uzma a. Evelyn Barry: Investigating quality in large-scale Open Source Software; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 1-4 hURL: <http://doi.acm.org/10.1145/1082983.1083268> - reference download: 2012-02-01, BibWeb/PDF
- Raymond, Eric: A Brief History of Hackerdom, revised version; 2000 hURL: <http://www.catb.org/~esr/writings/hacker-history/hacker-history.html>
- Raymond, Eric S.: How To Become A Hacker; hURL: <http://www.catb.org/esr/faqs/hacker-howto.html>
- Raymond, Eric S.: The Cathedral and the Bazaar; in: First Monday, 3 March (1998), No. 3, p. o.A.
- Raymond, Eric S.: Homesteading the Noosphere: An Introductory Contradiction; in: First Monday, 3 (1999), No. 10, p. o.A.

- Raymond, Eric S.: The cathedral and the bazaar : musings on Linux and open source by an accidental revolutionary; Peking [...]: ???, 2001
- Reed, Matthew W. et al.: Developing and Learning Web Services with Open Source Software: An Experience Report; in: JCSC, 22 April (2007), No. 4, pp. 93-100 hURL: <http://dl.acm.org/citation.cfm?id=1229637.1229654> - reference download: 2011-12-29, BibWeb/PDF
- Reese, Björn a. Daniel Sternberg: Working without Copyleft; 2001 hURL: <http://www.oreillynet.com/lpt/a/1403>
- Reincke, Karsten: Classical Scholar Texts With Footnotes based on LaTeX, BibTeX, Koma, jurabib and mykeds-CSR; 2012, FreeWeb/Html hURL: <http://www.fodina.de/en/closedprojects/latex-addons/classical-scholar.html> - reference download: 2013-02-10
- Reincke, Karsten: (Geistes-) Wissenschaftliche Texte mit jurabib. Dienst am Leser, Dienst am Scholaren: über Anmerkungsapparate in Fußnoten - aber richtig. [n.l.], 2012 hURL: <http://download.fodina.de/fodinaClassicalScholarFoNoDe.pdf> - reference download: 2013-02-10, FreeWeb/PDF
- Reincke, Karsten, Greg Sharpe, a. contributors: Open Source License Compendium. How to Achieve Open Source License Compliance; 2015, FreeWeb/PDF hURL: <http://www.oslic.org/releases/oslic.pdf> - reference download: 2015-01-20
- Reitmayer, Gerhard a. Dieter Schmalstieg: An Open Software Architecture for Virtual Reality Interaction; in: Proceedings of the ACM Symposium on Virtual Reality Software and Technology; New York, NY, USA: ACM, 2001 (= VRST '01) hURL: <http://doi.acm.org/10.1145/505008.505018> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-427-4, pp. 47-54
- Renner, Thomas et al.: Open Source Software. Einsatzpotentiale und Wirtschaftlichkeit; eine Studien der Fraunhofer Gesellschaft; Stuttgart: Fraunhofer IRB Verlag, 2005, Print, ISBN 3-8167-7008-8
- Reynolds, Carl J a. Jeremy C Wyatt: Open Source, Open Standards, and Health Care Information Systems; in: JMIR, 13 (2011), No. 1, p. wp hURL: <http://www.jmir.org/2011/1/e24>, BibWeb/HTML
- Rigby, Peter C., Daniel M. German, a. Margaret-Anne Storey: Open Source Software Peer Review Practices: A Case Study of the Apache Server; in: Proceedings of the 30th International

Conference on Software Engineering; New York, NY, USA: ACM, 2008 (= ICSE '08)  
hURL: <http://doi.acm.org/10.1145/1368088.1368162> - reference download: 2012-02-01,  
BibWeb/PDF, ISBN 978-1-60558-079-1, pp. 541-550

- Rigby, Peter C. a. Margaret-Anne Storey: Understanding Broadcast Based Peer Review on Open Source Software Projects; in: Proceedings of the 33rd International Conference on Software Engineering; New York, NY, USA: ACM, 2011 (= ICSE '11) hURL: <http://doi.acm.org/10.1145/1985793.1985867> - reference download: 2011-12-29, BibWen/PDF, ISBN 978-1-4503-0445-0, pp. 541-550
- Rivlin, Gary: Linus Torvalds - Leader of the Free World; in: Wired Magazin, (2003), No. 11, pp. 152
- Robbins, Arnold: What's GNU? in: Linux Journal, 1 March (1994), p. 9:1 hURL: <http://dl.acm.org/citation.cfm?id=328204.328213> - reference download: 2011-12-28, BibWeb/HTML
- Robert W, Guomulkiewics: How Copyleft uses License Right to succeed in the OPen Source Software Revolution and the Implications fpr Article 2b; in: Houston Law Review, 36 (???), pp. 179
- Roberts, Keith A.: Generic Methodology for Open Source Software Development; in: SIGSOFT Software Engineering Notes, 30 March (2005), No. 2, pp. 1-5 hURL: <http://doi.acm.org/10.1145/1050849.1050863> - reference download: 2011-12-29, BibWeb/PDF
- Rose, Marshall T.: The Open Book, A Practical Perspective on OSI; Englewood Clis NJ: Prentice Hall, 1990, Print, ISBN 0-13-643016-3
- Rosen, Lawrence: Geek Law[:] A Question of Licenses; in: Linux Journal, 89 September (2001), p. 14 hURL: <http://dl.acm.org/citation.cfm?id=509824.509838> - reference download: 2011-12-28, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Copyright Questions; in: Linux Journal, 88 August (2001), p. 13 hURL: <http://dl.acm.org/citation.cfm?id=509800.509813> - reference download: 2011-12-28, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] License FUD; in: Linux Journal, 92 December (2001), p. 14 hURL: <http://dl.acm.org/citation.cfm?id=512620.512634> - reference download: 2011-12-29, BibWeb/HTML

- Rosen, Lawrence: Geek Law[:] Naming Open-Source Software; in: Linux Journal, 90 October (2001), p. 11 hURL: <http://dl.acm.org/citation.cfm?id=509852.509863>i - reference download: 2011-12-28, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Allocation of the Risks; in: Linux Jo, 101 September (2002), p. 17 hURL: <http://dl.acm.org/citation.cfm?id=566949.566966>i - reference download: 2011-12-29, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Bad Law; in: Linux Journal, 98 June (2002), p. 13 hURL: <http://dl.acm.org/citation.cfm?id=513489.513502>i - reference download: 2011-12-29, BibWeb/PDF
- Rosen, Lawrence: Geek Law[:] Dealing with Patents in Softwar Licences; in: Linux Journal, 93 January (2002), p. 14 hURL: <http://dl.acm.org/citation.cfm?id=512788.512802>i - reference download: 2011-12-29, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Dealing With Patents in Software Licenses, Part II; in: Linux Journal, 94 February (2002), p. 15 hURL: <http://dl.acm.org/citation.cfm?id=513039.513054>i - reference download: 2011-12-28, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Fair Use; in: Linux Journal, 100 August (2002), p. 18 hURL: <http://dl.acm.org/citation.cfm?id=563953.563971>i - reference download: 2011-12-28, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] License Defamation; in: Linux Journal, 99 July (2002), p. 15 hURL: <http://dl.acm.org/citation.cfm?id=513581.513596>i - reference download: 2011-12-29, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Unbiased License FUD; in: Linux Journal, 95 March (2002), p. 15 hURL: <http://dl.acm.org/citation.cfm?id=513085.513100>i - reference download: 2011-12-29, BibWeb/HTML
- Rosen, Lawrence: Geek Law[:] Why the Public Domain Isn't a License; in: Linux Journal, 102 October (2002), p. 12 hURL: <http://dl.acm.org/citation.cfm?id=571785.571797>i - reference download: 2011-12-29, BibWeb/HTML
- Rosen, Lawrence: IAAL[:] Derivative Works; in: Linux Journal, 105 January (2003), p. 13 hURL: <http://dl.acm.org/citation.cfm?id=603771.603784>i - reference download: 2011-12-28, BibWeb/HTML

- Rosen, Lawrence: Open Source Licensing. Software Freedom and Intellectual Property Law; Upper Saddle River, New Jersey: Prentice Hall PTr, 2005, ISBN 0-13-148787-6
- Rosen, Lawrence: OSL 3.0: A Better License for Open Source Software; in: CRi, 6 (2007), pp. 166-171, Copy
- Rosenberg, D. K.: Open source - The unauthorized white papers; Chicago, 2000
- Rossi, Christina a. Andrea Bonaccorsi : Intrinsic motivations and preit-oriented rms supplying Open Source products and services; in: First Monday, 10 May (2005), No. 5, p. o.a.
- Rossi, Cristina a. Andrea Bonaccorsi : Why prot-oriented companies enter the OS eld?: Intrinsic vs. extrinsic incentives; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 12:1-12:5 hURL: <http://doi.acm.org/10.1145/1082983.1083269>i - reference download: 2011-12-29, BibWeb/PDF
- Rossi, Maria Alessandra: Decoding the "Fre/Open Sour ce(F/OSS) Software Puzzle": a survey of theoretical and empirical contributions; 2004 hURL: <http://opensource.mit.edu/papers/rossi.pdf>i
- Run, M. a. C. Ebert: Using Open Source Software in Product Development: A primer; in: IEEE SOFTWARE, 21 (2004), No. 1, pp. 82-86
- Sabin, Mihaela: Free and Open Source Software Development of IT Systems; in: Proceedings of the 2011 Conference on Information Technology Education; New York, NY, USA: ACM, 2011 (= SIGITE '11) hURL: <http://doi.acm.org/10.1145/2047594.2047601>i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-1017-8, pp. 27-32
- Sakurai, Cledson Akio a. Moacyr Martucci Junior: An Open System Architecture for Operation Support System at Telecommunications Service Providers; in: Proceedings of the 1st International Symposium on Information and Communication Technologies; o.O.: Trinity College Dublin, 2003 (= (ISICT '03)) hURL: <http://dl.acm.org/citation.cfm?id=963600.963705>i - reference download: 2012-02-01, BibWeb/PDF, pp. 524-529
- Samoladas, Ioannis et al.: Open Source Software Development Should Strive for Even Greater Code Maintainability; in: Communications of the ACM, 47 October (2004), No. 10, pp. 83-87 hURL: <http://doi.acm.org/10.1145/1022594.1022598>i - reference download: 2011-12-29, BibWeb/PDF

- Samuelson, Pamela: IBM's Pragmatic Embrace of Open Source; in: Communications of the ACM, 49 (2006), No. 10, pp. 21-25 hURL: <http://doi.acm.org/10.1145/1164394.1164412>i - reference download: 2011-12-28, BibWeb/PDF
- Samuelson, Pamela: Legally Speaking[:] When is a "License" Really a Sale? in: Communications of the ACM, 52 March (2009), No. 3, pp. 27-29 hURL: <http://doi.acm.org/10.1145/1467247.1467258>i - reference download: 2011-12-29, BibWeb/PDF
- Sandred, J.: Managing open source projects; New York, 2001
- Santos Jr., Carlos Denner et al.: Intellectual Property Policy and Attractiveness: A Longitudinal Study of Free and Open Source Software Projects; in: Proceedings of the ACM 2011 conference on Computer supported cooperative work; New York, NY, USA: ACM, 2011 (= CSCW '11) hURL: <http://doi.acm.org/10.1145/1958824.1958950>i - reference download: 2011-12-28, BibWeb/PDF, ISBN 978-1-4503-0556-3, pp. 705-708
- Sauer, Robert M.: Why develop open-source software? The role of non-pecuniary benefits, monetary rewards, and open-source licence type; in: Oxford Review of Economic Policy, 23 (2007), No. 4, pp. 605-619, BibWeb/PDF
- Sauerburger, Heinz, editor : Open Source Software; dpunkt.verlag, 2004
- Savage, S. S.: Conquering Open Source Fears; in: Linux Executive Report, (2006) hURL: <http://www.ibm.com/linux/i>
- Scacchi, W.: Understanding the Requirements for Developing Open Source Software Systems; in: IEEE Proceedings Software, 149 (2002), pp. 24-39
- Scacchi, Walt: OpenEC/B: Electronic Commerce and Free/Open Source Software Development; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 8:1-8:5 hURL: <http://doi.acm.org/10.1145/1082983.1083270>i - reference download: 2011-12-29, BibWeb/PDF
- Scacchi, Walt: Free/Open Source Software Development: Recent Research Results and Emerging Opportunities; in: The 6th Joint Meeting on European software engineering conference and the ACM SIGSOFT symposium on the foundations of software engineering: companion papers; New York, NY, USA: ACM, 2007 (= ESEC-FSE companion '07) hURL: <http://doi.acm.org/10.1145/1295014.1295019>i - reference download: 2011-12-29, BibWeb/PDF,

ISBN 978-1-59593-812-1, pp. 459-468

- Scacchi, Walt: The Future of Research in Free/Open Source Software Development; in: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research; New York, NY, USA: ACM, 2010 (= FoSER '10) hURL: <http://doi.acm.org/10.1145/1882362>. 1882427i - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0427-6, pp. 315-320
- Schi, Aaron: The economics of open source software: A survey of the early literature; in: The Review of Network Economics, 1 March (2002), No. 1, pp. 66-74
- Schiner, Thomas: Open Source Software - Freie Software im deutschen Urheber- und Vertragsrecht; München, 2002
- Schlesinger, David: Working with Open Source: A Practical Guide; in: interactions, 14 November/December (2007), pp. 35-37 hURL: <http://doi.acm.org/10.1145/1300655.1300678i> - reference download: 2011-12-29, BibWeb/PDF
- Schmitz, L.: Linuxworld: Debatte um Pool für Open-Source-Patente; in: Computerwoche, (2005) hURL: <http://www.computerwoche.de/index.cfm?pageid=254&artid=79815i>
- Schneider, Jochen: Handbuch des EDV-Rechts; 4th edition. Köln: Dr. Otto Schmidt, 2009
- Schricker a. Ulrich Loewenheim, editors: Urheberrecht; Kommentar; 4th edition. München: Beck, 2010
- Schryen, Guido: Is open source security a myth? in: Communications of the ACM, 54 (2011), pp. 130-140 hURL: <http://doi.acm.org/10.1145/1941487.1941516i>, BibWeb/PDF
- Schryen, Guido a. Rouven Kadura: Open source vs. closed source software: towards measuring security; in: Sung Y. Shin a. Sascha Ossowski, editors: Proceedings of the 2009 ACM symposium on Applied Computing; New York, NY, USA: ACM, 2009 (= SAC '09) hURL: <http://doi.acm.org/10.1145/1529282.1529731i> - reference download: 2012-01-06, BibWeb/PDF, ISBN 978-1-60558-166-8, pp. 2016-2023
- Schulz, Carsten: VSI-Gutachten zu Open-Source-Software. Die scharfe Klinge des Gesetzes? in: Linux-Magazin, (2003), pp. 68
- Schulz, Carsten: Dezentrale Softwareentwicklungs- und Softwarevermarktungskonzepte.

Vertragsstrukturen in Open Source Modellen; Köln, 2005

- Schäfer, Fabian: Der virale Eekt. Entwicklungsrisiken im Umfeld von Open Source Software; Karlsruhe: Universitätsverlag Karlsruhe, 2007, BibWeb/PDF, ISBN 978-3-86644-141-5
- Searls, Doc: Linux for Suits: Linus Takes a Pass on the New GPL Draft; in: Linux Journal, 145 May (2006), p. 15 hURL: <http://dl.acm.org/citation.cfm?id=1134160.1134175> - reference download: 2011-12-28, BibWeb/HTML
- Searls, Doc: EOF[:] Why to Build on Foss in the First Place; in: Linux Journal, 165 January (2008), pp. Article No. 16 hURL: <http://dl.acm.org/citation.cfm?id=1344189.1344205> - reference download: 2011-12-29, BibWeb/HTML
- Searls, Doc: EOF[:] The Power of Definitions; in: Linux Journal, 177 January (2009), pp. Article No. 15 hURL: <http://dl.acm.org/citation.cfm?id=1502508.1502523> - reference download: 2011-12-28, BibWeb/HTML
- Sebold, Gerd: Oene Wissensökonomie. Analysen zur Wissenssoziologie der Free/Open Source Softwareentwicklung; Dissertation; Wiesbaden: VS Verlag für Sozialwissenschaften, 2008, Print a. BibWeb/PDF, ISBN 978-3-531-15705-4
- Seel, Bernd a. Miriam Kraft: Einführung in das Prinzip Open Source; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 9-19, Print
- Seemayer, Walter a. Jason Matusow: Das Microsoft-Shared-Source-Programm aus der Business-Perspektive; In Lutterbeck a. Bärwol : Open Source Jahrbuch, 2005, pp. 185-200 hURL: <http://...mitaufnehmen..i>
- Sen, Ravi, Chandrasekar Subramaniam, a. Matthew Nelson: Determinants of the Choice of Open Source Software License; in: J. Manage. Inf. Syst. 25 December (2008), pp. 207-240 hURL: <http://dl.acm.org/citation.cfm?id=1554453.1554460>, ISSN 0742-1222
- Sen, Ravi, Chandrasekar Subramaniam, a. Matthew L. Nelson: Open source software licenses: Strong-copyleft, non-copyleft, or somewhere in between? in: Decision Support Systems, 52 (2011), No. 1, pp. 199-206 hURL: <http://www.sciencedirect.com/science/article/pii/S0167923611001242> - reference download: 2012-02-01, BibWeb/PDF
- Sester, Peter: Open-Source-Software: Vertragsrecht, Haftungsrisiken und IPR-Fragen; in: CR

[Computer und Recht], (2000), pp. 797

- Sethanandha, Bhuricha Deen: Improving Open Source Software Patch Contribution Process: Methods and Tools; in: Proceedings of the 33rd International Conference on Software Engineering; New York, NY, USA: ACM, 2011 (= ICSE '11) hURL: <http://doi.acm.org/10.1145/1985793.1986018>i - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-4503-0445-0, pp. 1134-1135
- Shibuya, Bianca a. Tetsuo Tamai: Understanding the Process of Participating in Open Source Communities; In Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2009, pp. 1-6 hURL: <http://dx.doi.org/10.1109/FLOSS.2009.5071352>i - reference download: 2012-02-01, BibWeb/PDF
- Siepmann, Jürgen: Lizenz- und haftungsrechtliche Fragen bei der kommerziellen Nutzung Freier Software; in: JurPC Web-Dok, (1999), p. 163
- Siepmann, Jürgen: Freie Software - Rechtfreier Raum? Rechtssicherheit im Umgang mit Open Source Software; München, 2000
- Singh, Param Vir: The Small-World Effect: The Influence of Macro-Level Properties of Developer Collaboration Networks on Open-Source Project Success; in: Transactions on Software Engineering Methodology, 20 (2010), No. 2, pp. 6:1-6:27 hURL: <http://doi.acm.org/10.1145/1824760.1824763>i - reference download: 2011-12-29, BibWeb/PDF
- Siponen, Mikko: A Justification for Software Rights; in: SIGCAS, 36 September (2006), No. 3, pp. 11-20 hURL: <http://dl.acm.org/citation.cfm?id=1195716.1195718>i - reference download: 2011-12-29, BibWeb/PDF
- Sirkkala, Petri, Timo Aaltonen, a. Imed Hammouda: Opening Industrial Software: Planting an Onion; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 57-69, BibWeb/PDF
- Smith, Bradford L.: The Future of Software: Enabling the Marketplace to Decide; In Hahn: Government Policy toward Open Source Software, 2002
- Sojer, Manuel: Reusing Open Source Code. Value Creation and Value Appropriation. Perspectives on Knowledge Reuse; wuth a Foreword by Univ.-Prof. Dr. Joachim Henkel; Wiesbaden: Gabler, 2011 (= Gabler Research[:] Innovation und Entrepreneurship), BibWEB/PDF, ISBN

- Sojer, Manuel a. Joachim Henkel: License Risks from Ad Hoc Reuse of Code from the Internet; in: Communications of the ACM, 54 December (2011), No. 12, pp. 74-81 hURL: <http://doi.acm.org/10.1145/2043174.2043193> - reference download: 2011-12-28, BibWeb/PDF
- Soto, Martin a. Marcus Ciolkowski : The QualOSS Open Source Assessment Model Measuring the Performance of Open Source Communities; in: Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement; Washington, DC, USA: IEEE Computer Society, 2009 (= ESEM '09) hURL: <http://dx.doi.org/10.1109/ESEM.2009.5314237> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-4244-4842-5, pp. 498-501
- Sowe, Sulayman K., Ioannis Stamelos, a. Lefteris Angelis: Understanding knowledge sharing activities in free/open source software projects: An empirical study; in: Journal of Systems and Software, 81 (2008), No. 3, pp. 431-446 hURL: <http://www.sciencedirect.com/science/article/pii/S0164121207000842> - reference download: 2012-02-03, BibWeb/PDF
- Spielkamp, Mathias: Creative Commons[:] Andere Zeiten, andere Lizenzen; In Djordjevic et al.: Urheberrecht im Alltag, 2008, pp. 219-221, Print
- Spielkamp, Mathias: Lessigletters-Remix[:] Die Creative-Commons-Initiative; In Djordjevic et al.: Urheberrecht im Alltag, 2008, pp. 223-230, Print
- Spindler, Gerald: Rechtsfragen der Open Source Software. Gutachten im Auftrags des VSI; München, 2003
- Spindler, Gerald: Stellungnahme [zum Gutachten der VSI]; in: Linux-Magazin, (2003), No. 9, p. 70
- Spindler, Gerald: Ausgewählte urheberrechtliche Problem von Open Source Software unter der GPL; in: Alfread Bülesbacg a. Thomas Dreier , editors: Wem gehört die Information im 21. Jahrhundert; 2004
- Spindler, Gerald: Open Source Software auf dem gerichtlichen Prüfstand - Dingliche Qualifikation und Inhaltskontrolle; in: Kommunikation und Recht, (2004), pp. 528-524
- Spindler, Gerald; Spindler, Gerald, editor : Rechtsfragen bei Open Source Software; Köln: Verlag Dr. Otto Schmidt KG, 2004, Print, ISBN 3-504-56080-0

- Spindler, Gerald a. Andreas Wiebe: Open Source-Vertrieb - Rechteeinräumung und Nutzungsberechtigung; in: Computerrecht, (2003), pp. 873-879
- Splittergerber, Andrea; Schröder, Georg F., editor : Lizenzen und Open Source rechtlich einwandfrei nutzen. Eine klare Darstellung der Lizenzierung, Nutzungsrechtseinräumung und deren Auswirkung auf Vertragsgestaltung; Kissing: Weka Media, 2005, Print, ISBN 3-8245-1286-3
- St. Laurent, Andrew N.: Understanding open source and free software licensing: guide to navigating licensing issues in existing & new software; Beijing and Köln: O'Reilly, 2004
- Sta, CACM: True seeds of open source software; in: Commun. ACM, 52 January (2009), pp. 6-6 hURL: <http://doi.acm.org/10.1145/1435417.1435420i>, ISSN 0001-0782
- Stahl, Matthew T.: Open-source software: not quite endsville; in: Drug Discovery Today, 10 (2005), No. 3, pp. 219-222 hURL: <http://www.sciencedirect.com/science/article/pii/S1359644604033641i> - reference download: 2012-02-09, BibWeb/PDF
- Stallman, Richard: Viewpoint: Why We Must Fight UCITA; in: Communications of the ACM, 43 June (2000), No. 6, pp. 27-28 hURL: <http://doi.acm.org/10.1145/336460.336470i> - reference download: 2011-12-29, BibWeb/PDF
- Stallman, Richard: Can Freedom Withstand E-Books? in: Communications of the ACM, 44 March (2001), No. 3, p. 111 hURL: <http://doi.acm.org/10.1145/365181.365227i> - reference download: 2011-12-29, BibWeb/PDF
- Stallman, Richard: Viewpoint: Why "Open Source" Misses the Point of Free Software; in: Communications of the ACM, 52 June (2009), No. 6, pp. 31-33 hURL: <http://doi.acm.org/10.1145/1516046.1516058i> - reference download: 2011-12-29, BibRef/PDF
- Stallman, Richard M.: The Danger of Software Patents; 2001, FreeWeb/HTML hURL: <http://www.gnu.org/philosophy/stallman-mec-india.htmli> - reference download: 2013-02-18
- Stallman, Richard M.: Can You Trust Your Computer? [originally written in 2002]; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 115-117, Print
- Stallman, Richard M.: The Danger of Software Patents; transcript of a speech given at University of Cambridge, London on the 25th of March 2002; In Stallman: Free Software,

Free Society: Selected Essays, 2002, pp. 95-111, Print

- Stallman, Richard M.: Free Software Denition; originally written in 1996; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 41-43, Print
- Stallman, Richard M.; Gay, Joshua, editor : Free Software, Free Society: Selected Essays of Richard M. Stallman; [with an] Introduction by Lawrence Lessig; Boston, MA USA: GNU Press, 2002, Print, ISBN 1-882114-98-1
- Stallman, Richard M.: Free Software: Freedom and Cooperation; transcript of a speech given at New York University on 29 May 2001; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 155-186, Print
- Stallman, Richard M.: Free Software Needs Free Documentation; originally written in 2000; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 67-68, Print
- Stallman, Richard M.: The GNU Manifesto; originally written in 1984; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 31-39, Print
- Stallman, Richard M.: The GNU Project; originally published in 'Open Sources: Voices from the Open Source Revolution, O'Reilly, 1999'; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 15-30, Print
- Stallman, Richard M.: The Right to Read; originally written in 1997; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 73-77, Print
- Stallman, Richard M.: Selling Free Software; originally written in 1996; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 63-65, Print
- Stallman, Richard M.: What is Copyleft? originally written in 1996; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 89-90, Print
- Stallman, Richard M.: What's in a Name? originally written in 2000; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 51-53, Print
- Stallman, Richard M.: Why 'Free Software' is Better than 'Open Software'; originally written in 1998; In Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 55-60, Print
- Stallman, Richard M.: Why Software Should Not Have Owners; originally written in 1994; In

Stallman: Free Software, Free Society: Selected Essays, 2002, pp. 45-49, Print

- Stallman, Richard M.: Let's Limit the Effect of Software Patents, Since We Can't Eliminate Them; in: Wired, n.st. January (2012), p. wp hURL: <http://www.wired.com/opinion/2012/11/richard-stallman-software-patents/i> - reference download: 2013-02-18, FreeWeb/HTML, ISSN n.st.
- Stallman, Richard M.: Fighting Software Patents - Singly and Together; n.st. [2004], FreeWeb/HTML hURL: <http://www.gnu.org/philosophy/fighting-software-patents.html> - reference download: 2013-02-18
- Steinbring, Marc a. Thorsten Hampel : Connecting Babbling Bazaars - Der Open-Source-Gedanke im Wandel zum oenen Service; In Asche et al.: Open Source. Kommerzialisierungsmöglichkeiten und Chancen für die Zusammenarbeit von Hochschulen und Unternehmen, 2008, pp. 73-97, Print
- Stewart, Katherine J., Anthony P. Ammeter, a. Likoebe M. Maruping: Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects; in: Informations Systems Research, 17 (2006), No. 2, pp. 126-144, BibWeb/PDF
- Stewart, Katherine J., David P. Darcy, a. Sherae L. Daniel: Observations on Patterns of Development in Open Source Software Projects; In Proceedings of the Fifth Workshop on Open Source Software Engineering, 2005, pp. 1-5 hURL: <http://doi.acm.org/10.1145/1082983.1083272> - reference download: 2011-12-29, BibWeb/PDF
- Stol, Klaas-Jan a. Muhammad Ali Babar: Challenges in Using Open Source Software in Product Development: A Review of the Literature; [General Chairs: Justin Erenkrantz and Hyrum K. Wright]; In Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2010, pp. 17-22 hURL: <http://doi.acm.org/10.1145/1833272.1833276> - reference download: 2011-12-29, BibWeb/PDF
- Stol, Klaas-Jan et al.: The Use of Empirical Methods in Open Source Software Research: Facts, Trends and Future Directions; In Proceedings of the 2009 ICSE Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development, 2009, pp. 19-24 hURL: <http://dx.doi.org/10.1109/FLOSS.2009.5071355> - reference download: 2012-02-01, BibWeb/PDF

- Subramaniam, Chandrasekar, Ravi Sen, a. Matthew L. Nelson: Determinants of open source software project success: A longitudinal study; in: Decision Support System, 46 (2009), pp. 576-585, BibWeb/PDF
- Subramanyam, Ramanath a. Mu Xia: Free/Libre Open Source Software development in developing and developed countries: A conceptual framework with an exploratory study; in: Decision Support Systems, 46 (2008), No. 1, pp. 173-186 hURL: <http://www.sciencedirect.com/science/article/pii/S016792360800119Xi> - reference download: 2012-02-03, BibWeb/PDF
- Suchomski, Bernd: Proprietäres Patentrecht beim Einsatz von Open Source Software. Eine rechtliche Analyse aus unternehmerischer Sicht; Bonn: Tgramedia, 2011 (= Medien Internet und Recht, [Vol./No.] 3), Print, ISBN 978-3-941192-03-4
- Sujecki, Bartosz : Open Source Software im deutschen Vetrags- und Urheberrecht; in: Medien und Recht, (2005), pp. 40-48
- Sweet, David: Andamooka: Open Support for Open Content; in: Linux Journal, 82 February (2001), pp. Article No. 13 hURL: <http://dl.acm.org/citation.cfm?id=364716.364729i> - reference download: 2011-12-28, BibWeb/HTML
- Syme, Serena a. L. Jean Camp: The Governance of Code: Open Land vs. UCITA Land; in: SIGCAS, 32 (2002), No. 3, p. 2 hURL: <http://doi.acm.org/10.1145/644618.644623i> - reference download: 2011-12-29, BibWeb/HTML
- Taubert, Niels C.: Produktive Anarchie? Netzwerke freier Softwareentwicklung; Bielefeld: transcript, 2006 (= Science Studies), Print, ISBN 3-89942-418-2
- Terry, Michael, Matthew Kay, a. Ben Lafreniere: Perceptions and Practices of Usability in the Free/Open Source Software (FoSS) Community; in: Proceedings of the 28th International Conference on Human Factors in Computing Systems; New York, NY, USA: ACM, 2010 (= CHI '10) hURL: <http://doi.acm.org/10.1145/1753326.1753476i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-929-9, pp. 999-1008
- Terry, Michael et al.: ingimp: Introducing Instrumentation to an End-User Open Source Application; in: Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems; New York, NY, USA: ACM, 2008 (= CHI '08) hURL: <http://doi.acm.org/10.1145/1357054.1357152i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-011-1, pp. 607-616

- Teupen, Christian: 'Copyleft' im deutschen Urheberrecht; Implikationen von Open Source Software im Urhebergesetz; Berlin: Duncker & Humblot, 2007 (= Schriften zum Bürgerlichen Recht, [Vol./No.] 367), Print, ISBN 978-3-428-12325-4
- The Linux Foundation: SPDX License List; 2013, FreeWeb/HTML hURL: <http://spdx.org/licenses/i> - reference download: 2014-03-14
- Themelidis, Markos: Open Source : die Freiheitsvision der Hacker; Frankfurt a.M.: ???, 2004
- Theunissen, W. H. Morkel, Andrew Boake, a. Derrick G. Kourie: In Search of the Sweet Spot: Agile Open Collaborative Corporate Software Development; in: Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries; Republic of South Africa: South African Institute for Computer Scientists and Information Technologists, 2005 (= SAICSIT '05) hURL: <http://dl.acm.org/citation.cfm?id=1145675.1145705i> - reference download: 2011-12-28, BibWeb/PDF, ISBN 1-59593-258-5, pp. 268-277
- Torkar, Richard, Pau Minoves, a. Janina Garrigos: Adopting Free/Libre/Open Source Software. Practices, Techniques and Methods for Industrial Use; in: JAIS, 12 (2011), No. 1, pp. 88-122, BibWeb/PDF
- Torvald, Linus: Just for Fun: wie ... [Biographie]; ???, 2004
- Torvalds, Linus: Just for fun : wie ein Freak die Computerwelt revolutionierte; die Biographie des Linux-Ernders; München: ???, 2004
- Tsai, John: For Better or Worse: Introducing the GNU General Public License Version 3; in: Berkeley Technology Law Review, 23 (2008), pp. 547-581, Copy
- Tsunoda, Masateru et al.: Analyzing OSS Developers' Working Time Using Mailing Lists Archives; in: Proceedings of the 2006 InternationalWorkshop on Mining Software Repositories; New York, NY, USA: ACM, 2006 (= MSR '06) hURL: <http://doi.acm.org/10.1145/1137983.1138031i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 1-59593-397-2, pp. 181-182
- Turner, David: Anatomy of GPL Violations; in: Free Software Foundation Bulletin, (2002), No. 1, pp. 2-3

- Turner, David: The LGPL and Java; 2004, FreeWeb/HTML hURL: <http://www.gnu.org/licenses/lgpl-java.en.html> - reference download: 2015-02-09
- Tuunanen, Timo, Jussi Koskinen, a. Tommi Kärkkäinen: Automated software license analysis; in: Automated Software Engineering, 16 (2009), pp. 455-490, BibWeb/PDF
- Twidale, Michael: Silver Bullet or Fool's Gold: Supporting Usability in Open Source Software Development; in: Proceedings of the 27th International Conference on Software Engineering; New York, NY, USA: ACM, 2005 (= ICSE '05) hURL: <http://doi.acm.org/10.1145/1062455.1062468> - reference download: 2012-02-01, BibWeb/PDF, ISBN 1-58113-963-2, p. 35
- Uhr, Wolfgang, Werner Esswein, a. Eric Schoop, editors: Wirtschaftsinformatik 2003 / Band II. Medien - Märkte - Mobilität; Heidelberg: Physica-Verlag, 2003
- Välimäki, Mikko: Copyleft Licensing and EC Competition Law; in: E.C.L.R, 27 (2006), No. 3, pp. 130-136, Copy
- Välimäki, Mikko, Ville Oksanen, a. Juha Laine: An Empirical Look at the Problems of Open Source Adoption in Finnish Municipalities; in: Proceedings of the 7th International Conference on Electronic Commerce; New York, NY, USA: ACM, 2005 (= ICEC '05) hURL: <http://doi.acm.org/10.1145/1089551.1089643> - reference download: 2011-12-29, ISBN 1-59593-112-0, pp. 514-520
- Valkov, Svilen: Innovative Concept of Open Source Enterprise Resource Planning (ERP) System; in: Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing; New York, NY, USA: ACM, 2008 (= CompSysTech '08) hURL: <http://doi.acm.org/10.1145/1500879.1500893> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-954-9641-52-3, pp. 11.6.1-11.6.7
- Vamplew, Peter a. Julian Dermoudy: An Anti-Plagiarism Editor for Software Development Courses; in: Alison Young a. Denis Tolhurst, editors: Australasian Computing Education Conference; Newcastle (Australia), 2005 (= Australia. Conferences in Research and Practice in Information Technology, [Vol./No.] 42), BibWeb/PDF, pp. 83-90
- Van den Brande, Ywain, Shane Coughlan, a. Till Jaeger, editors: The International Free and Open Source Software Law Book; Munich (Germany): Open Source Press, 2011, Print, ISBN 978-3-941841-49-9

- van Wendel de Joode, R., J. A. de Brujin, a. M. J. G. van Eeten: Protecting the Virtual Commons. Self-Organizing Open Source and Free Software Communities and Innovative Intellectual Property Regimes; The Hague: T.M.C. Asser Press, 2003 (= Information Technology & Law Series, [Vol./No.] 3), Print, ISBN 90-6704-159-9
- Ven, Kris a. Jan Verelst : The Importance of External Support in the Adoption of Open Source Server Software; conference contribution; In Boldyre et al.: Open Source Ecosystems, 2009, pp. 116-128, BibWeb/PDF
- Vetter, Greg R.: 'Infectious' Open Source Software: Spreading Incentives or Promoting Resistance? in: Rutgers Law Journal, 36:53 (2005), pp. 53-162
- Viesel, Edward: Freiheit statt Freibier. Geschichte und Praxis der freien digitalen Welt - mit einer Einführung in Linux; Münster: Unrast-Verlag, 2006, Print, ISBN 3-897771-450-7
- von Hippel, Eric a. Georg von Krogh: Open source software and the private-collective innovation model: Issues for organization science; in: Organization Science, 14 (2002), No. 2, pp. 209-223
- von Hippel, Eric a. Georg von Krogh: The Promise of Research on Open Source Software; in: Management Science, 52 (2006), No. 7, pp. 975-983
- von Krogh, G. a. E. von Hippel : The Promise of Research on Open Source Software; in: Management Science, 2006 (52), pp. 975-983
- von Krogh, Georg a. Sebastian Spaeth: The open source software phenomenon: Characteristics that promote research; in: Journal of Strategic Information Systems, 16 (2007), p. 236-253, BibWeb/PDF
- Välimäki, Mikko: The Rise of Open Source Licensing; A Challenge to the Use of Intellectual Property in the Software Industry; PhD thesis hURL: <http://pub.turre.com>
- Välimäki, Mikko: Dual Licensing in Open Source Software Industry; 2003 hURL: <http://opensource.mit.edu/papers/valimaki.pdf>
- Wang, Yi, Defeng Guo, a. Huihui Shi: Measuring the Evolution of Open Source Software Systems with their Communities; in: SIGSOFT Software Engineering Notes, 32 November (2007), No. 6, pp. 1-7 hURL: <http://doi.acm.org/10.1145/1317471.1317479> – reference download: 2011-12-29, BibWeb/PDF

- Watson, Richard T. et al.: The Business of Open Source; in: Communications of the ACM, 51 (2008), No. 4, pp. 41-46 hURL: <http://doi.acm.org/10.1145/1330311.1330321>
- Weber, S.: The Success of Open Source; Cambridge MA: Harvard University Press, 2004
- Weiss, Aaron: The Politics of Free (Software); in: netWorker, 5 September (2001), pp. 26-31 hURL: <http://doi.acm.org/10.1145/383719.383727>
- Weiss, Michael: Economics of Collectives; in: Proceedings of the 15th International Software Product Line Conference; Volume 2, New York, NY, USA: ACM, 2011 hURL: <http://doi.acm.org/10.1145/2019136.2019181> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0789-5, pp. 39:1-39:8
- West, Joel : How open is open enough? Melding proprietary and open source platform strategies; in: Research Policy, 32 (2003), pp. 1259-1285
- Wheeler, David A.: Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers! 2002 hURL: [http://www.dwheeler.com/oss\\_fs\\_why.html](http://www.dwheeler.com/oss_fs_why.html)
- Wichmann, Thorsten: Linux- und Open-Source-Strategien; Berlin, Heidelberg and New York: Springer, 2005, BibWeb/PDF, ISBN 3-540-22810-1
- Widmer, Mike J.: Open Source Software - Urheberrechtliche Aspekte freier Software; Dissertation; Bern: Stämpfli Verlag, 2003, Print
- Wiebe, A.: Softwarepatente und Open Source; in: CR [Computer und Recht], 20 (2004), No. 12, pp. 881-888
- Wikipedia (de): Microsoft Public License; n.l., 2013 [n.y.], FreeWeb/HTML hURL: [http://de.wikipedia.org/wiki/Microsoft\\_Public\\_License](http://de.wikipedia.org/wiki/Microsoft_Public_License) - reference download: 2013-02-26
- Wikipedia (de): Microsoft Reciprocal License; n.l., 2013 [n.y.], FreeWeb/HTML hURL: <http://de.wikipedia.org/wiki/Ms-RL> - reference download: 2013-02-26
- Wikipedia (en): Free and open source software; n.l., 2011, FreeWeb/HTML (German Version unter <http://de.wikipedia.org/wiki/FLOSS>) hURL: [http://en.wikipedia.org/wiki/Free\\_and\\_open\\_source\\_software](http://en.wikipedia.org/wiki/Free_and_open_source_software) - reference download: 2011-09-08
- Wikipedia (en): MIT License; n.l., 2011, FreeWeb/HTML hURL: <http://en.wikipedia.org/>

wiki/MIT\_Licensei - reference download: 2011-09-20

- Wikipedia (en): Copyleft; n.l., 2013 [n.y.], FreeWeb/HTML hURL: <http://en.wikipedia.org/wiki/Copyleft> - reference download: 2013-02-02
- Wikipedia (en): Permissive free software licence; n.l., 2013 [n.y.], FreeWeb/HTML hURL: [http://en.wikipedia.org/wiki/Permissive\\_free\\_software\\_licence](http://en.wikipedia.org/wiki/Permissive_free_software_licence) - reference download: 2013-02-02
- Wikipedia (en): Shared source; n.l., 2013 [n.y.], FreeWeb/HTML hURL: [http://en.wikipedia.org/wiki/Shared\\_source](http://en.wikipedia.org/wiki/Shared_source) - reference download: 2013-02-26
- Williams, Sam: Free as in Freedom. Richard Stallman's Crusade for Free Software; Beijing [... etc.]: O'Reilly, 2002, Print, ISBN 0-596-00287-4
- Witzel, Michaela: AGB-Recht und Open Source Lizenzmodelle; in: ITRB (IT-Rechtsberater), (2003), pp. 175
- Wolf, M., K. Miller , a. F. Grodzinsky: On the meaning of free software; in: Ethics and Information Technology, 11 (2009), pp. 279-286 hURL: <http://dx.doi.org/10.1007/s10676-009-9207-9> - reference download: 2012-02-03, BibWeb/PDF
- Wolf, Marty J. et al.: Open Source Software: Intellectual Challenges to the Status Quo; in: Proceedings of the 33rd SIGCSE Technical Symposium on Computer Science Education; New York, NY, USA: ACM, 2002 (= SIGCSE '02) hURL: <http://doi.acm.org/10.1145/563340.563464> - reference download: 2011-12-29, BibWeb/PDF, ISBN 1-58113-473-8, pp. 317-318
- Wolf, Marty J., Keith W. Miller , a. Frances S. Grodzinsky: Free, Source-Code-Available, or Proprietary: An Ethically Charged, Context-Sensitive Choice; in: SIGCAS, 39 June (2009), No. 1, pp. 15-26 hURL: <http://doi.acm.org/10.1145/1565795.1565797>
- Wolfe, Alexander: Toolkit: GNU Tools: Still Relevant? in: Queue, 1 December / January (2003/2004), pp. 14-17 hURL: <http://doi.acm.org/10.1145/966789.966795> - reference download: 2011-12-29, BibWeb/PDF
- Wuermeling, Ulrich a. Thies Deike: Open Source Software: Eine juristische Risikoanalyse; in: CR [Computer und Recht], (2003), pp. 87

- Wynants, Marleen a. Jan Cornelius, editors: How Open is the Future? Economic, Social & Cultural Scenarios inspired by Free & Open-Source Software; Brüssel: VUB Brussels University Press, 2005
- Xing, Guangming: Teaching Software Engineering Using Open Source Software; in: Proceedings of the 48th Annual Southeast Regional Conference; New York, NY, USA: ACM, 2010 (= ACM SE '10) hURL: <http://doi.acm.org/10.1145/1900008.1900085i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0064-3, pp. 57:1-57:3
- Xu, Bo, Donald R. Jones, a. Bingjia Shao: Volunteers' involvement in online community based software development; in: Information & Management, 46 (2009), pp. 151-158, BibWeb/PDF
- Yamakami, Toshihiko: Foundation-based Mobile Platform Software Engineering: Implications to Convergence to Open Source Software; in: Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human; New York, NY, USA: ACM, 2009 (= ICIS '09) hURL: <http://doi.acm.org/10.1145/1655925.1655962i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-60558-710-3, pp. 206-211
- Yamakami, Toshihiko: OSS as a digital ecosystem: A Reference Model for Digital Ecosystem of OSS; in: Proceedings of the International Conference on Management of Emergent Digital EcoSystems; New York, NY, USA: ACM, 2010 (= MEDES '10) hURL: <http://doi.acm.org/10.1145/1936254.1936291i> - reference download: 2011-12-29, BibWeb/PDF, ISBN 978-1-4503-0047-6, pp. 207-208
- Yatani, Koji et al.: Understanding How and Why Open Source Contributors Use Diagrams in the Development of Ubuntu; in: Proceedings of the 27th International Conference on Human Factors in Computing Systems; New York, NY, USA: ACM, 2009 (= CHI '09) hURL: <http://doi.acm.org/10.1145/1518701.1518853i> - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-60558-246-7, pp. 995-1004
- Ye, Yunwen a. Kouichi Kishida: Toward An Understanding of the Motivation of Open Source Software Developers; in: Proceedings of the 25th International Conference on Software Engineering; Washington, DC, USA: IEEE Computer Society, 2003 (= ICSE '03) hURL: <http://dl.acm.org/citation.cfm?id=776816.776867i>, ISBN 0-7695-1877-X, pp. 419-429
- Yildirim, Nihan a. Hacer Ansal: Foresighting FLOSS (free/libre/open source software) from a developing country perspective: The case of Turkey; in: Technovation, 31 (2011), No. 12, pp. 666 - 678 hURL: <http://www.sciencedirect.com/science/article/pii/S0166497211001052i>,

- Yue, Kwok-Bun et al.: The Use of Free and Open Source Software in Real-World Capstone Projects; in: JCSC, 26 April (2011), No. 4, pp. 85-92 hURL: <http://dl.acm.org/citation.cfm?id=1953573.1953587>i - reference download: 2011-12-29, BibWeb/PDF
- Yue, Kwok-Bun et al.: Open Courseware and Computer Science Education; in: JCSC, 20 October (2004), No. 1, pp. 178-186 hURL: <http://dl.acm.org/citation.cfm?id=1040231>.1040255i
- Zacchiroli, Stefano: Debian: 18 years of free software, do-ocracy, and democracy; in: Proceedings of the 2011 Workshop on Open Source and Design of Communication; New York, NY, USA: ACM, 2011 (= OSDOC '11) hURL: <http://doi.acm.org/10.1145/2016716.2016740>i, ISBN 978-1-4503-0873-1, pp. 87-87
- Zhang, Wen, Ye Yang, a. Qing Wang: Network Analysis of OSS Evolution: An Empirical Study on ArgoUML Project; in: Proceedings of the 12th International Workshop on Principles of Software Evolution and the 7th annual ERCIM Workshop on Software Evolution; New York, NY, USA: ACM, 2011 (= IWPSE-EVOL '11) hURL: <http://doi.acm.org/10.1145/2024445.2024459>i - reference download: 2012-02-01, BibWeb/PDF, ISBN 978-1-4503-0848-9, pp. 71-80
- Zhou, Ying a. Joseph Davis: Open source software reliability model: an empirical approach; in: Proceedings of the 6th workshop on Open source software engineering; New York, NY, USA: ACM, 2005 (= 5-WOSSE) hURL: <http://doi.acm.org/10.1145/1082983.1083273>i, ISBN 1-59593-127-9, pp. 1-6
- Zittrain, Jonathan: Normative Principles for Evaluating Free and Proprietary Software; in: University of Chicago Law Review, 71 (2004), No. 1, pp. 265-287, BibWeb/PDF
- Zucker, William A.: Intellectual Property and Open Source: Copyright, Copyleft, and Other Issues for the User Community; in: Cutter IT Journal, 16 (2003), No. 5, pp. 27-34, Copy