

소프트웨어안전 분야에서 공개소프트웨어 활성화 방안

Activation Methods for Open Source Software
in the Software Safety Fields

권영환/송지환/진희승

2018.12.

이 보고서는 2018년도 과학기술정보통신부 정보통신진흥기
금을 지원받아 수행한 연구결과의 보고서로서 내용은 연구
자의 견해이며, 과학기술정보통신부의 공식입장과 다를 수
있습니다.

목 차

제1장 서 론	1
제1절 연구 필요성 및 목적	1
제2절 연구 방법	2
제2장 주요 개념	3
제1절 소프트웨어안전 개요	3
제2절 공개소프트웨어	8
제3장 해외 공개소프트웨어 활용 동향	17
제1절 소프트웨어 신기술 분야	17
제2절 주요 안전 산업 분야	38
제3절 요약 및 시사점	55
제4장 국내 소프트웨어안전 분야 공개소프트웨어 현황	59
제1절 설문 조사 수행 방법	59
제2절 설문 조사 결과	64
제3절 조사 결과 요약 및 시사점	97
제5장 소프트웨어안전 분야 공개소프트웨어 활성화 방안	99
제1절 공개소프트웨어 활용 방안	99

제2절 공개소프트웨어 활성화 방안	103
 제6장 결론	 109
제1절 연구 요약	109
제2절 한계점 및 향후 연구 방향	111
 Reference	 112

표 목 차

<표 2-1> OSI에서 정의한 오픈소스의 10가지 특징	11
<표 3-1> 빅데이터 플랫폼에서 가치 추출을 위한 단계들	28
<표 3-2> 하둡의 기능을 보완하는 빅데이터 관련 공개SW들	29
<표 4-1> 안전 관련 주요 산업	63
<표 4-2> 개발에서 활용되는 SW 용도	64
<표 4-3> 설문 조사 항목 요약	66

- 표의 일련번호는 <표 1-1>, <표 1-2>, . . . <표 2-1>과 같이 순서대로 부여하며, 앞의 번호는 장을 지칭하고 뒤의 번호는 표의 일련번호를 지칭. 세모괄호 사용
 - 단위는 표의 우측 상단에 괄호안에 표기함을 원칙으로 한다.
 - <예> (단위 : 천원, 명, %) - 순서는 수치의 유형에 따라 정한다.
 - 표에 딸린 주(註, note)와 자료(source)는 주, 자료의 순으로 적는다.
- 표를 제외한 **그림, 그래프, 사진** 등은 모두 [그림 1-1], [그림 1-2], . . . [그림 2-1]과 같이 순서대로 부여하며 앞의 번호는 장을, 뒤의 번호는 표의 일련번호를 지칭. 네모괄호 사용.
- 표 및 그림 작성시 해당 페이지의 여백을 고려하여 지나치게 작거나 크지 않게 적절히 조정하고, 표제목·그림제목 및 표번호·그림번호는 반드시 작성한다.

그 립 목 차

[그림 1-1] 안전 분야 공개SW 활성화 방안 연구 방법	2
[그림 2-1] 위험과 안전의 관계	4
[그림 2-2] 품질 프로세스와 안전개발 프로세스 비교	7
[그림 2-3] SW품질 절차를 포함한 SW안전 개발 절차	8
[그림 2-4] 차량용 인포테인먼트 시장에서의 리눅스 활용 예상	12
[그림 2-5] 2018년 가장 인기있는 데이터베이스	13
[그림 2-6] CRAN에 공개된 R 패키지의 수	15
[그림 2-7] 공개SW 라이선스 활용의 변화 (2010년과 2017년 비교)	16
[그림 3-1] 미국 심층학습 시장 전망(2014년부터 2025년까지)	18
[그림 3-2] 텐서플로우를 활용한 인공 신경망 모델링	19
[그림 3-3] Accord.Net을 활용한 데이터 분류	20
[그림 3-4] MS CNTK 구조	21
[그림 3-5] 아파치 Mahout의 알고리즘	22
[그림 3-6] Caffe와 Caffe2의 동작 차이점	23
[그림 3-7] MXNet의 확장 가능한 구조	24
[그림 3-8] 빅데이터의 특징	25
[그림 3-9] 전세계 빅데이터 시장 전망	26
[그림 3-10] 하둡 에코시스템	27
[그림 3-11] R의 활용 용도	30
[그림 3-12] IT 인프라 시장에서의 클라우드 비중 및 주요 클라우드 사업자들	31
[그림 3-13] 오픈스택의 구성 요소	32

[그림 3-14] 클라우드 파운드리 구조	34
[그림 3-15] 일반적인 가상화 기술과 도커의 차이	35
[그림 3-16] 쿠버네티스의 역할	36
[그림 3-17] 2017년 깃허브 동향에 포함된 공개SW 활용 분야	38
[그림 3-18] 바이두 아폴로 이정표	40
[그림 3-19] 아폴로 3.0 플랫폼	41
[그림 3-20] 아폴로 3.0에서의 가디언 관련 SW 구조	41
[그림 3-21] Apollo scape 예시 화면	42
[그림 3-22] Github를 통해 공개되어진 Apollo 2.5 버전 자율 테스트	43
[그림 3-23] 바이두 아폴로 프로젝트 생태계	44
[그림 3-24] Open Autonomous Safety의 주요 기능들	45
[그림 3-25] Autonomous Grade Linux의 주요 기능들	46
[그림 3-26] Automotive Grade Linux 의 엔터테인먼트 콘솔 예시	47
[그림 3-27] Wwise automotive의 작동 예시	48
[그림 3-28] GENIVI 플랫폼 비전	48
[그림 3-29] Wind River Helix Cockpit의 구조	49
[그림 3-30] openETCS의 목표	51
[그림 3-31] 프로젝트 참가 기업	52
[그림 3-32] Streamflow 화면	54
[그림 3-33] Laika BOSS의 동작	54
[그림 4-1] 현황 조사 수행 절차	59
[그림 4-2] IEC 61508 SW 개발 모델	62
[그림 4-3] 조사 대상 업체 규모	64

[그림 4-4] 분야별 조사 대상 업체 규모	64
[그림 4-5] 설문 대상 업체의 개발 SW 유형 및 사용 개발 언어(복수 응답)	65
[그림 4-6] 설문 조사 대상자 담당 업무	66
[그림 4-7] 설문 조사 대상자 근무 기간	66
[그림 4-8] 안전 분야 적용 표준 및 지침 준수 현황 (복수 응답)	67
[그림 4-9] ISO 26262의 SW 개발 프로세스	67
[그림 4-10] 자동차 분야 적용 표준 및 지침 (복수 응답)	68
[그림 4-11] 자동차 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)	69
[그림 4-12] IEC 62279의 철도 SW 개발 프로세스	70
[그림 4-13] 철도 분야 적용 표준 및 지침 (복수 응답)	70
[그림 4-14] 철도 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)	71
[그림 4-15] DO-178C의 항공 SW 개발 프로세스	72
[그림 4-16] 항공(국방) 분야 적용 표준 및 지침 (복수 응답)	73
[그림 4-17] 항공(국방) 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)	73
[그림 4-18] 무기체계 SW 획득 절차	74
[그림 4-19] 무기체계 SW 개발 및 관리 매뉴얼의 공개SW 관련 지침	74
[그림 4-20] SW 안전성(신뢰성) 확보를 위한 필요사항 (복수 응답)	75
[그림 4-21] SW 안전성(신뢰성) 확보를 위한 수행 활동 (복수 응답)	75
[그림 4-22] 분야별 공개SW 사내 지침 현황	76
[그림 4-23] 분야별 공개SW 사내 지침 내용 (복수 응답)	77
[그림 4-24] 분야별 공개SW 관련 전담 부서 활동 내용 (복수 응답)	77
[그림 4-25] 기업 규모별 공개SW 관련 사내 지침 및 전담 부서 현황	78
[그림 4-26] 안전 분야에서의 SW 활용 용도 (복수 응답)	79

[그림 4-27] 개발 시스템에서 공개SW 사용 현황	79
[그림 4-28] 공개SW 활용 용도 (복수 응답)	80
[그림 4-29] 공개SW 활용 목적 (복수 응답)	80
[그림 4-30] 공개SW 인지 경로	81
[그림 4-31] 공개SW 사용 이유 (복수 응답)	82
[그림 4-32] 공개SW 사용하지 않는 이유 (복수 응답)	82
[그림 4-33] 응답자들이 생각하는 공개SW 장점 (복수 응답)	83
[그림 4-34] 응답자들이 생각하는 공개SW 단점 (복수 응답)	84
[그림 4-35] 기업 및 설문 응답자의 공개SW 정의 및 개념 이해도	85
[그림 4-36] 공개SW 라이선스 이해도	85
[그림 4-37] 알고 있는 공개SW 라이선스 (복수 응답)	86
[그림 4-38] 라이선스 주요 정책 이해도 (복수 응답)	87
[그림 4-39] 사용 중인 공개SW의 라이선스 정책 이해도	88
[그림 4-40] 공개SW 교육(세미나) 경험	89
[그림 4-41] 공개SW에 대한 기업 인식	89
[그림 4-42] 안전 분야별 공개SW 기업 인식	90
[그림 4-43] 기업들의 공개SW 활용 장려 이유	90
[그림 4-44] 기업들의 공개SW 활용 제약 이유	91
[그림 4-45] 공개SW 관리 활동의 적절성 여부	92
[그림 4-46] 공개SW 관리 활동 (복수 응답)	92
[그림 4-47] 공개SW 관리 필요성	93
[그림 4-48] 공개SW 관리가 필요한 이유 (복수 응답)	93
[그림 4-49] 공개SW 활성화를 위한 필요 요소 (복수 응답)	94

[그림 4-50] 공개SW의 홍보 필요성	95
[그림 4-51] 공개SW 홍보의 필요 이유 (복수 응답)	95
[그림 4-52] 공개SW 관련 정부/관련기관에 대한 요청사항 (복수 응답)	96
[그림 4-53] 공개SW 홍보의 불필요 이유 (복수 응답)	96
[그림 5-1] 공개SW 개발 방식의 장점들	99
[그림 5-2] 공개SW 개발 방법	101
[그림 5-3] 안전 분야 공개SW 활용 방안 요약	102
[그림 5-4] 안전 분야 공개SW 활용을 위한 활성화 방안	103
[그림 5-5] 안전 분야 공개SW 활성화를 위한 인력 양성 방향	106
[그림 5-6] 안전 분야 공개SW 활성화를 위한 거버넌스 구축 지원	108
[그림 5-7]	116

- 표의 일련번호는 <표 1-1>, <표 1-2>, . . . <표 2-1>과 같이 순서대로 부여하며, 앞의 번호는 장을 지칭하고 뒤의 번호는 표의 일련번호를 지칭. 세모괄호 사용
 - 단위는 표의 우측 상단에 괄호안에 표기함을 원칙으로 한다.
 - 〈예〉 (단위 : 천원, 명, %) - 순서는 수치의 유형에 따라 정한다.
 - 표에 딸린 주(註, note)와 자료(source)는 주, 자료의 순으로 적는다.
- 표를 제외한 **그림, 그래프, 사진** 등은 모두 [그림 1-1], [그림 1-2], . . . [그림 2-1]과 같이 순서대로 부여하며 앞의 번호는 장을, 뒤의 번호는 표의 일련번호를 지칭. 네모괄호 사용.
- 표 및 그림 작성시 해당 페이지의 여백을 고려하여 지나치게 작거나 크지 않게 적절히 조정하고, 표제목·그림제목 및 표번호·그림번호는 반드시 작성한다.

요 약 문

1. 제 목

소프트웨어안전 분야에서 공개소프트웨어 활성화 방안

2. 연구 필요성 및 목적

1) 필요성

공개소프트웨어는 제4차 산업혁명의 핵심 SW기술들인 인공지능, 빅데이터, 클라우드 분야의 기술혁신을 주도하고 있다. 또한 해외에서 자율주행 등 소프트웨어안전 분야 역시 공개SW의 활용하는 사례가 늘어나고 있다. 이에 반하여 국내에서는 SW안전 분야의 경쟁력 강화를 위한 공개SW 활용에 대한 논의조차 없으며 안전 확보를 위해서 공개SW를 활용할 수 없다는 의견이 다수이다.

하지만 공개소프트웨어 기반의 인공지능, 빅데이터 등의 기술들은 제4차 산업혁명과 맞물려 안전 분야에 영향을 주기 시작하고 있기 때문에 국내외 SW안전 분야에서 공개소프트웨어 현황을 파악하고 소프트웨어안전 정책에 있어 활용할 수 있는 방안 마련이 필요한 시점이 되었다.

2) 목적

이 연구는 SW 신기술 분야인 인공지능, 빅데이터, 클라우드 분야에서 공개SW 동향을 알아보고, 공개SW가 SW안전 분야에서 어떻게 활용되고 있는지 살펴본다. 그리고 국내 SW안전 분야에서 공개소프트웨어 현황을 설문 조사를 통해 분석한 후에 SW안전 분야의 경쟁력 강화를 위한 공개소프트웨어 활용 방안 및 이를 활성화하기 위한 방안을 제시한다.

3. 연구의 구성 및 범위

이 연구는 총 6장으로 구성되어 있다.

제 1장은 서론으로 SW안전 분야의 공개SW 활성화 방안에 대한 연구의 배경, 필요성, 연구의 목적 및 연구 수행 방법을 제시한다.

제 2장은 이론적 배경으로 SW안전과 공개SW에 대해 살펴본다. 첫 번째로 SW 안전 개념과 안전 확보를 위한 절차에 대해 살펴본다. 두 번째로 공개SW의 개념과 진화 과정에 대해 살펴본다.

제 3장은 SW 신기술 분야 및 SW안전 분야의 공개SW 동향을 살펴본다. 우선 인공지능, 빅데이터, 클라우드 등 제4차 산업혁명의 주요 SW 기술 분야에서 공개SW 동향을 알아보고 SW안전 분야인 자율주행 및 다른 분야들에서 공개SW 활용 동향을 해외를 중심으로 살펴보고 시사점을 제시한다.

제 4장은 국내 SW안전 분야에서 공개SW 현황을 파악하기 위한 설문 조사 결과를 소개한다. 해당 조사는 주요 안전 분야인 자동차, 철도, 항공(국방) 분야의 총 30개 기업들을 대상으로 해당 기업에서 SW안전 관련 업무를 수행하는 대상자를 선정하여 설문 조사를 하였다. 총 30명의 설문 대상자들을 설문 결과를 가지고 국내 SW안전 분야의 공개SW 현황을 분석하고 시사점을 제시한다.

제 5장은 국내 SW안전 분야의 경쟁력 강화를 위한 방안으로써 공개SW 활용 방안과 활성화 방안을 제언한다. 우선 공개SW 활용 방안으로 설문 결과 SW 활용도가 높은 안전 요구사항 분석 및 테스트 관련 SW와 자율주행 관련 SW 개발을 공개SW 개발 방식을 활용할 것을 제안한다. 그리고 국내 안전 분야의 공개SW 활성화 방안으로 공개SW 바로 알기, 고급 SW 개발자 양성, 공개SW 거버넌스 구축 지원을 제안한다.

마지막으로 제 6장은 결론으로 연구 요약, 향후 연구 방향, 연구 한계점 등으로 마무리한다.

4. 연구 내용 및 결과

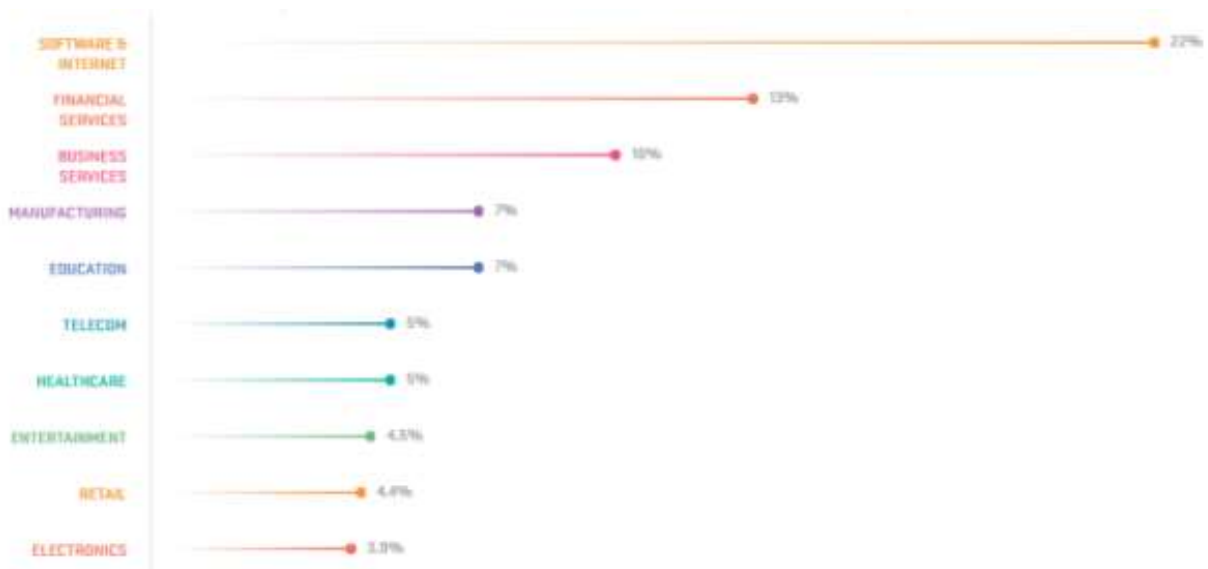
1) 기술 배경

SW 기술은 이제 다양한 산업에서 널리 활용되고 있으며 특히 안전과 밀접한 산업에서도 활용되고 있다. 원자력, 철도, 항공, 자동차 같은 안전 산업들도 사고 예방을 위한 안전 기술에 SW를 활용하고 있다. 반면에 SW가 이들 산업에서 활용되면서 SW 결함으로 인한 안전사고의 가능성이 상대적으로 늘어나고 있다.

자율 주행의 경우 SW 기반의 인공지능, 빅데이터, 클라우드 기술들이 자동차 산업에 활용되어 기술을 혁신하고 있다. 그 결과 자율 주행 시범 운행에서 안전사고가 이미 발생하였고 SW안전의 중요성을 인식시키는 계기가 되었다. 이에 SW 안전 개념이 최근 등장하였고 SW안전 확보를 위한 개발 절차들이 등장하고 있다.

특히 제4차 산업혁명의 핵심 기술들인 인공지능, 빅데이터, 클라우드 등 SW 신기술들의 확산은 각종 산업에 활용되어 인류 삶의 혁명적인 변화를 야기하고 있다. 이러한 SW 신기술 혁신의 배경에는 공개소프트웨어 개발 방식이 있다. 공개SW 개발 방식은 1980년대 자유SW 운동이 탄생하면서 개방·공유·협업의 정신을 SW 개발과 배포 과정에 도입하고자 등장하였다. 그리고 1990년대 말 오픈소스 개념이 등장하면서 공개SW 개발 방식이 상업적으로 활성화되기 시작하였고 구글, 아마존, 마이크로소프트, 페이스북 등 글로벌 IT 기업들의 생태계 조성 및 경쟁력 강화에 활용되기 시작하였다. 이러한 공개SW 개발 방식의 확산으로 인하여 공개SW 개발 방식은 SW 산업을 넘어 금융, 제조, 교육 분야에서도 활용되고 있다.

대표 그림 1: 2017년 깃허브 동향에 포함된 공개SW 활용 분야



출처: 깃허브, GitHub Octoverse 2017

2) 국내외 연구 현황

최근에 급격히 중요해지고 있는 SW안전 분야의 경쟁력 강화를 위해서 SW산업의 기술 혁신을 주도하는 공개SW 개발 방식을 활용은 빠르게 기술 추격을 할 수 있는 기회를 제공해 줄 수 있다. 하지만 국내 안전 분야에서 공개SW가 어떻게 활용되고 있는지에 대한 국내 연구 자료는 매우 부족하고 특히 국내 안전 분야에서 공개SW 활용 현황 조사 자료는 전무하다.

따라서 SW안전 분야에서 공개SW 개발 방식을 활용을 위한 정책 수립을 위한 국내 현황 조사가 필요하다. 우선 국내외 현황 비교를 위해 해외는 인터넷 검색을 통한 문헌 조사로 SW 신기술 분야와 안전 분야의 공개SW 동향 조사를 수행하였고 국내는 기업들을 대상으로 설문 조사를 수행하여 보다 정확한 자료를 얻고자 하였다.

3) 해외 안전 분야 관련 공개SW 동향

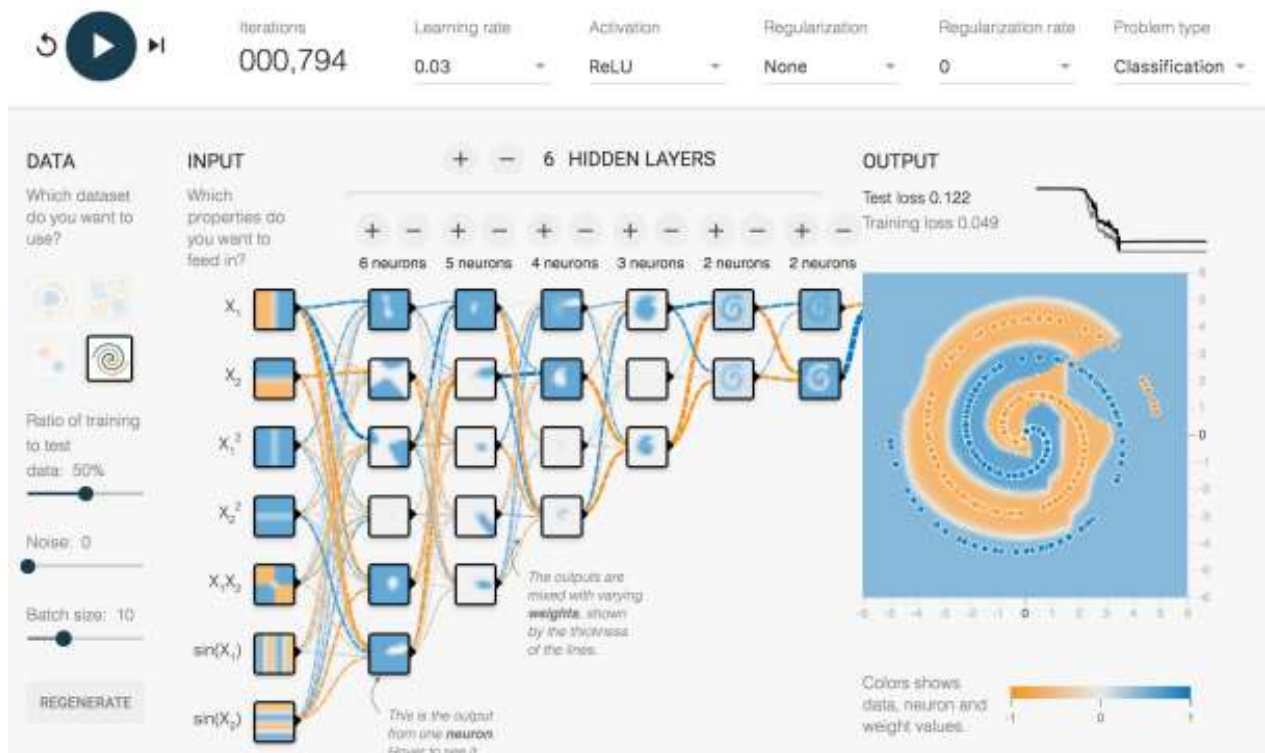
SW신기술인 인공지능, 빅데이터, 클라우드 분야는 현재 공개소프트웨어 개발 방식을 가장 널리 활용되고 있는 분야이다.

인공지능 분야 중 공개SW 개발 방식이 가장 많이 활용되는 심층학습 분야의 경우 구글, 마이크로소프트, 아마존, 페이스북, 바이두 등 기업들이 텐서플로우, Accord.Net, CNTK, Mahout, Caffe 등의 공개SW프로젝트들을 주도하고 있다.

빅데이터 분야도 아파치 재단의 하둡 생태계를 국내의 삼성을 비롯한 많은 기업들이 활용하고 있으며 R 역시 CRAN(Comprehensive R Archive Network)에 등록된 무수한 패키지들을 윈도, 맥, 리눅스 등 다양한 환경에서 활용할 수 있도록 발전하였다.

클라우드 분야는 현재 공개SW를 상업적으로 가장 잘 활용되고 있는 분야로 아마존, 구글, 마이크로소프트 등은 공용 클라우드 서비스의 기능 개선 및 신규 기능 개발을 위해 오픈스택 재단과 클라우드파운드리 재단을 후원하거나 공개SW 개발 방식으로 개발된 도커, 쿠버네티스 같은 기술들을 활용하여 IT 인프라 시장의 변화를 야기하고 있다.

대표 그림 2: 텐서플로우를 활용한 인공 신경망 모델링

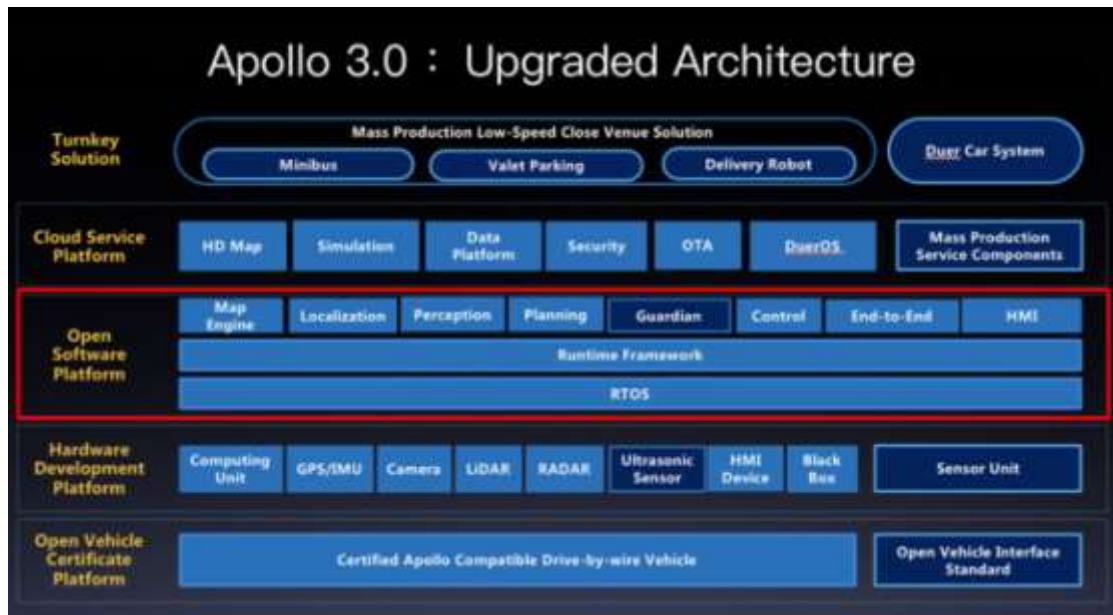


출처: 텐서플로우 블로그

안전 분야 역시 공개SW 개발 방식을 활용하는 사례들이 늘어나고 있는데 대표적인 분야가 자동차 분야이다.

자동차 분야의 경우 2009년에 차량용 멀티미디어 플랫폼을 표준화하기 위해 설립된 GENIVI에서 공개SW 개발 방식을 도입하였고 2018년 현재 113개의 회원사를 보유하고 있다. 2013년 차량용 임베디드 리눅스를 개발하기 위해 리눅스 재단의 프로젝트로 설립된 AGL(Automotive Grade Linux)도 공개SW 개발 방식을 도입하여 차량용 운영체제와 프레임워크를 개발하고 있다. 2018년 도요타와 아마존이 공식 지원한다고 발표하고 차량용 기능안전 및 첨단 운전자 보조 기능을 개발하려고 하고 있다. 바이두의 아폴로 프로젝트는 기업이 주도하는 자율주행 프로젝트로 2017년 처음 공개되었으며 공개SW 개발 방식을 활용하여 2021년까지 완전 자율 주행 기능을 개발하려고 하고 있다.

대표그림 3: 아폴로 3.0 플랫폼



출처: 바이두 아폴로

이 외에 철도 분야에서도 유럽 국가간 철도 제어 시스템의 호환성 제공을 위한 OpenETCS(European Train Control System) 프로젝트가 2015년까지 진행되었으며 항공 분야에서는 록히드 마틴이 SteamFlow라는 빅데이터 분석 플랫폼을 공개한 일부 사례들이 있다.

4) 국내 안전 분야 공개SW 현황

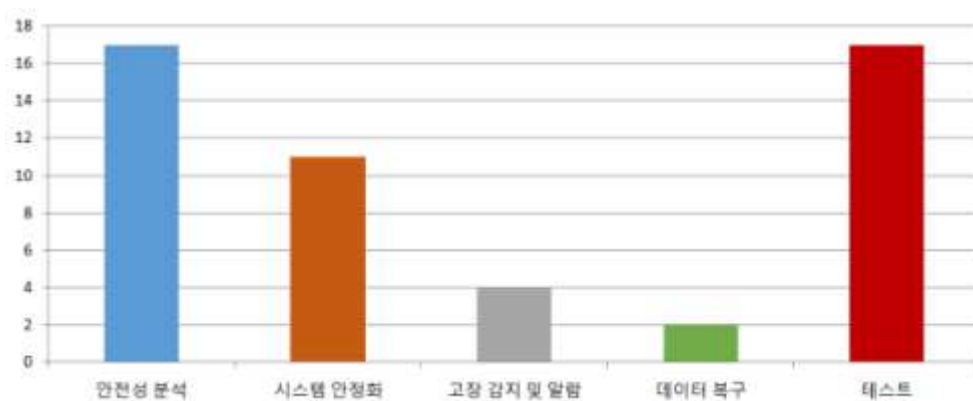
국내 안전 분야에서 공개SW 활용 현황을 파악하기 위해 자동차, 철도, 항공(국방) 분야별 10개의 기업들을 선정하여 총 30개 기업들의 SW안전 종사자들을 대상으로 설문 조사를 수행하였다. 설문 조사 내용은 설문 대상자들의 기본 정보, 국내외 제도 및 지침, SW 활용 현황, 공개SW 이해도, 공개SW 활용 현황 관련 설문 문항들로 구성하였다. 이에 설문 결과를 소개하고 이를 분석하여 안전 분야 기업들의 공개SW 활용 현황에 대한 시사점을 도출하였다.

기본 정보로써 조사 대상 기업들의 규모는 세 분야를 모두 포함하여 대기업 9개, 중견기업 8개, 중소기업 13개로 대체적으로 고르게 선정되었다. 설문 대상자

들의 담당 업무는 개발 업무 24명, 품질 관리 업무 4명, 개발 부서 관리자 2명으로 안전 분야의 SW 관련 업무를 수행하며 해당 회사 근무 기간은 10년 이상에서 20년 미만이 9명, 5년 이상에서 10년 미만이 12명, 3년 이상에서 5년 미만이 12명이고 3년 미만은 1명에 불과할 정도로 회사 내부 상황을 잘 알고 있는 경력자들로 선정되었다.

안전 분야 기업들은 SW개발 시 안전 관련 표준 또는 사내 지침 준수를 문의하는 복수 응답 설문에서 30개 기업들 중 국제 표준을 준수한다는 기업이 16개로 가장 많았고 국내 산업 규격 및 사내 지침을 준수한다는 기업들이 각각 10개였으며 인증을 받는다는 기업이 3개로 가장 적었다. SW안전성 확보를 위한 필요사항에 대한 중복 응답 설문에서 안전성 분석과 테스트가 필요하다는 의견이 각각 17개로 제일 많았다. 그리고 공개SW 관련 지침 및 조직 보유에 대한 설문에는 사내 지침이 있다는 응답이 8개였고 전담 조직이 있다는 응답이 4개로 안전 분야 기업들은 대체적으로 공개SW 관리를 하지 않고 있음을 알 수 있었다. 만약 SW안전 확보를 위한 R&D가 진행된다면 안전성 분석 및 테스트같이 안전 확보에 가장 필요한 분야로 지원할 필요가 있다.

대표그림 4: SW안전성 확보를 위한 필요사항 (복수 응답)

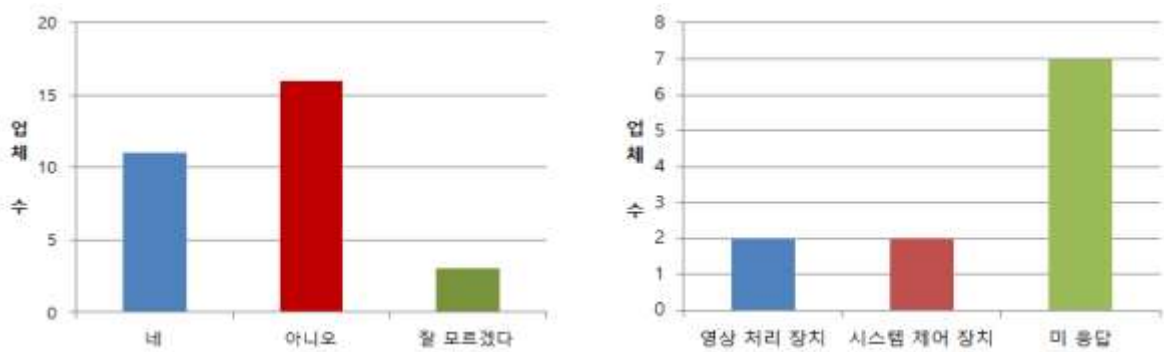


안전한 SW 개발을 위해 SW 활용 현황 관련 복수 응답 설문에서 개발 단계에서 SW를 활용한다는 기업이 24개로 가장 많았고, 시험 도구 SW를 활용한다는 기

업이 18개였고, 프로세스 지원 도구 SW를 활용한다는 기업이 14개이었고 요구 사항 및 설계 도구를 사용한다는 기업이 각각 4개와 3개로 소수 의견이 있었다.

그리고 개발하는 시스템에서 공개SW를 활용 여부를 묻는 설문에는 11개의 기업에서 공개SW를 사용하고 있다고 응답하였고 16개 기업에서 사용하고 있지 않는다고 응답하였으며 모른다는 응답도 3개로 나왔다. 그리고 11개의 기업 중 1개를 제외한 10개의 기업에서 공개SW를 내부적으로 사용하고 있다고 응답하였다.

대표그림 5: 개발 시스템에서 공개SW 사용 현황



공개SW를 사용하는 이유에 대한 복수 응답 설문에 11개 기업 중 6개 기업에서 개발 리소스 절감을 선택했고 3개의 기업은 높은 자유도와 편의성이라고 응답하였다. 반대로 공개SW를 사용하지 않는 이유에 대한 복수 응답 설문에 16개 기업 중 13개 기업이 기술 지원에 대한 제약을 선택했고 10개의 기업은 상용SW와 비교시 낮은 성능을 꼽았다.

국내 안전 분야 기업들의 약 1/3 정도만 개발 시스템에서 공개SW를 활용하고 있으며 그 이유는 개발 리소스 절감과 공개SW의 높은 자유도와 편의성 때문임을 알 수 있었다. 반대로 사용하지 않는 기업들은 기술 지원 제약 및 상용SW 대비 낮은 성능을 선택한 결과 국내 공개SW 활성화를 위해서는 개발자 역량 강화가 중요한 요소로 볼 수 있다.

안전 분야 기업들의 공개SW에 대한 이해도를 확인한 결과 30개 기업 중 잘 알고 있다는 응답은 1개였고 대부분 알고 있다는 응답은 4개에 불과했으며 개념

정도 알고 있다는 응답이 20개와 모름이라는 응답이 5개로 약 5/6 정도의 기업들이 공개SW에 대하여 거의 모르고 있다고 볼 수 있었다. 또한 라이선스 이해도를 묻는 설문에도 10명의 응답자가 모른다고 선택했으며 개념 정도만 알고 있다는 응답이 16명이어서 공개SW 이해도와 유사한 결과를 얻었다. 이를 토대로 생각하면 국내 안전 분야 공개SW 활용을 위해서는 1차적으로 공개SW에 대한 이해도를 향상시킬 수 있는 홍보 정책들이 선행될 필요가 있다.

대표그림 6: 기업 및 설문 응답자의 공개SW 개념 이해도

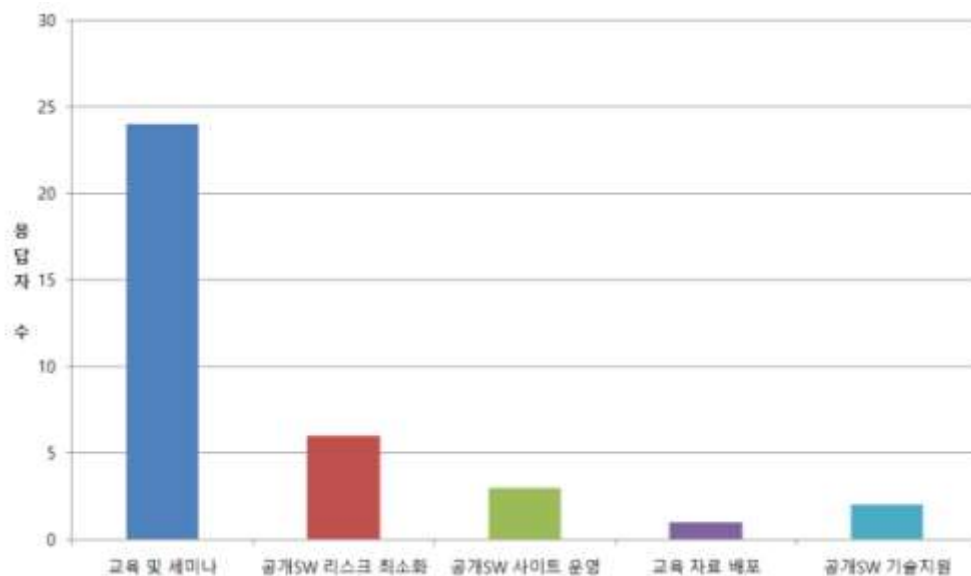


안전 분야 기업들을 대상으로 공개SW 현황 조사 대상자 30명에게 공개SW 교육(세미나 경험 유무에 대한 응답에 단지 3명만이 경험이 있다고 하였고 나머지 27명은 경험이 없다고 하였다. 그리고 공개SW에 대한 기업 인식에 관련된 설문에서 단지 5개의 기업만이 장려하고 있고 나머지 25개 기업 중 제약한다는 기업이 10개이며 관련 없거나 모르겠다라는 응답이 15개에 이를 정도로 공개SW에 대한 인식이 부정적이었다.

이러한 부정적인 인식과 달리 설문 응답자들은 기업 내부에서 공개SW 관리의 필요성에 대한 응답에 19명이 필요하다고 응답하였고 불필요하다는 응답은 단지 3명이었으며 모르겠다라는 응답은 8명이었을 정도로 공개SW 관리의 필요성을 인지하고 있었고 전반적으로 라이선스 교육, 공개SW 활용 방안에 대한 정보의 필요성도 인식하고 있었다.

따라서, 공개SW 활성화를 위한 정부 및 공공기관에 대한 요청사항에 대한 복수 응답설문에 교육 및 세미나에 대한 응답이 24명으로 압도적으로 많았으며 공개SW 리스크 최소화가 6명이었고 나머지는 공개SW 관련 사이트 운영, 공개SW 기술 지원, 교육 자료 배포로 나왔다.

대표그림 7: 공개SW 관련 정부/관련기관에 대한 요청사항 (복수 응답)



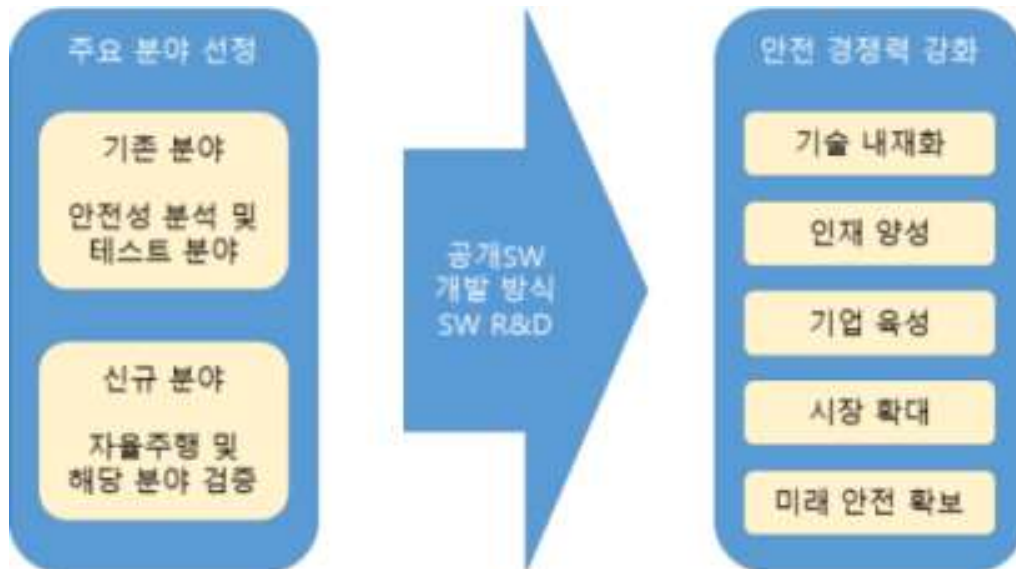
5) 소프트웨어안전 분야 공개SW 활성화 방안

SW안전 분야의 경쟁력 강화를 위해서는 공개SW 활성화가 필요하기 위해서 우선적으로 공개SW 개발 방식의 활용 방안을 제안한다. 공개SW 개발 방식은 개발 효율성, 최신 기술 도입, 기업 저변 확보, 기술 내재화, 인재 양성 등 다양한 장점들을 가지고 있기 때문에 SW안전 분야에서 공개SW 개발 방식을 활용한다면 기술 경쟁력과 산업 경쟁력 강화에 많은 도움이 될 수 있다.

우선 공개SW 개발 방식을 적용한 연구 개발 분야로 기존 분야에서 안전 확보에 가장 많이 필요하다고 응답된 안전성 분석 및 테스트 분야와 신규 분야에서 제4차 산업혁명의 주요 분야이면서 안전 분야에서 공개SW 개발 방식이 가장 활발한 자율주행 관련 분야를 선정하고 관련 연구 개발 사업을 추진할 필요가 있다. 이런 공개SW 개발 방식으로 SW안전 기술 내재화, 인재 양성, 기업 육성, 시

장 확대 효과를 얻고 이를 토대로 미래 사회 안전을 확보함으로써 안전 산업 경쟁력 강화에 이바지할 수 있다.

대표그림 8: 안전 분야 공개SW 활용 방안



대표그림 9: 안전 분야 공개SW 활성화 방안



하지만 안전 분야에서 공개SW 활용 방안을 성공적으로 수행하기에는 이번 설문조사 결과 국내 안전 분야의 공개SW 관련 기반(인식 부족, 개발자 부족, 관리 부족)이 매우 부족하다. 따라서 안전 분야 공개SW 활용의 성공을 위해서는 안전 분야의 공개SW 활성화 정책이 뒷받침이 필요하다.

우선적으로 공개SW 바로 알기를 위해 공개SW 사업화 방안, 해외 진출 방안, 라이선스 관련 정보 같은 정보 제공으로 공개SW에 인식을 제고할 필요가 있다. 다음으로 공개SW를 제대로 활용하고 공개SW 개발 방식으로 SW를 개발할 수 있는 고급 SW 개발자 양성이 필요하다. 마지막으로 공개SW를 안전하고 효율적으로 사용할 수 있는 환경을 제공하기 위한 공개SW 거버넌스 구축 지원이 필요하다.

5. 정책적 활용 내용 및 기대 효과

이 보고서는 안전 분야 경쟁력 강화를 위한 SW안전 기술 확보 정책과 SW안전 기업 육성 정책 수립을 위한 기초 자료로 활용될 수 있다.

SW안전 기술 확보 방안으로 공개SW 개발 방식을 활용하여 안전성 분석 및 테스트 SW 관련 기술 연구 개발 및 제4차 산업혁명의 주요 분야인 자율주행 관련 기술 연구 개발을 제안하였다. 최근 SW산업의 기술 혁신의 중심인 공개SW 개발 방식을 활용함으로써 수준 높은 연구 개발물을 획득을 통한 기술 내재화, 인재 육성 등을 기대할 수 있다.

SW안전 기업 정책의 일환으로 안전 분야의 공개SW 활성화하기 위한 공개SW 바로 알기, 고급 SW개발자 양성, 공개SW 거버넌스 구축 지원을 제안하였다. 이미 자동차 분야는 공개SW 활용을 통한 기술 혁신이 해외에서 시작하고 있기 때문에 새로운 산업 트렌드를 기업들에게 전파함으로써 기업 역량을 강화시키고 고급 인재를 공급함으로써 기업 경쟁력을 강화를 기대할 수 있다.

최종적으로는 SW안전 기술 확보와 SW안전 기업 육성을 통해 SW 중심의 자동화가 진행되는 제4차 산업혁명의 미래사회의 안전 확보에 기여할 수 있을 것이다.

SUMMARY

1. Title

Open Source Software Activation in Software Safety Fields

2. Study Needs and Objectives

1) Needs

Open Source Software(OSS) is leading innovation in Artificial Intelligence(AI), BigData, and Cloud areas, which are the core SW technologies of the Fourth Industrial Revolution. In addition, there are more cases of using OSS in software safety field such as autonomous driving from abroad. On the other hand, there are many domestic opinions that there is no discussion about OSS utilization for enhancing competitiveness of SW safety field and OSS can not be used to secure safety.

However, since OSS based AI and BigData technologies begin to affect the safety field in conjunction with the 4th Industrial Revolution, it is necessary to understand the status of OSS in domestic and overseas SW safety field. It is time to prepare plans to utilize OSS in software safety related policies.

2) Objectives

This study examines the OSS trends in the AI, BigData, and Cloud fields in new SW technologies and examines OSS usages in SW safety field. After analyzing the status of OSS in domestic SW safety field through surveys, it proposes ways to utilize OSS to enhance competitiveness of SW safety field and suggests ways to activate them.

3. Composition and Scope of Study

This study is consisted of 6 chapters.

Chapter 1 presents study background, needs, objectives and methods for OSS activation methods in SW safety field as the introduction.

Chapter 2 discusses SW safety and OSS as the theoretical background. Firstly, we discusses SW safety concepts and procedures for ensuring safety. Secondly, the OSS concepts and evolutionary process are examined.

Chapter 3 discusses OSS trends in new SW technologies and SW safety fields. Firstly, we studies OSS trends in the main SW technologies such as AI, BigData and Cloud and then examines OSS trends in SW safety fields such autonomous driving and others.

Chapter 4 introduces the survey results to understand the OSS status in domestic SW safety fields. The survey was conducted on 30 companies in the automobile, railway and aeronautical (defense) fields of the major safety areas, by selecting the persons who will perform the SW safety related works. The survey results of 30 respondents are analyzed and the implications are presented.

Chapter 5 suggests how to utilize and to enhance OSS in order to enhance competitiveness of the domestic SW safety fields. Firstly, we suggest using the OSS development method to develop analyzing SWs of the safety requirements and testing SW, and autonomous driving related SWs. Additionally, as a way to activate OSS usages in the domestic safety fields, we propose to inform OSS, to cultivate advanced SW developers, and to support the establishment of OSS governance.

Finally, Chapter 6 concludes with the study summary, the future study direction, and the study limitations.

CONTENTS

Chapter 1. Introduction	1
Section 1. Study Needs and Objectives	1
Section 2. Study Method	2
Chapter 2. Main Concepts	3
Section 1. Software Safety Overview	3
Section 2. Open Source Software	8
Chapter 3. Overseas Trends on Open Source Software Usages	17
Section 1. Software New Technologies Areas	17
Section 2. Main Safety Industry Areas	38
Section 3. Summary and Implications	55
Chapter 4. Open Source Softwares in Domestic Software Safety Fields	59
Section 1. Survey Conducting Method	59
Section 2. Survey Results	64
Section 3. Summary and Implications	97
Chapter 5. Methods of Open Source Software Activation in Software Safety Fields	99
Section 1. Methods of Open Source Software Utilization	99
Section 2. Methods of Open Source Software Activation	103
Chapter 6. Conclusion	109

Section 1. Study Summary 109

Section 2. Implications and Future Works 111

Reference 112

제1장 서론

제1절 연구 필요성 및 목적

산업이 발전함에 따라 많은 분야에서 지능화, 무인화, 네트워크화 등의 첨단화가 진행되고 있다. 이로 인해 기존 HW중심의 제조업에서 SW 비중이 점차 증가하게 되었고 사람과 밀접하게 연관된 시스템들이 SW 오류로 인한 사고의 위험에 노출되게 되었다. 이에 국제적으로 인간의 생명과 환경적인 피해를 끼칠 수 있는 분야의 안전 확보에 대한 필요성이 대두되었으며, 전기/전자/프로그래밍 가능한 전자 시스템의 안전 표준인 IEC 61508¹⁾이 제정되었다. 이후 안전 산업별로 IEC 61508을 기반으로 각 산업 분야에 특화된 안전 표준을 제정하여 준수하고 있으며, 대표적으로 자동차 분야의 ISO 26262, 철도 분야의 IEC 62279, EN 50128, 항공 분야의 DO-178B/C 표준이 있다. 이와 같이 안전 확보를 위해 많은 노력들을 기울이고 있다.

최근 제4차 산업시대의 ABCi(AI, BigData, Cloud, IoT)로 일컫는 핵심 SW 기술들이 발전함에 따라 안전 분야를 포함한 산업 전반에 걸쳐 초연결, 지능화 시대를 선도하고 있으며 스마트시티, 스마트팩토리, 헬스케어, 자율주행 등 새로운 시장들이 창출되고 있다. 이러한 핵심 SW 기술들의 진화에는 공개SW 개발 방식이 있다. 공개SW 개발 방식은 개방, 공유, 협업으로 SW를 개발하는 방법으로 다수의 참여자들이 SW개발에 참여함으로써 규모의 경제를 기반으로 생태계 구축에 매우 유리한 개발 방식이다. 이러한 공개SW 개발 방식의 장점을 활용하고자 공개SW의 활용 범위가 넓어지고 있으며 구글, 아마존, 마이크로소프트 같은 글로벌 기업들을 중심으로 적극적으로 공개SW 개발 방식을 활용하고 있다.

그 결과 공개소프트웨어는 제4차 산업혁명의 핵심 SW기술들인 인공지능, 빅데이터, 클라우드 분야의 기술혁신을 주도하게 되었고 SW산업의 비즈니스 혁신을 이끌고 있다. 또한 해외에서 자율주행 등 소프트웨어안전 분야 역시 공개SW의 활용하는 사례가 생기고 있다. 이에 반해서 국내는 안전 확보를 위해서 검증되지 않은 공개SW를 활용할 수 없다는 의견이 다수이며 SW안전 경쟁력 강화를 위한 공개SW 개발 방식의 활용에 대한 논의조차 없다. 하지만 공개SW 기반의 인공지능, 빅데이터 등의 기술들은 제4차 산업혁명과 맞물려 안전 분야에 영향을 주기 시작하고 있기 때문에 국내외 SW안전 분야에서 공개SW 현황을 파악하고 SW안전 정책에 있어 활용할 수 있는 방안 마련이 필요한 시점이 되었다.

1) IEC 61508 : 쉘 산업 영역을 포괄하는 전기/전자/SW 관련 안전 시스템의 기능 안전성 표준

이 연구의 목적은 SW안전 분야의 공개SW 현황을 파악하여 국내 SW안전 분야 경쟁력 강화를 위한 공개SW 활성화 방안을 제언하는 것이다. 공개SW 활성화 제언에 앞서 안전 분야에서 공개SW 활용 방안에 대하여 먼저 제언한다. 현실적이고 타당한 SW안전 분야에서 공개SW 활용 방안과 활성화 방안을 제언하기 위해 SW안전 분야와 관련된 국내외 공개SW 현황 조사를 수행하고 그 결과를 분석하여 시사점을 우선적으로 도출하고자 한다.

제2절 연구 방법

SW안전 분야에서 공개SW 활성화 방안을 연구하기 위해 총 3단계에 걸친 과정을 수행하였다. 1단계는 SW안전 분야의 공개SW 활성화 방안 제언을 위한 SW안전 분야 공개SW 현황 조사 방향 설정을 위한 문제 인식을 위한 단계로 제2장에 기술된 SW안전 개요 및 공개SW 개요로 정리되었다. 2단계는 제언을 위한 안전 분야 공개SW 현황 조사 및 시사점 도출 단계로 제3장 해외 공개SW 동향을 SW 신기술과 안전 분야로 구분하여 기술하였고 제4장 국내 안전 분야 현황 설문 조사 수행 결과를 기술하였다. 그리고 해외 동향과 국내 현황을 기반으로 공개SW 활용과 활성화 제언을 위한 시사점을 도출하였다. 마지막 단계인 3단계는 도출된 시사점을 기반으로 국내 SW안전 분야 경쟁력 강화를 위한 공개SW 활용 방안과 활성화 방안을 제언하였다.

[그림 1-1] 안전 분야 공개SW 활성화 방안 연구 방법



제2장 주요 개념

제1절 소프트웨어안전 개요

SW기술의 발전은 기존 SW산업만이 아니라 다양한 타산업에도 활용되고 있으며 특히 안전과 밀접한 산업에도 널리 활용되고 있다. 원자력, 철도, 항공, 자동차 산업들은 시스템 결함에 의한 사고로 인하여 인명이나 재산상의 피해를 크게 줄 수 있기 때문에 이들 산업 분야는 사고를 예방하기 위한 안전 기술들이 발전하게 되었다. 최근 SW기술들이 이들 산업에서 보편적으로 활용되면서 SW로 인한 안전사고의 가능성이 점차적으로 늘어나고 있으며 제4차 산업혁명의 핵심 기술들인 인공지능, 빅데이터, 클라우드 등 SW 신기술들의 확산은 해당 산업 분야에서 안전과 SW를 더욱 밀접하게 연결시키고 있다. 따라서 최근 SW의 안전 확보 및 SW를 활용한 안전 확보에 대한 관심이 늘어나고 있다.

1. 안전 및 소프트웨어안전

국제 안전규격을 위한 가이드인 ISO/IEC GUIDE51²⁾은 안전을 ‘수용할 수 없는 위험이 없는 것(freedom from unacceptable risk)’ 이라고 정의하고 있다. 이는 다르게 표현하면 시스템 혹은 SW에서 받아들일 수 있는 위험은 안전에 위배되지 않는다고 해석할 수 있다.

즉 시스템 또는 SW에서 대비할 수 있는 위험은 안전에 영향을 주지 않는다고 할 수 있다. 시스템 관점에서 안전을 주요 안전(Primary safety), 기능안전(Functional safety), 간접 안전(Indirect safety)으로 구분할 수 있다. 주요 안전은 화재, 감전과 같은 하드웨어에 의한 직접 사고로부터 안전을 정의한다. 기능안전은 위험 평가에 의해 설계 과정에서 위험이 제거하기 위한 안전을 의미한다. 간접 안전은 데이터베이스 정보 에러 같이 잘못된 정보로 발생할 수 있는 위험에 대한 안전을 말한다³⁾.

소프트웨어와 연관된 안전 개념에 대한 정의로써 SW안전이라는 개념이 국내에서는 따로 정의하고 있다. 2018년 12월 기준 전면 개정중인 소프트웨어산업진흥법의 제2조에 포함된 SW안전은 ‘외부로부터의 침해행위가 없는 상태에서 소프트웨어의 내부적인 오작동 및 안

2) ISO/IEC Guide 51 (2014), "Safety aspects - Guidelines for their inclusion in standards"

3) Smith, David J. and Simpson, Kenneth G. L. (2004), "Functional Safety(A Straightforward Guide to Applying IEC 61508 and Related Standards)", Hutterwirth-Heinemann

전기능(사전 위험분석 등을 통하여 위험발생을 방지하는 기능을 말한다) 미비 등으로 발생할 수 있는 사고로부터 사람의 생명이나 신체에 대한 위험에 충분한 대비가 되어 있는 상태'로 정의하고 있다.

국내에서 SW안전 개념은 ISO/IEC GUIDE51의 안전 개념보다 매우 구체적으로 정의되고 있으며 SW로 인하여 발생하는 오작동 및 안전기능이 부족해서 발생하는 사고에 의해 사람의 생명이나 신체에 위험하지 않는 상황을 SW안전으로 보고 있다. 이에 대해 자세히 설명하기 위해 안전 기능과 안전무결성에 대해 알아야 한다.

안전 기능(Safety Function)은 안전한 소프트웨어를 개발하기 위한 소프트웨어의 여러 속성 중 하나로 시스템이나 장비의 안전 확보를 위해 입력 조건에 대하여 사전에 정의한 대로 의도된 안전 기능을 올바르게 수행해야 한다. 따라서 개발 과정에서 제품의 기능 안전을 확보하기 위해 안전 기능을 도출하는 단계, 도출된 기능을 구현하는 과정에서 원하는 수준의 안전성을 확보할 수 있는 방안을 마련하는 단계, 이렇게 크게 두 가지 부분으로 나누어서 살펴볼 수 있다. 그리고 안전을 위해 안전 기능이 올바르게 선정되었는지를 확인해야 하고 최종 개발 결과물에서 최초 의도한 안전 기능이 정상적으로 작동하는가를 확인해야 한다.

안전 무결성은 위험이 문제가 되지 않는 상태인 수용 가능한 위험만을 남겨두고 나머지 위험은 제거 또는 감소시킨 상태를 의미한다. 시스템 설계 단계는 여러 큰 위험(risk)들이 존재하기 때문에 안전상 불안한 상황으로 볼 수 있다. 따라서 개별 위험에 대하여 설계, 제작 및 운영 단계에서 각각의 안전 기능 또는 안전 대책을 마련하여 위험성의 크기를 줄여야 한다. 이때 위험의 파급 효과가 감소되어 문제가 되지 않는 경우(수용 가능한 수준이 된 경우)에는 위험이 안전한 상태(Safety State)라고 할 수 있다. 안전 무결성(Safety Integrity)은 주어진 모든 조건하에 있는 안전관련 시스템이 주어진 시간 내에 요구되는 안전기능을 만족스럽게 수행함으로써 위험이 감소시키기 위한 방법이 필요한 상태라고 할 수 있다.

[그림 2-1] 위험과 안전의 관계



출처 : 정보통신산업진흥원(2017), 『공통 분야 소프트웨어신뢰·안전성 확보를 위한 가이드』

드』

안전기능 요구사항은 위험원 분석을 통해 도출되고, 안전 무결성 요구사항은 위험성 평가를 통해 도출된다. 그리고 도출된 결과에 의해 안전 무결성 수준(Safety Integrity Level)이 최종적으로 결정된다. 안전 무결성 수준이 높을수록 해당 장비 또는 시스템의 고장 발생 가능성을 낮게 하여 안전 사고의 파급 효과를 줄여야 한다.

따라서 안전한 소프트웨어 개발은 시스템 및 장비의 안전 확보를 위해 필요한 요소로서 소프트웨어 개발 단계를 체계화함으로써 안전에 대한 관리를 수행할 수 있다. 그 결과 소프트웨어 개발 프로세스에 대한 접근법은 소프트웨어 중심의 프로세스 관점에서 시스템과 소프트웨어를 동시에 바라보는 개념으로 변화하고 있다. 이와 같은 변화는 소프트웨어를 전체 시스템의 일부로 인식하여 전체 시스템 레벨에서 위험성 분석을 우선적으로 수행해야 하고, 분석 결과를 토대로 시스템 안전요구사항을 도출해야 한다는 것을 의미한다. 결국 소프트웨어 안전을 보장한다는 의미는 소프트웨어가 가지고 있는 여러 가지 속성 중에서 특히 안전기능(Safety Function)이 올바르게 선정되었는지를 확인해야 하고, 최종 결과물이 수행하는 안전기능이 정상적으로 작동하는가를 확인해야 한다.

이러한 목적을 위해서는 다음의 세 가지 요소에 대한 고려가 필요하다⁴⁾.

첫째, SW개발 안전 수명주기에 대한 이해이다. SW개발 조직 내의 모든 개발자들이 SW개발 안전 수명주기를 이해하고 단계 결과물들을 사전에 계획하고 계획에 의해 만들어 나가야 한다. 그리고 업무의 특성에 따라 분야별 국제 표준을 준수하여 개발을 진행하여야 한다.

둘째, 시스템 및 SW 위험 분석 방법에 대한 이해가 필요하다. SW의 안전 확보를 위해서는 시스템 수준의 위험분석 수행하여 시스템 안전 요구사항을 도출하고 이를 만족하기 위한 개발이 진행되어야 한다. 이를 위해서는 시스템 수준에서 도출된 안전기능 요구사항 중에서 SW를 활용해야 하는 요구사항들을 SW 관점에서 정의하고 명세화함으로써 이를 토대로 SW가 개발되어야 한다.

셋째, SW 안전기능이 의도한 바와 같이 구현이 되었는지를 확인하기 위한 정확한 검증 방법을 적용해야 한다. SW 안전성을 검증하기 위한 시험은 시스템 레벨에서 이루어져야 하며 검증 목표는 시스템 안전 요구사항 및 SW 안전 요구사항을 충족하고 있는지를 검증하는 것이다. 개별 안전 표준은 각 수명주기별 수행해야 할 안전 활동을 위한 수준별

4) 정보통신산업진흥원(2017), 『공통 분야 소프트웨어신뢰·안전성 확보를 위한 가이드』

Technique/Measure, Method, Objectives 등을 제시하고 있다.

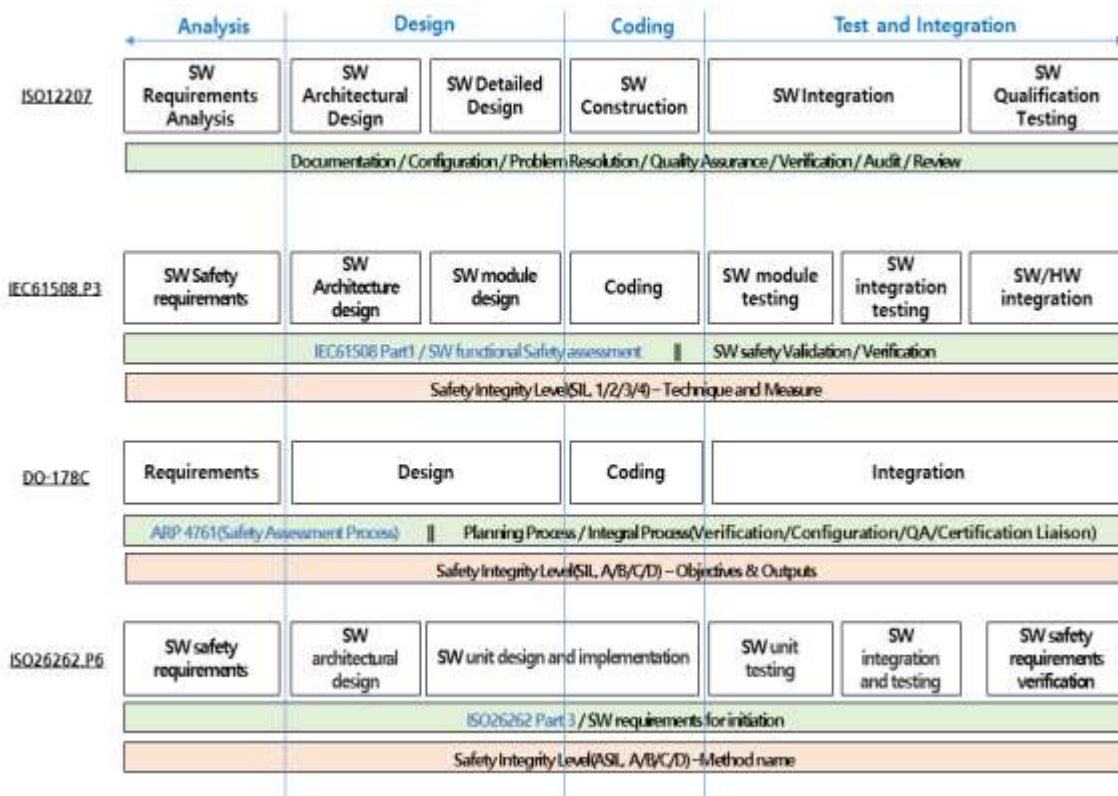
2. 안전 확보 방안

SW의 안전 확보를 위해서는 SW개발 과정에서 안전 요구사항을 만족하는 개발을 필요로 한다. 이를 체계적으로 정리하기 위해서 국제적으로 안전 표준을 정의하였고 대표적인 SW 안전 표준으로는 국제 기능 안전 모표준이라 할 수 있는 IEC 61508 표준에서 SW개발 절차를 다루는 Part3, 항공분야 SW안전 개발 표준인 DO-178C, 자동차 분야의 안전 표준인 ISO 26262에서 SW안전 개발 절차가 제시되어 있는 Part6들이 있다.

전통적으로 SW 개발 절차는 오랜 기간 동안 상호 참조를 통해 서로 영향을 주면서 발전해왔기 때문에 SW개발 품질 절차와 SW안전 절차간의 기본적으로 유사성이 있다. 그러다 보니 IEC 61508 표준의 SW개발 절차는 ISO/IEC 12207에서 정의한 표준 SW 생명 주기 절차를 참조하고 있다. 두 표준 사이의 가장 큰 차이점은 안전한 시스템 개발을 위해 SW개발 전에 시스템 관점에서 전체의 위험을 분석하고 평가하는 단계를 거치는 것이다. 이 과정을 통해 안전 기능과 안전 무결성 수준이 도출되고 여기서 도출된 안전 기능이 개발 전 단계에서 안전 무결성 등급에 맞게 각 수명주기 단계별로 분석되고 설계되고 구현되고 통합되고 있는지에 대한 보증활동이 추가된다.

아래 내용은 품질 프로세스 표준과 3개의 분야별 안전 표준간의 소프트웨어 개발 프로세스를 비교하여 제시한 것이다. 요구 분석 단계는 모두 동일하게 수행하지만 안전 표준에서는 앞 단계에서 위험 분석 및 평가 단계가 더 들어가 있다.

[그림 2-2] 품질 프로세스와 안전개발 프로세스 비교



IEC 61508은 Part1에서 SW 기능안전 평가(SW functional safety Assessment)를 수행하고 DO-178는 ARP 4761⁵⁾에서 선행적으로 위험 평가를 하도록 규정되어 있으며 ISO/IEC 26262은 Part3에서 SW 위험 분석 및 평가를 수행하도록 되어 있다. 위험 분석 및 평가에 따라 도출된 SW의 안전기능 및 안전 무결성 수준이 SW개발 안전 요구사항으로 반영되어야 한다.

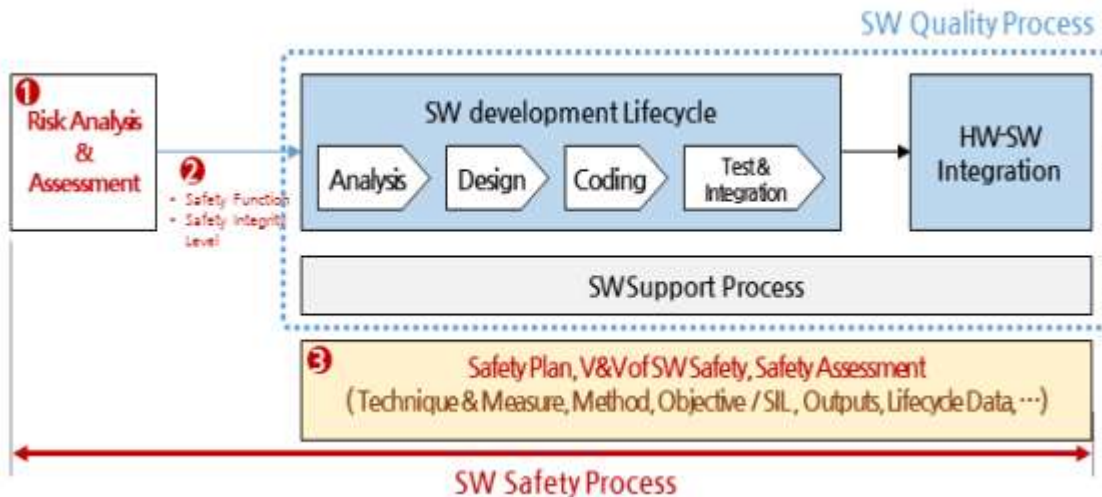
이후 SW개발 수명주기 동안 수행하는 활동은 품질 보증활동과 유사하다. 하지만 SW안전 개발 절차는 단계별 수행해야 할 활동, 기법, 산출물 등을 안전 무결성 수준에 맞게 제시해야 한다. 단계별 수행 내용은 프로젝트 수행 초기에 안전관리 계획에 의해 계획되고 실행되며 단계별 또는 절차 종료 시 안전성 보증에 대한 검토 및 확인을 수행해야 한다.

아래 그림은 앞서 비교한 그림을 SW품질 절차와 SW안전 개발 절차로 구분하여 간략화한 것이다. 품질 절차보다 더 추가적으로 수행해야 하는 내용은 먼저 위험 분석 및 평가이며 이를 통해 안전기능과 안전 무결성 수준을 결정하고 이 결정된 사항에 맞도록 각 개발 수

5) SAE(Society of Automotive Engineers)에서 1994-1996년 동안 개발되어 표준으로 발간된 "민간 항공 시스템 및 장비의 안전도 평가 수행 지침 및 방법(Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment)"

명주기에서 안전 보증 및 확인 활동을 수행한다.

[그림 2-3] SW품질 절차를 포함한 SW안전 개발 절차



이와 같은 SW안전 개발 절차는 SW가 가지고 있는 여러 가지 속성 중에서 특히 안전 요구사항이 올바르게 도출되었는지를 검증할 수 있어야 한다. 그리고 요구사항을 만족하는 기능안전이 제대로 구현되어야 하며 최종적으로 시스템 수준에서 개발한 안전기능들이 정상적으로 작동하는가를 확인해야 한다. 따라서 앞서 기술한 SW안전 개발의 핵심 내용에 대한 SW안전 표준의 세부 활동 및 기법/산출물 내에 반영이 필요하며 안전기능과 안전 무결성 수준을 만족하는지에 대한 각각의 Techniques, Measures, Methods and Objectives 등 전체 생애주기에 걸쳐 관리되며 기능안전성을 확보하기 위한 안전 보증 활동들이 고려되어야 한다.

제2절 공개소프트웨어

공개소프트웨어(SW)는 해외에서 오픈소스 (Open Source) 또는 오픈소스 소프트웨어로 보편적으로 불리며 SW의 핵심 요소인 소스 코드가 공개되어 정해진 라이선스 범위 안에서 사용자가 자유롭게 사용, 변경, 공유(배포)할 수 있는 SW를 의미한다. 공개SW는 1980년대 자유소프트웨어(SW) 운동으로 탄생한 자유SW로부터 시작되었으며 1990년대 후반 오픈소스 개념이 나오면서 진화를 통해 성장하였다. 현재는 대표적인 IT 회사인 구글, 아마존, 마이크

로소프트, 페이스북, IBM 등의 비즈니스에서 중요한 고려사항으로 성장할 정도로 널리 확산되었다.

1. 공개소프트웨어 소개

과거 SW는 HW 동작을 돕기 위한 하나의 구성요소로 인식되어 왔으나 SW 기술이 발전함에 따라 상업적으로 이용되기 시작하였다. 상용SW 시장이 성장함에 따라 상품화를 위한 경쟁력 강화를 위해 기존 SW 개발 과정의 공유와 협업보다는 폐쇄적인 개발을 통해 상업적 가치를 차지하려는 시도가 늘어남에 따라 SW의 파편화가 심화되기 시작하였다.

그에 따라 SW 발전이 저해되고 SW 기술 발전의 혜택이 소수에게 집중됨에 따라 반작용으로 SW 발전을 위해 소스코드를 공개하여 누구나 사용 가능하게 하고 SW 혜택을 다 같이 누리게 하고자 1980년대 리처드 스톨만(Richard Stallman) 중심으로 철학적인 자유소프트웨어 운동⁶⁾이 시작되었다. 자유소프트웨어 운동은 자유소프트웨어 개념이 탄생시켰고 이 운동을 확산시키고자 자유소프트웨어 재단(Free Software Foundation, FSF)⁷⁾이 1985년 설립되었다. 그리고 다양한 논의 끝에 자유소프트웨어를 누구나 자유롭게 SW를 실행하고 복사하고 배포하고 연구하고 수정 및 개선할 수 있는 SW로 규정하였다.

자유소프트웨어 재단은 누구나 사용할 수 있는 컴퓨터 운영체제를 확보하기 위한 노력으로 GNU 프로젝트⁸⁾를 추진하였고 많은 SW 개발자들이 GNU 프로젝트에 참여하여 파편화된 UNIX 운영체제의 단점을 극복하기 위해 노력하였다. 그리고 공유의 정신을 강조하기 위해 강한 코드 공개 조항을 가진 Copyleft 라이선스인 GNU GPL(GNU General Public License)이 1989년에 제정하였으며 GPL은 지금까지 공개SW의 대표적인 라이선스로 인식되고 있다.

그럼에도 불구하고 핵심 운영체제 프로그램을 확보하지 못하였으나 1991년 핀란드 헬싱키 대학의 리누스 토발즈가 리눅스 커널을 개발하여 GNU 프로젝트에 기부함으로써 공개SW의 발전에 중요한 전환점이 되었다. 우리나라에서도 1990년대 후반부터 공개SW의 가치를 인식하고 이를 확산하기 위한 공개SW 진흥 계획이 본격적으로 시작되면서 리눅스 보급 및 개발자들을 양성하였다.

이러한 초기 공개SW인 자유SW는 GPL 라이선스로 인하여 기업들의 적극적인 참여가 어려웠다. 그 이유는 GPL의 강한 코드 공개 조항으로 인하여 기존 공개SW를 활용하여 새로

6) GNU Project, Philosophy of the GNU Project (<https://www.gnu.org/philosophy/philosophy.html>)

7) Free Software Foundation, Free Software Foundation (<https://www.fsf.org/about/>)

8) GNU Project, What is GNU? (<https://www.gnu.org/>)

운 SW를 개발하여도 이를 공개해야 했으므로 개발된 SW의 독점적 지위를 누릴 수가 없었기 때문이다. 이러한 자유SW에 변화를 가져온 것이 1990년대 후반에 등장한 오픈소스 개념이다.

1990년대 후반 인터넷 익스플로러와 넷스케이프의 2차 브라우저 전쟁에서 마이크로소프트(MS)가 높은 윈도우 점유율을 기반으로 익스플로러를 무료로 보급하면서 넷스케이프의 시장 점유율이 급격히 줄어들게 되었다. 이에 따른 대응으로 넷스케이프사는 경쟁력 강화를 위해 소스코드를 공개하고 누구나 개발에 참여하고 넷스케이프를 무료로 사용할 수 있도록 하였다.

넷스케이프의 이 결정은 SW 기업들과 개발자들로부터 큰 영향을 주었고, 독점적 SW 기업들에 대항할 수 있는 하나의 가능성으로 인정받았다. 그리고 1998년 많은 SW 개발자들은 산호세에서 넷스케이프의 사례를 기반으로 공개SW 개발 방식을 활용하기 위한 논의 과정에서 오픈소스 개념이 탄생하게 되었다. 오픈소스는 자유SW처럼 소스코드가 공개되어 누구나 SW 개발에 참여할 수 있기 때문에 빠른 기술 혁신이 가능하였고 자유SW의 단점인 강한 코드 공개 조항을 완화시켰다. 이를 통해 기업들이 공개SW를 상업적으로 활용할 수 있는 가능성을 열어주었다.

오픈소스를 활성화하기 위해 비영리단체인 오픈소스 이니셔티브(Open Source Initiative, OSI)⁹⁾를 설립하였고 오픈소스 개념을 <표 2-3>처럼 10가지 특징들¹⁰⁾로 정의하였다. 그리고 OSI는 오픈소스 10가지 특징들을 따르는 오픈소스 라이선스를 승인해주고 있으며 2018년 12월 기준 약 80개 정도의 라이선스들을 승인하고 있다. GPL같은 강한 코드 공개 조항이 없는 아파치 라이선스, MIT 라이선스, BSD 라이선스 같은 허가적 라이선스(Permissive License)들도 오픈소스 라이선스로 인정하고 있다. 보다 자세한 한글 정보는 오픈소스SW라이선스 종합정보시스템(<https://olis.or.kr>)과 공개SW 포털(<https://www.oss.kr>)에서 공개SW 라이선스 정보를 얻을 수 있다.

<표 2-1> OSI에서 정의한 오픈소스의 10가지 특징

순서	OSI 규정	주요 내용
1	Free Redistribution	SW 판매나 양도를 제한하지 않고 자유롭게 재배포 허용
2	Source Code	소스코드와 컴파일 형태를 모두 배포

9) Open Source Initiative (<https://opensource.org>)

10) Open Source Definition (<https://opensource.org/osd>)의 10가지 특징을 기반으로 재구성

3	Derived Works	변경이나 2차저작물을 허용하고 원래의 SW 라이선스와 동일한 조건으로 배포를 허용
4	Integrity of The Author's Source Code	패치파일 형태의 재배포를 허용하지만, 원작상 변경된 소스코드로 빌드가 가능한 SW로 배포하여야 함
5	No Discrimination Against Persons or Groups	어떠한 개인 및 단체에 대한 차별 금지
6	No Discrimination Against Fields of Endeavor	SW 사용 분야에 대한 차별 금지
7	Distribution of Licence	라이선스는 재배포시에도 동일하게 적용
8	License Must Not Be a Specific to a Product	라이선스는 (유형의) 제품이 아니라 (무형의) SW에 적용
9	License Must Not Restrict Other Software	같이 배포되는 다른 소프트웨어 대한 제약 금지 (차별 금지)
10	License Must Be Technology-Neutral	라이선스는 기술에 중립적 (차별 금지)

공개SW의 장점은 공개된 소스코드를 활용할 수 있기 때문에 기술적으로 SW 종속성 탈피, 외부 기술의 내재화, SW 품질 향상, 개발기간 단축 같은 기술 혁신들이 가능하고 경제적으로도 공개SW 활용에 따른 개발 비용과 SW 라이선스 비용 절약 효과가 있다. 따라서 1990년대에는 운영체제, 데이터베이스, 웹 서버용 SW 같이 인터넷 서비스에 필요한 SW들이 주로 공개SW로 개발되었으나 현재는 인공지능, 클라우드, 빅데이터 같은 제4차 산업혁명을 이끌어가는 SW 신기술 분야에서도 널리 활용되고 있다.

특히 최근에는 국내에서 공개SW가 무료라는 잘못된 인식 덕분에 상업적 활용이 활성화가 되지 않는 것과 반대로 해외에서는 구글, 아마존, MS, 페이스북, IBM 등 다양한 기업들이 공개SW를 적극적으로 활용하고 있다¹¹⁾. 이는 OSI의 탄생 배경에서 언급하였듯이 강한 코드 공개 조항을 가진 GPL 이외에도 다양한 허가적 라이선스들이 오픈소스 라이선스로 공인받으면서 기업들도 공개SW 라이선스를 활용하는데 큰 어려움이 없어졌기 때문이다.

2. 공개소프트웨어의 상용화 동향

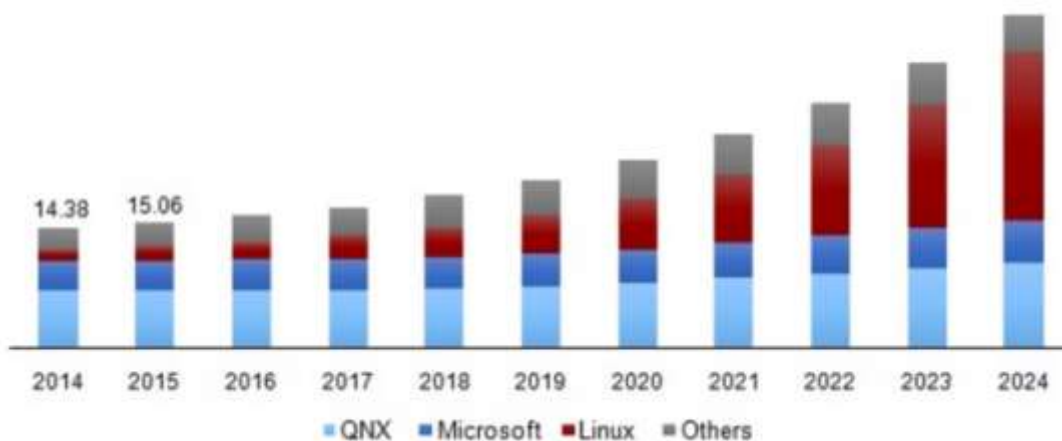
11) 권영환(2018), 공개 소프트웨어 산업의 이해 - 해외 기업들의 공개소프트웨어 활용, 소프트웨어 정책연구소

초기 공개SW 생태계가 구축되기 시작했던 시기에는 인터넷 서비스를 제공하기 위해 리눅스 계열 운영체제, MySQL 같은 데이터베이스, Apache HTTP 서버 같은 공개SW들을 활용하여 낮은 비용으로 인터넷 서버를 구축하는 용도로 주로 활용되었으며 이와 같은 초기 공개SW들은 지금도 널리 활용되고 있다.

GNU 프로젝트의 핵심 SW인 리눅스 커널은 1991년 공개된 이후 공개SW 활성화를 가능하게 한 대표적인 사례이다. 리눅스 커널은 RedHat, Suse 같은 초기 공개SW 기업을 탄생시킨 밑거름이었으며 해당 기업들은 리눅스 커널을 활용하여 사용자들이 쉽게 설치하고 사용할 수 있도록 리눅스 배포판을 개발하여 성장하였다. 그리고 2000년대에는 구글이 리눅스 커널을 기반으로 모바일 플랫폼의 핵심이 된 안드로이드의 핵심 모듈로써 활용하여 모바일 플랫폼 시장을 주도하고 있다. 리눅스 커널은 다양한 분야에서 활용되고 있으며 미래 자동차의 핵심 기능인 인포테인먼트 분야에서도 활용되어 관련 시장은 [그림 2-4]처럼 리눅스의 활용이 폭발적으로 증가할 것으로 예상되고 있다.

[그림 2-4] 차량용 인포테인먼트 시장에서의 리눅스 활용 예상

(연평균 성장률 : 32.9%)

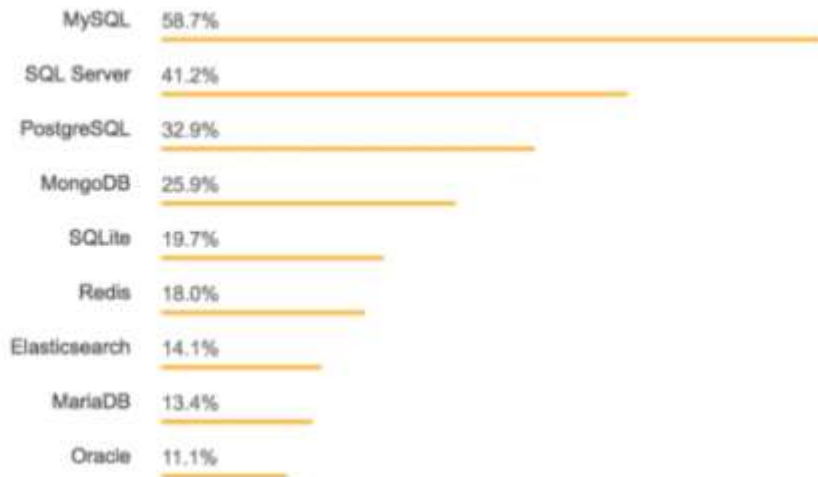


출처: Global automotive infotainment market worth \$40.17 billion by 2024, Autonomous Vehicle Technology (<https://www.autonomousvehicletech.com>), Aug 2017.

리눅스 커널과 함께 초기 공개SW의 성장을 이끌어왔던 데이터베이스 분야의 핵심 공개SW인 MySQL는 세계에서 인기있는 데이터베이스 시스템들 중 하나로 1995년 처음 공개되었다. 지금까지도 계속 신규 기능 추가 및 성능 개선을 통해 지속적으로 개발되고 있는 대표적인 공개SW로 이를 기반으로 하여 다양한 공개SW 기반 데이터베이스 프로젝트들이 파

생되었으며, 대표적인 마리아디비도 MySQL을 기반으로 파생된 공개SW이다.

[그림 2-5] 2018년 가장 인기있는 데이터베이스



출처: Most popular databases in 2018 according to StackOverflow survey, EverSQL (<https://www.eversql.com>), March 2018.

최근 SW 산업은 인공지능, 클라우드, 빅데이터 같은 SW 신기술들이 주도하고 있다. 이들 SW 신기술 분야에서 공개SW 개발 방식이 확산되고 있으며, 텐서플로우, 파이토치, 하둡, 오픈스택 등 다양한 공개SW 기반 프로젝트들이 SW 시장에서 널리 인정 받고 있다.

텐서플로우¹²⁾는 구글에서 2009년도부터 개발하였던 DistBelief의 후속 프로젝트로 구글 브레인팀의 주도로 개발한 머신러닝 오픈소스 SW 라이브러리이다. 2015년 아파치 라이선스로 공개되었으며 2018년 현재 머신러닝 분야에서 가장 인기있는 공개SW 프로젝트가 되었다. 텐서플로우를 활용하여 최신 머신러닝 기술을 CPU, GPU, TPU 등의 다양한 컴퓨팅 환경에서 이용할 수 있도록 개발되었다.

따라서 퀄컴, 엔비디아, 인텔, 우버, 에어비엔비, AMD 등 대표적인 IT 기업들도 텐서플로우를 활용하고 있다. 그리고 이를 활용하는 기업들이 지속적으로 늘어날 예상되고 있다. 텐서플로우는 깃허브에서 2017년 가장 많이 포크된 프로젝트로써 약 2.4만번 포크(Fork)되었으며 참여한 개발자수는 약 7300여명으로 5위를 차지하고 있다. 관련 프로젝트인 텐서플로우 모델(Tensorflow/Models)도 약 8천번 포크되어 포크 순위로 5위를 차지하고 있다¹³⁾

12) Tensorflow, <https://www.tensorflow.org>.

오픈스택¹⁴⁾은 2010년 미 나사(NASA)와 Rackspace에서 공동 추진한 클라우드 플랫폼 개발 프로젝트를 기반으로 하고 있다. 2012년 오픈스택 재단이 창립되어 아파치 라이선스를 활용하고 있으며 클라우드 플랫폼 구축에 필요한 서버, 스토리지, 네트워킹 자원을 가상화하기 위한 다양한 프로젝트들이 진행되고 있다. 오픈스택은 가상화 기술을 기반으로 클라우드 로드 분산을 위한 에지 컴퓨팅, 통신 분야의 네트워크 가상화, 금융 분야의 빅데이터 분석 및 고성능 컴퓨팅 자원이 필요한 과학 분야 등 다양한 분야에서 활용될 수 있다. 따라서 레드햇, 시스코, HP, 델, IBM, Huawei 등 500개 이상의 기업들이 오픈스택을 활용하고 있다.

R¹⁵⁾은 1992년부터 뉴질랜드 오클랜드 대학에서 개발되기 시작하여 2000년에 안정적인 빅데이터 분석이 가능해진 통계 프로그래밍 언어 겸 공개SW이다. 현재 R 재단에서 GPL 라이선스를 적용하여 관리하고 있으며 하둡과도 연동이 가능하며 MS 윈도우, 애플 맥, 리눅스, 유닉스 등 다양한 환경에서 동작 가능하기에 빅데이터 분석을 위해 많이 활용되고 있다. 특히 데이터 입출력, 조작 처리, 계산뿐만 아니라 시각화 작업이 가능한 통계 SW로써 투표 데이터, 마이닝 데이터 (mining data), 교육 및 연구용 데이터 등 다양한 유형의 빅데이터를 효율적으로 처리가 가능하다.

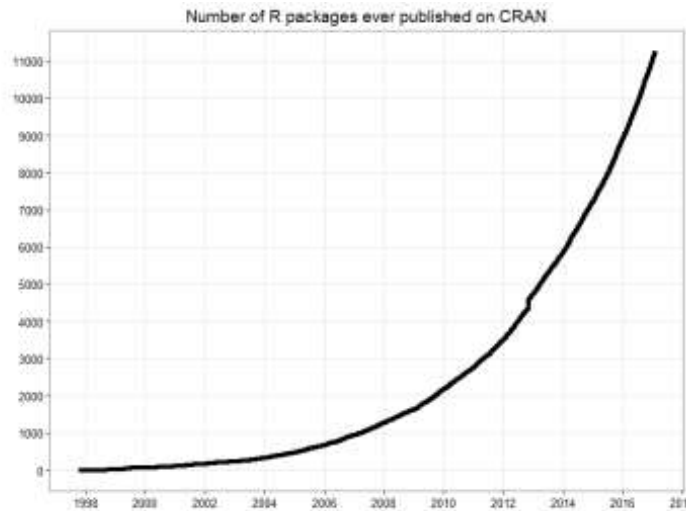
따라서 구글과 페이스북에서는 주요 분석 플랫폼으로 활용하고 있으며 오라클, SAP, IBM에서도 주요 솔루션에 적용하고 있다. 현재 약 200만명 이상의 사용자를 가지고 있는 R은 사용자들이 주도적으로 R 컨소시엄을 조직하며 전세계 사용자, 관리자, 개발자를 지원하고 있다. 또한 다양한 버전의 R 관련 코드, 문서를 다운로드 가능한 CRAN(Comprehensive R Archive Network)의 R 패키지 수는 아래 그림처럼 급속히 늘어나고 있다.

13) The State of Octoverse 2017, <https://octoverse.github.com>.

14) Openstack, <https://www.openstack.org>.

15) The R Project for Statistical Computing, <https://www.r-project.org/>

[그림 2-6] CRAN에 공개된 R 패키지의 수



출처: CRAN now has 10,000 R packages. Here how to find the ones you need, Revolutions Blog (<http://blog.revolutionanalytics.com/>), January 2017.

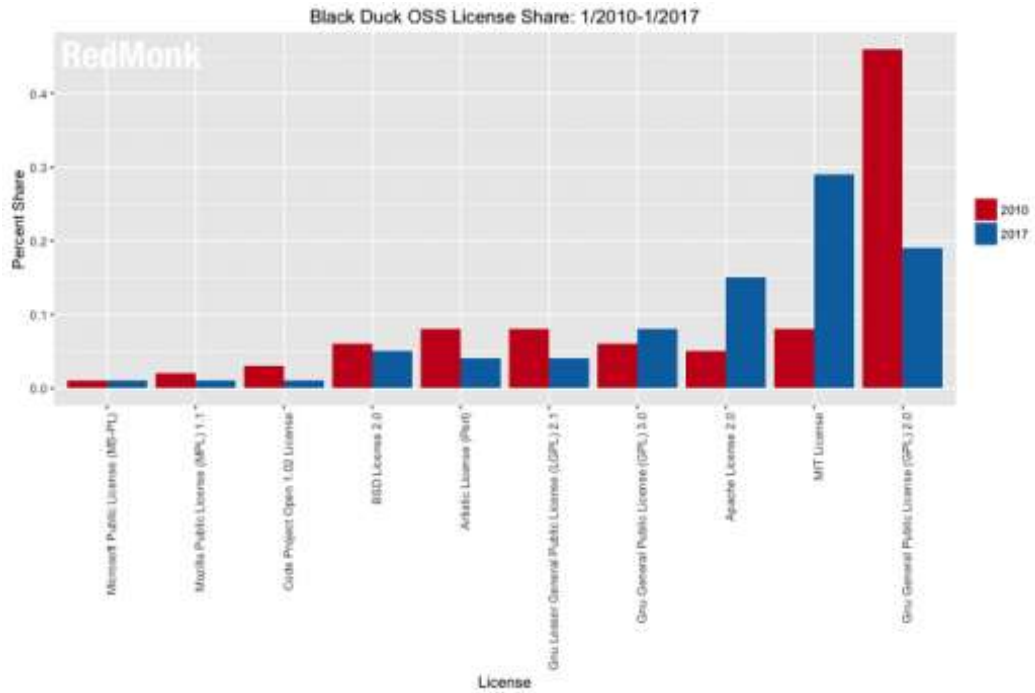
이와 같이 다양한 공개SW들이 출현되고 이를 활용하는 기업들도 꾸준히 증가하고 있다. 초기에는 리눅스 커널과 MySQL 등을 활용하여 일부 기업들이 상업화에 성공하였다. 현재는 인공지능, 빅데이터, 클라우드 등 다양한 분야에서 공개SW가 활용되고 있으며 다양한 공개 SW들을 활용하는 기업들이 다양해지고 있다¹⁶⁾.

OSI가 설립되고 약 20년이 지난 현재 오픈소스 라이선스 활용에 변화가 생기고 있다. 블랙덕에서 발표한 자료에 의하면 2010년대에 들어서 자유SW의 대표 라이선스이면서 오픈소스의 핵심 라이선스이었던 GPL의 사용이 감소하고 MIT와 아파치 라이선스의 활용이 급격히 증가하고 있다. 2017년을 기준으로 MIT 라이선스 비중은 약 30%에 육박할 정도로 약 3배 가량 늘어났으며 아파치 라이선스도 약 20%에 근접할 정도로 2배 이상 늘었다. 반면에 GPL 2.0의 경우 45% 넘던 점유율에서 20% 미만으로 절반 이상 감소했다¹⁷⁾. 이러한 현상은 OSI에 탄생하면서 성장한 오픈소스의 확산에 따른 결과라고 할 수 있다.

16) Jake Flomenberg, The Rise of OPEN Innovation: The 3P's for Building a Durable OPEN Software Company, Revolutions Blog (<http://blog.revolutionanalytics.com/>), February 2016.

17) Jono Bacon, The decline of GPL?, opensource.com(<https://opensource.com>), February 2017.

[그림 2-7] 공개SW 라이선스 활용의 변화 (2010년과 2017년 비교)



출처: Jono Bacon, The decline of GPL?, opensource.com(<https://opensource.com>), February 2017.

제3장 해외 공개소프트웨어 활용 동향

공개SW는 컴퓨터 운영체제, 데이터베이스, 웹 서버용 SW로 시작하여 현재 인공지능, 빅데이터, 클라우드 분야로 확산되었다. 제4차 산업혁명으로 인하여 SW 신기술들은 안전과 밀접한 자동차 분야 등 비SW 산업 분야에서도 활용되기 시작했다. 이에 반하여 국내에서는 SW 신기술 분야의 공개SW 활용 및 비SW 산업 분야에서의 공개SW 활용에 대한 현황 자료가 부족한 현실이다. 따라서 본 보고서에서는 SW안전 분야의 공개SW 현황을 파악하기 위해 SW 신기술 분야와 SW안전 주요 산업 분야에 대해 인터넷 검색을 통한 문헌 조사를 통한 공개SW 동향을 소개한다.

제1절 소프트웨어 신기술 분야

공개SW가 주목을 받는 것은 제4차 산업혁명의 핵심 기술이라고 할 수 있는 인공지능, 빅데이터, 클라우드 분야에서 신기술 개발을 주도하고 있기 때문이다. SW 신기술 분야의 최신 기술들의 대부분이 공개SW 개발 방식으로 만들어지고 있으며, 구글, 아마존, MS, 페이스북 등 유명 IT 기업들에서 활용되고 있다. 따라서 공개SW가 주도하는 SW 신기술 분야에서 공개SW가 어떠한 위치에 있는지 알아보기 위해 우선적으로 인공지능, 빅데이터, 클라우드 분야의 대표적인 공개SW들의 동향을 살펴보도록 한다.

1. 인공지능 분야

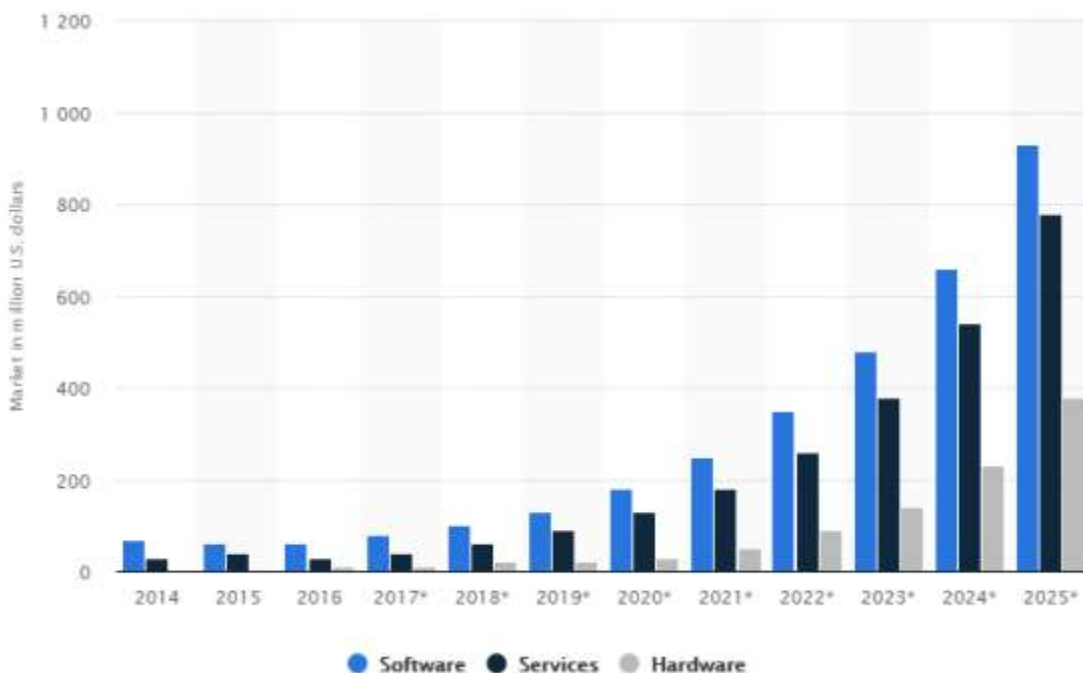
인공지능이란 용어는 1956년 다트머스 회의에서 ‘생각하는 기계’에 관련된 논의 중에 처음으로 만들어졌다. 이에 대한 정의는 연구 분야별 또는 학자별로 서로 다른 의견을 가지고 있기 때문에 명확하지 않다. 하지만 인공지능 기술의 발전으로 인하여 현재는 인공지능이 여러 분야에서 활용되고 있으며 최근에는 제4차 산업혁명의 핵심 기술로 부각되고 있다. 다양한 SW 기술들과 함께 음성인식, 이미지인식, 영상인식 분야에서 두각을 나타내고 있으며 이를 산업적으로 적용하여 자율주행차, 로봇, 스마트 가전 등 새로운 제품들을 만들어

내고 있다.

인공지능은 인간의 지능을 모방하기 위해 나온 기술로 인간 같은 감정 등을 지닌 강 인공지능과 인식, 추론 능력 등을 가진 약 인공지능으로 구분되고 있다. 인공지능은 크게 3가지로 단계를 거쳐 구현된다. 학습, 추론, 인식의 3단계를 걸쳐서 이루어진다. 이 과정을 효율적으로 처리하기 위해서 인공지능 연구자들은 인간의 뇌 구조를 단순화하여 효율적인 학습 및 추론이 가능하도록 인공신경망 구조를 구현하였다. 인공신경망 구조는 복잡한 정보와 다양한 변수를 동시에 고려한 여러 단계의 추론 과정이 가능하도록 설계되었다. 그 결과 인공지능 분야의 성능이 크게 향상되었고 2016년 알파고와 이세돌 9단의 바둑 대결에서 알파고가 4:1로 승리하게 되었다.

그 이후 인공지능에 대한 폭발적인 관심으로 시장이 급성장하게 되었고 인공지능의 핵심 분야인 심층학습 분야의 시장이 [그림 3-1]처럼 SW 분야, 서비스 분야, HW 분야 모두 성장할 것으로 예상되고 있으며 2018년 대비 2025년까지 약 10배 가량 성장할 것으로 예측된다.

[그림 3-1] 미국 심층학습 시장 전망(2014년부터 2025년까지)



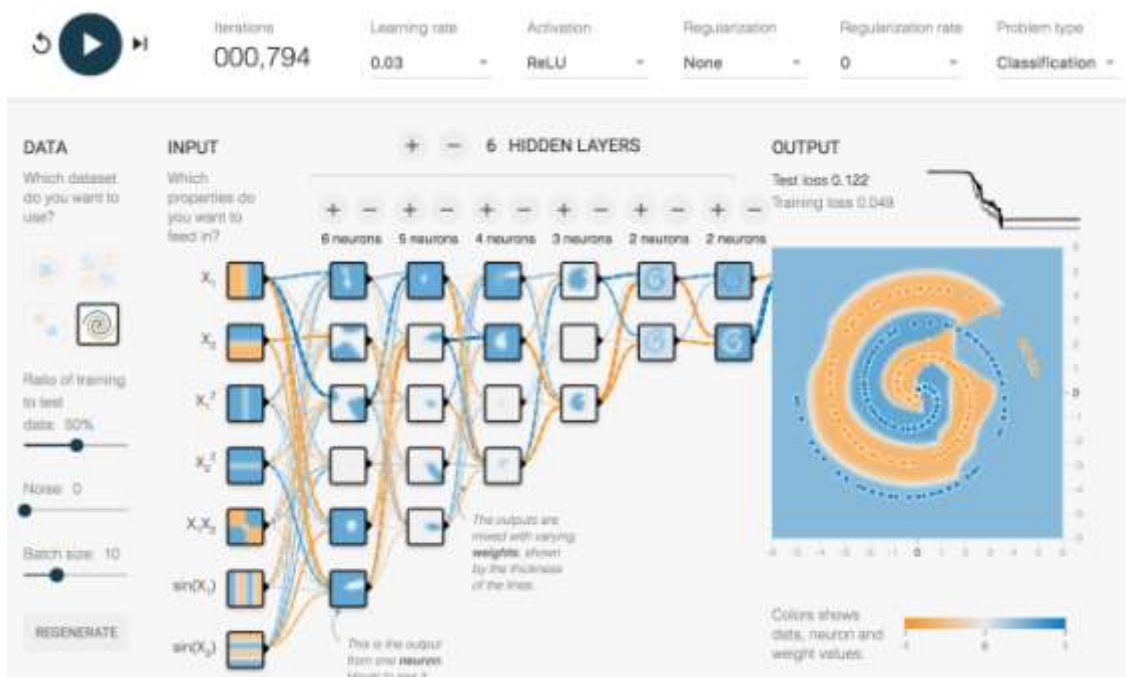
출처: statista(2018), Deep learning market revenues in the United States, from 2014 to 2025, by segment (in million U.S. dollars)

이미 구글 등 글로벌 IT 기업들은 인공지능 관련 소스코드, 라이브러리, 학습 데이터 및

시험 환경 등 개발 플랫폼을 공개SW 형태로 공개하고 있다. 글로벌 IT 기업들은 인공지능 혁신 생태계를 주도하기 위해 많은 투자를 통해 개발한 인공지능 플랫폼을 무료로 공개하여 관련 협력사와 개발자를 확보하기 위해서 이다. 그 결과 인공지능 기술을 활용하여 다양한 산업 및 사회 분야에서 IT 기업들이 공개한 인공지능 개발 툴을 중심으로 많은 스타트업들 및 관련 연구자들을 포함하여 인공지능 기술 발전이 이루어지고 있다.

텐서플로우(Tensorflow)¹⁸⁾는 구글에서 개발한 머신러닝 라이브러리로 2015년 11월에 공개 SW로 공개하였다. 텐서플로우는 머신러닝 및 신경망 학습 연구를 위해 개발되어 DistBelief보다 유연하고 확장성이 향상되었다. 또한, 학습 과정에서 결과를 직관적인 그래프로 표현할 수 있다.

[그림 3-2] 텐서플로우를 활용한 인공 신경망 모델링



출처: 텐서플로우 블로그, <https://tensorflow.blog/category/tensorflow/>

텐서플로우는 2018년 8월까지, 949명의 기여자들에 의해 39,383번의 커밋이 있었으며, 코드 라인 수는 1,821,016 라인에 이른다. 대부분 C++에 의해 작성되었다. 저장소로 깃허브¹⁹⁾

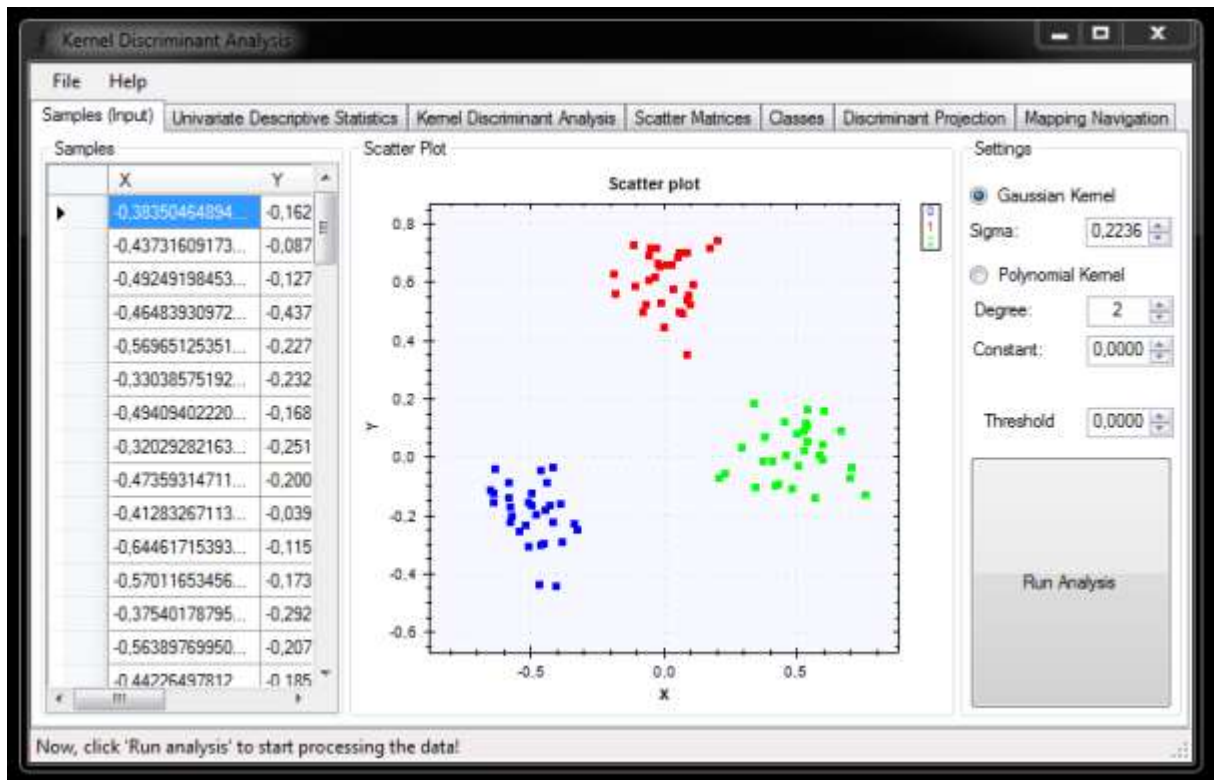
18) <https://www.tensorflow.org/>

19) <https://github.com/tensorflow>

를 사용하고 있다. COCOMO²⁰⁾ 모델에 의하면 대략 520년의 노력이 들어간 것으로 추정되며 아파치 2.0 라이선스를 따르고 있다.

Accord.NET²¹⁾은 MS에서 개발한 닷넷(.NET) 기반 머신러닝 프레임워크로 2010년에 공개되었으며 인공지능 및 컴퓨터 비전을 위해 개발된 AForge.NET를 기반으로 확장되었다. 이미지, 신호, 오디오 데이터 처리를 위한 라이브러리를 포함하고 있으며 윈도우 애플리케이션 개발을 위한 닷넷 플랫폼에서 사용 가능하다. Accord.NET 프로젝트는 2018년 8월까지 74명의 기여자들에 의해 2,436번의 커밋이 있었으며, 코드 라인 수는 2,308,485 라인에 이른다. 대부분 C#에 의해 작성되었으며 저장소는 깃허브²²⁾를 사용하고 있다. COCOMO 모델에 의하면 대략 658년의 노력이 들어간 것으로 추정되며 LGPL(GNU Lesser General Public License)을 따르고 있다.

[그림 3-3] Accord.Net을 활용한 데이터 분류



출처: Accord.Net Machine Learning Framework(<http://accord-framework.net/>)

20) COCOMO (Constructive Cost Model)의 정의 - 시스템의 비용을 산정하기 위해 시스템을 구성하고 있는 모듈과 서브 시스템의 비용 합계를 계산하는 방식 - 소프트웨어 유형을 모듈별(Organic, Semi-Detached, Embedded)로 구분하고 각 유형에 대한 총 인월(man/month)수와 개발 기간을 계산하는 방식 - 소프트웨어 개발비, 유지보수 비용의 건적, 개발 단계 및 업무 활동과 산출물을 정의하여 포함한 비용 산정

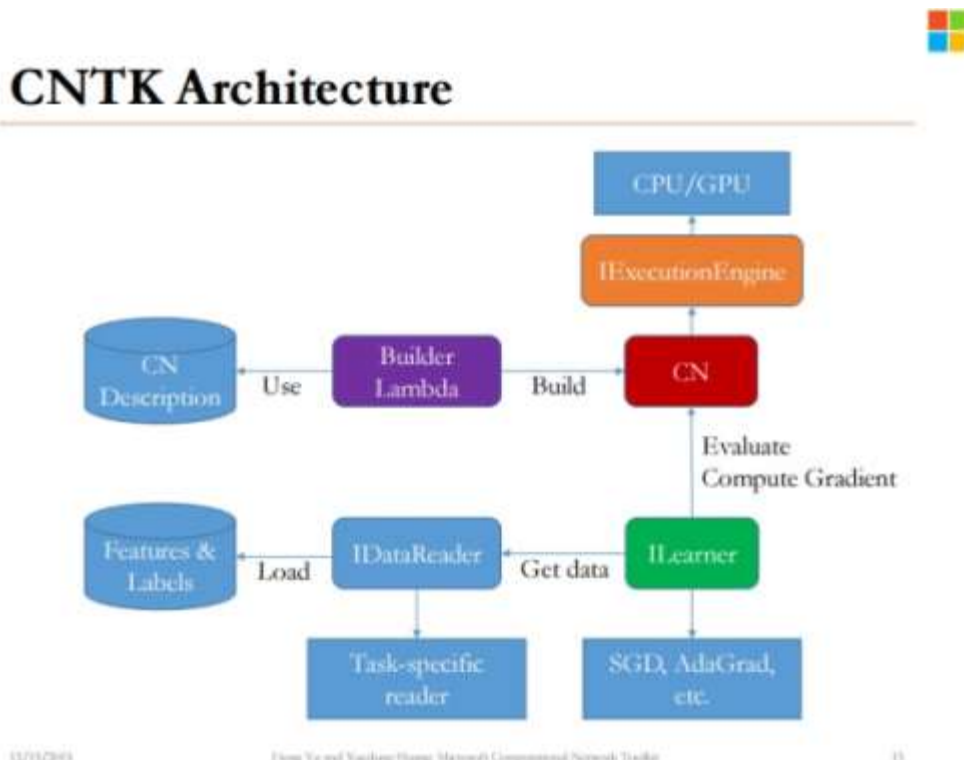
21) <https://www.openhub.net/p/Accord-NET>

22) <https://github.com/accord-net/framework>

Accord.NET은 음성, 얼굴, 문자, 행동 인식 및 감정 분석 연구에서 주로 사용되고 있으며 응용프로그램으로 해변 박테리아 농도 예측을 위한 Virtual Beach 3, 스마트폰 카메라로 일반 문서 또는 명함의 전화번호를 식별하는 Point and Call 및 사람의 동공 식별 기능을 탑재한 PhotoCam 등이 있다.

CNTK(Cognitive Toolkit)²³⁾은 MS에서 개발된 심층학습 프레임워크로 2016년에 공개하였다. 초기에는 BrainScript라는 이름의 스크립트 언어만 사용이 가능했으나 2016년 말에 파이썬 API를 제공하고 있다. 파이썬 라이브러리로서 텐서플로우와 그 기능이 유사하다고 할 수 있다. 2015년에 이미지 인식 세계 대회(ILSVRC)와 MS의 COCO 챌린지에서 1위를 차지한 바 있다. 2018년 8월까지 269명의 기여자들에 의해 15,931번의 커밋이 있었으며, 코드 라인 수는 324,472 라인에 이른다. 대부분 C++에 의해 작성되었다. 저장소는 깃허브²⁴⁾를 사용하고 있다. COCOMO 모델에 의하면 대략 85년의 노력이 들어간 것으로 추정되며 MIT 라이선스를 따르고 있다.

[그림 3-4] MS CNTK 구조



출처: Bloter, MS, 오픈소스 딥러닝 툴킷 깃허브에 공개

23) <https://www.microsoft.com/en-us/cognitive-toolkit/>

24) <https://github.com/Microsoft/CNTK>

CNTK의 주요 활용 사례로는 번역, 음성 인식, 음식 이미지 식별 등 MS의 음성비서 SW인 코타나(Cortana)와 스카이프 Translator 같은 응용 프로그램에 적용된 것으로 알려져 있으며 소리 데이터 분석을 통한 오일 및 수도관 파손 및 누출 탐지 시스템 분야에서도 활용되는 것으로 알려져 있다.

아파치 재단의 Mahout²⁵⁾은 루씬²⁶⁾ 프로젝트의 하위 프로젝트로써 머신러닝 라이브러리 구현을 목표로 개발되어, 2010년 아파치 공식 프로젝트 단계인 탑-레벨로 전환되었다. 확장성 제공을 위해 대용량 데이터 처리 시스템인 하둡 파일시스템을 사용하였으며 최근 성능과 사용자 편의성이 향상된 스파크 시스템 등에 활용되고 있다. Mahout은 2018년 8월까지 59명의 기여자들에 의해 3,969번의 커밋이 있었으며, 코드 라인 수는 144,123 라인에 이르며 대부분 자바로 작성되었다. 저장소는 깃허브²⁷⁾를 사용하고 있으며 COCOMO 모델에 의하면 대략 37년의 노력이 들어간 것으로 추정되면 아파치 라이선스를 따르고 있다.

[그림 3-5] 아파치 Mahout의 알고리즘

Algorithms in Mahout



See <http://cwiki.apache.org/confluence/display/MAHOUT/Algorithms>

출처: BigData Academy, Hadoop Training in Mumbai

25) <https://www.openhub.net/p/mahout>

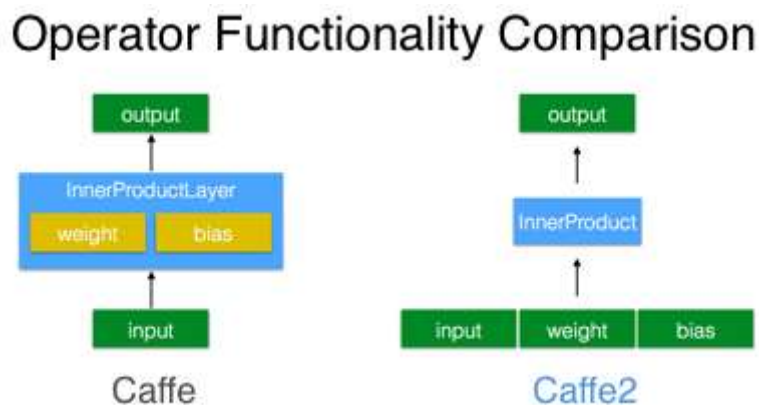
26) <http://lucene.apache.org/>

27) <https://github.com/apache/mahout>

Mahout의 주요 활용 사례로는 Adobe, Twitter, Foursquare, Mendeley, Drupal에서 개인화 서비스 제공을 위한 추천시스템 등에 활용 되고 있으며 NewsCred는 뉴스 기사 클러스터링 및 주요 내용 추출에 활용되고 있다.

Caffe²⁸⁾는 원래 UC 버클리에서 개발된 오픈소스 기반 심층학습 프레임워크로 이미지 분류 등의 용도로 사용되었다. 2017년 페이스북 인공지능 연구그룹에서 Recurrent 신경망 기능을 추가함으로써 Caffe2²⁹⁾를 추가로 공개하였다. Caffe2는 기존 Caffe를 기반으로 엔비디아, 퀄컴, 인텔, MS 등 여러 기업들과 공동 연구 개발하였으며 학습 모델의 성능, 이식성, 활용성 등을 향상 시키는데 중점을 두었다. Caffe 2 프로젝트는 2018년 8월까지 332명의 기여자들에 의해 3,678번의 커밋이 있었으며, 코드 라인 수는 275,560 라인에 이르며 대부분 C++에 의해 작성되었고 파이썬 인터페이스를 지원하고 있다. 저장소는 깃허브³⁰⁾를 사용하고 있으며 COCOMO 모델에 의하면 대략 73년의 노력이 들어간 것으로 추정되며 BSD 라이선스를 따르고 있다.

[그림 3-6] Caffe와 Caffe2의 동작 차이점



출처: Caffe2(<https://caffe2.ai/>)

28) Caffe는 2013년 미(美) UC 버클리 AI 연구팀(Berkeley Vision and Learning Center, BVLC)이 사용자의 사용성, 데이터 처리 성능을 고려하여 개발한 딥러닝 플랫폼으로 이미지 분류, 인식에 특화

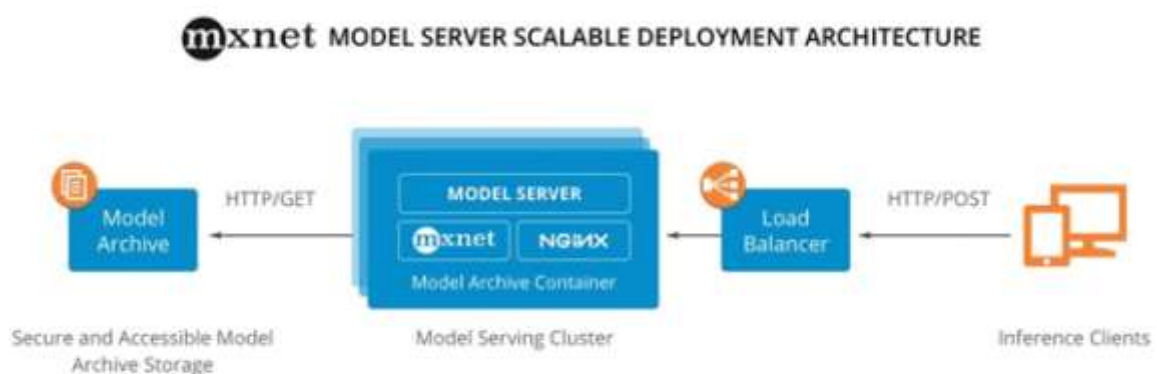
29) <https://www.openhub.net/p/caffe2>

30) <https://github.com/caffe2/caffe2>

Caffe 2의 주요 활용 사례로는 데스크탑, 서버 등에서 사용될 수 있는 리눅스, 윈도우, iOS, 안드로이드 등 다양한 플랫폼을 지원하면서 GPU 활용을 통해 성능을 극대화시킬 수 있다. 야후에서는 아파치 스팍(Apark)과 연동하여 CaffeOnSpark라는 분산 심층학습 프레임워크를 새로이 만들었다. 그리고 2018년에 페이스북의 주도로 파이토치(PyTorch)에 흡수되었다.

MXNet³¹⁾은 아마존의 AWS, MS의 애저(Azure) 등 유명 클라우드 서비스에서 서비스되고 있는 아파치 재단의 심층학습 프레임워크이다. 카네기멜론, 스탠포드, MIT 등의 학계와 MS, 인텔, 바이두, 엔디비아 같은 기업들이 공동으로 개발하고 있다. 학습 과정에서 메모리 사용의 효율성, 시스템 확장성, 사용자 편의성, 유연성, 모바일 지원을 통한 이식성(Portable) 등에 초점을 두었으며 다양한 프로그래밍 언어(파이썬, 스칼라, 매트랩, C++, Perl, R, Go, Julia 등)에서 비슷한 성능으로 활용 가능하다. MXNet은³²⁾ 2018년 8월까지, 682명의 기여자들에 의해 7,778번의 커밋이 있었으며, 코드 라인 수는 305,755 라인에 이르며 대부분 C++에 의해 작성되었다. 저장소는 깃허브³³⁾를 사용하고 있으며 COCOMO 모델에 의하면 대략 31년의 노력이 들어간 것으로 추정된다. 아파치 재단의 프로젝트써 아파치 라이선스를 따르고 있다.

[그림 3-7] MXNet의 확장 가능한 구조



출처: 아마존 AWS

31) <http://mxnet.incubator.apache.org/>

32) <https://www.openhub.net/p/mxnet>

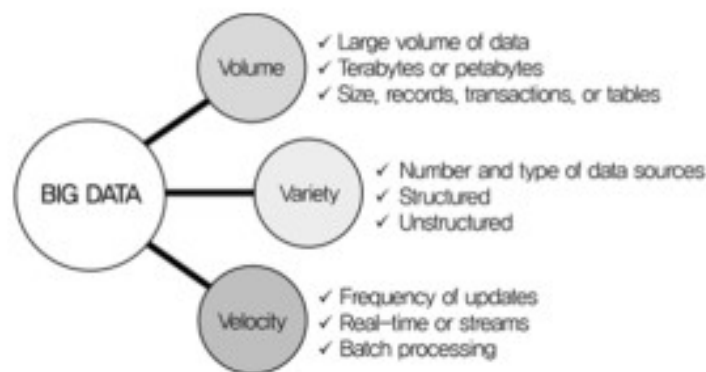
33) <https://github.com/apache/incubator-mxnet>

MXNet의 주요 활용 사례로는 2차원 이미지를 3차원으로 변환하는 연구, 시간에 따른 사용자와 영화 사이의 관계 예측 연구 및 악보의 기호 및 텍스트 분석 연구 등에 널리 활용되고 있다. 특히 AWS 및 애저 같은 클라우드 서비스에서 이미 활용되고 있다.

2. 빅데이터 분야

빅데이터는 단순히 큰 데이터를 의미하는 것이 아니라 [그림 3-8]처럼 데이터 양이 많고 빠르게 변화하며 속성이 다양한 데이터를 의미한다. 보통 비정형 데이터를 중심으로 효과적으로 데이터를 처리하는 분야와 정형 데이터베이스에서 효율적으로 저장하기 위한 분야로 구분 가능하다.

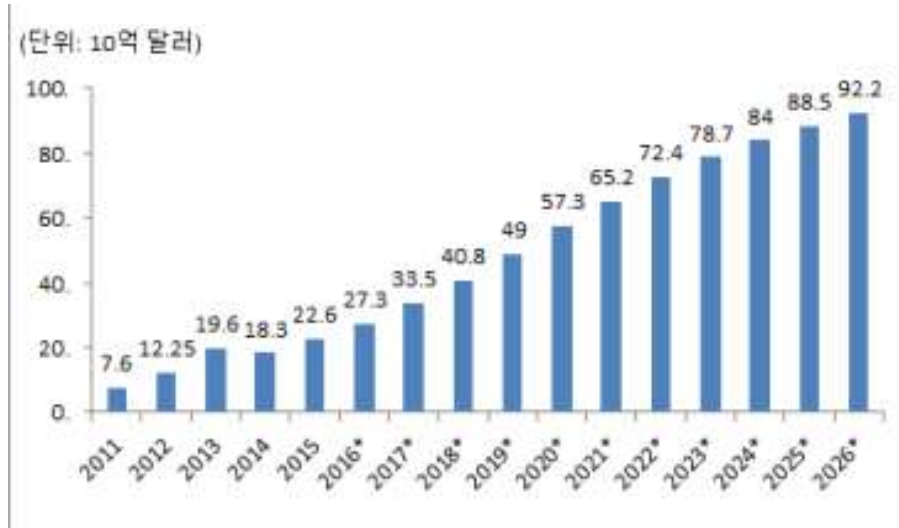
[그림 3-8] 빅데이터의 특징



출처 : TDWI Research, “Big Data Analytics Report,” , 2011

제4차 산업혁명에 의하여 초연결 사회로 진화하는 시대에서 빅데이터 시장은 폭발적으로 늘어날 것으로 예상되며, 빅데이터 분석을 통해 정제된 데이터는 인공지능 기술에 의해 자동화, 자율화 기능을 제공하는데 큰 역할을 할 것으로 예측된다. 따라서 소프트웨어, 하드웨어, 서비스를 모두 포함한 세계 빅데이터 시장은 앞으로 2026년에 총 922억 달러의 규모로 성장할 것으로 전망된다. 이는 2014년 기록했던 183억 달러에서 약 404% 증가한 수치이며, 2014년부터 2026년까지의 연평균 성장률은 14.4%에 달한다.

[그림 3-9] 전세계 빅데이터 시장 전망



출처 : Wikibon Big Data in the Public Cloud Forecast, 2016-2026 (Wikibon)

빅데이터 플랫폼은 빅데이터에서 가치를 추출하기 위한 일련의 과정을 지원하기 위한 프로세스를 제공해 준다. <표 3-1>처럼 데이터 수집 → 저장 → 처리 → 분석 → 표현 등의 단계를 통해 원시데이터에서 의미와 가치를 추출하며 큰 틀에서 데이터 수집, 저장, 처리 영역에서 전통적인 통계적 처리와 차별화가 된다. 분석, 시각화 측면에서는 전통적인 통계 분석의 인과관계³⁴⁾뿐만 아니라 연관관계, 상관관계³⁵⁾에 대한 분석도 포함된다.

<표 3-1> 빅데이터 플랫폼에서 가치 추출을 위한 단계들

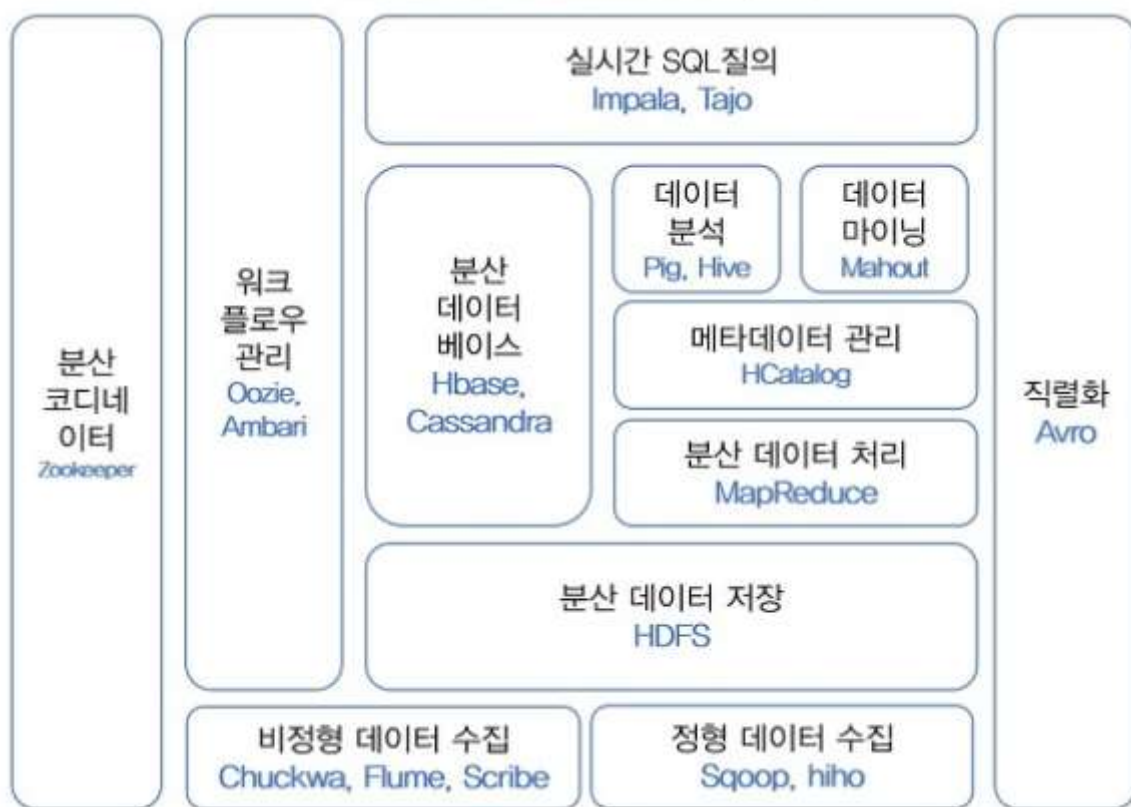
수집	저장	처리	분석	표현
비정형 데이터 정형 데이터 ETL ³⁶⁾ Web Robot 등	원본 데이터 NoSQL 메모리 분산 파일 시스템 데이터보안	배치처리 분산 병렬 처리 실시간처리 인-메모리 처리 등	텍스트 분석 머신 러닝 통계 데이터 마이닝 SNS 분석 등	시각화 (수치, 키워드, 그래프 구조 등)

34) 인과관계(Causality) : causation 혹은 cause and effect라고도 함. 선행하는 한 변인(cause)이 후행하는 다른 변인(effect)의 원인에 책임이 있으며, 후행 변인은 선행 변인에 부분적으로 의존하는 현상

35) 상관관계(Correlation) : 한 통계적 변인이 다른 여러 통계적 변인들과 공변하는 관계. 회귀분석, 상관계수 등을 활용하여 측정

대표적인 빅데이터 공개SW로 하둡(Hadoop:High-Availability Distributed Object-Oriented Platform) 프로젝트가 있다. 하둡은 자바 기반 빅데이터 프레임워크로서 대용량의 데이터를 여러 개의 저장소에서 분산 처리를 지원하기 위해 개발되었으며, 인공지능 공개SW인 Mahout과 마찬가지로 아파치 루씬 하부 프로젝트 중 하나로 하둡은 [그림 3-10]과 <표 3-2> 같은 다양한 서브 프로젝트들과 연동을 통해 강력한 빅데이터 프레임워크 기능을 제공하고 있다. 하둡³⁷⁾은 2018년 8월까지 247명의 기여자들에 의해 18,931번의 커밋이 있었으며, 코드 라인 수는 2,918,489 라인에 이르며 대부분 Java로 작성되었다. 저장소는 깃허브³⁸⁾를 사용하고 있으며 COCOMO 모델에 의하면 대략 842년의 노력이 들어간 것으로 추정되며 아파치 라이선스를 따르고 있다.

[그림 3-10] 하둡 에코시스템



출처: 하둡 에코시스템(Hadoop-Ecosystem)이란?³⁹⁾

36) Extraction, Transformation, Loading

37) <https://www.openhub.net/p/Hadoop>

38) <https://github.com/apache/hadoop>

39) <http://over153cm.tistory.com/entry/하둡-에코시스템HadoopEcosystem이란>

〈표 3-2〉 하둡의 기능을 보완하는 빅데이터 관련 공개SW들

빅데이터 구분	주요 기술	기술별 주요 기능
스트리밍 데이터 수집	Flume	- 비정형 데이터 수집 - 클라우드데라에서 개발됐으며, 현재 아파치 인큐베이션에 포함됨
	Scribe	- 비정형 데이터 수집 플랫폼 - 중앙 집중 서버로 전송하는 방식으로, 페이스북에서 개발
	Chuckwa	- 비정형 데이터 수집 플랫폼으로 HDFS에 분산 데이터를 저장
정형 데이터 수집	Sqoop	- 정형 데이터 수집 - 관계형 DB로부터 데이터 가져오기 - HDFS, NoSQL 등 다양한 저장소에의 전송 지원
	Hiho	- 대용량 정형 데이터 수집 및 전송 솔루션
분산 데이터 베이스	Hbase	- 분산 데이터베이스 - HDFS기반의 컬럼 기반 NoSQL 데이터베이스, 구글의 BigTable 논문을 기반으로 개발됨 - 야후, 트위터 등이 사용하며, 국내 NHN도 라인에 적용
	Cassandra	- 공개SW 분산 데이터베이스 관리 시스템 - 컬럼 중심 DB와 행 중심 DB의 복합형 - NoSQL의 하나
실시간 SQL 질의	Impala	- 하둡 기반의 실시간 SQL 질의 시스템 - 클라우드데라에서 개발 - 맵리듀스로 처리하지 않고, 자체 개발한 엔진 사용
	Tajo	- 국내 대학 학생들이 주도해 개발한 하둡 기반의 DW 시스템 - 2013년 아파치 재단의 인큐베이션 프로젝트로 선정
메타 데이터 관리	HCatalog	- 빅데이터 메타 정보 관리
데이터 분석	Hive	- 하둡 기반의 데이터웨어하우징용 솔루션 - 유사 SQL 기반 빅데이터 처리 - 페이스북에서 개발하여, 오픈 소스로 공개
	Pig	- 데이터 분석 - MapReduce 대신 자체 언어 Pig Latin 제공
인메모리 처리	Spark	- 공개SW 클러스터 컴퓨팅 프레임워크 - UC 버클리 AMPLab에서 처음 개발함
데이터 마이닝	Mahout	- 데이터 마이닝 - 하둡 기반의 오픈 소스
워크 플로우 관리	Oozie	- 빅데이터 처리 과정 관리

		- 하둡 작업을 관리
분산 코디네이터	Zookeeper	- 빅데이터 서버 시스템 관리 - 분산 환경 서버들간의 상호 조정 서비스
직렬화	Avro	- RPC(Remote Procedure Call)과 데이터 직렬화를 지원하는 프레임워크
리소스 매니저	YARN	- 리소스 관리 플랫폼 - 분산 컴퓨팅 환경 제공 - 클러스터내 컴퓨팅 자원 관리 - 사용자의 애플리케이션 스케줄링 사용 관리

출처: Hadoop 이해⁴⁰⁾와 “하둡 에코시스템(Hadoop-Ecosystem)이란?⁴¹⁾ “내용 재구성

하둡(Hadoop)의 대표적인 활용 사례로는 개인용 정보 관리 및 업무 생산성 애플리케이션인 에버노트⁴²⁾는 막대한 양의 데이터로 인해 시스템이 포화 상태를 해결하고자 하둡(Hadoop)과 대규모 병렬 처리 분석 데이터베이스인 파엑셀(ParAccel)을 결합한 하이브리드로 전환하였다. 그리고 하둡을 사용해 계층의 데이터를 분석하는 연구도 진행되고 있으며 국내에서는 삼성SDS가 바이오 인포매틱스 서비스를 위해 하둡과 NoSQL를 사용하고 있는 것으로 알려져 있다.

R⁴²⁾은 원래 Bell 연구소에서 만든 통계처리 프로그래밍 언어인 ‘S’를 참고하여 로버트 젠틀맨(Robert Gentleman)과 로스 이하카(Ross Ihaka)가 누구나 자유롭게 사용할 수 있도록 공개SW로 구현한 것이다. R은 다양한 통계 기법과 수치 해석 기법을 지원하고 있으며 사용자가 제작한 패키지를 추가하여 기능을 확장할 수 있다. 핵심적인 패키지는 R과 함께 설치되며, CRAN(Comprehensive R Archive Network)을 통해서 추가로 다운 받을 수도 있다. R⁴³⁾은 2018년 8월까지, 33명의 기여자들에 의해 51,782번의 커밋이 있었으며, 코드 라인 수는 782,321 라인에 이르며 대부분 C에 의해 작성되었다. COCOMO 모델에 의할 때 대략 215년의 노력이 들어간 것으로 추정되며 GPL(GNU General Public License) 2.0 라이선스를 따르고 있다.

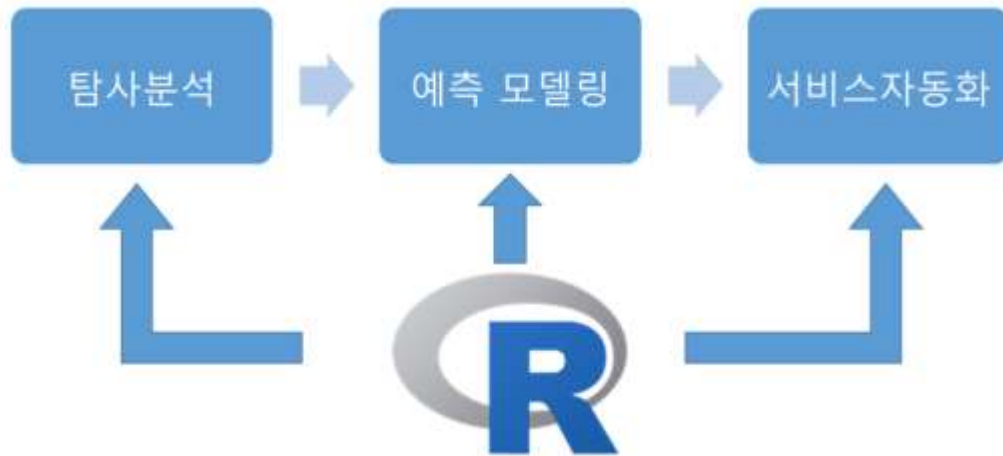
40) <https://iamdaisy.tistory.com/11>

41) <http://over153cm.tistory.com/entry/하둡-에코시스템HadoopEcosystem이란>

42) The R Project for Statistical Computing, <https://www.r-project.org/>

43) https://www.openhub.net/p/r_project

[그림 3-11] R의 활용 용도



출처: NC Tech Dev

R은 윈도, 맥 OS 및 리눅스를 포함한 UNIX 플랫폼에서 이용 가능하며 통계 계산과 소프트웨어 개발이 필요한 통계학자와 연구자들에게 좋은 환경을 제공해준다. R의 장점은 강력한 데이터 시각화 기능을 제공해주고 있기 때문에 원 데이터 분석 결과를 효과적으로 전달할 수 있으며 활발한 생태계를 통해 최신 기술을 활용하기 위한 패키지에 빠르게 접근할 수 있고 많은 통계학자들이 개발 과정에 참여함으로써 통계의 개념들이 R의 코드와 패키지에 적용되어 있다. 단점으로는 효율적인 컴퓨터 자원 활용이 부족해서 데이터 처리 속도가 느리고 통계에 대한 전문 지식이 없이는 접근하기 어렵다는 점이 있다.

R은 구글, 페이스북, 아마존, 트위터 등 많은 데이터를 분석하기 위한 기업들에서 사용되고 있으며 특히 MS에서는 R과 호환되는 Microsoft R Open이라는 윈도우 환경에서 보다 효율적으로 동작하는 개선 버전의 배포판을 제공하고 있으며 2015년에는 R 상용 서비스를 제공하는 Revolution Analytics을 인수하였다. 오라클은 2012년부터 오라클 R 엔터프라이즈라는 R 상용 버전을 제공하고 있으며 IBM도 하둡에서 R을 지원하기 위한 기능들을 제공하고 있다. 이 뿐만 아니라 데이터 처리 및 통계 분야의 Mathematica, MATLAB, SPSS, Statistica, SAS 등도 R을 지원하기 하려고 하고 있다⁴⁴⁾.

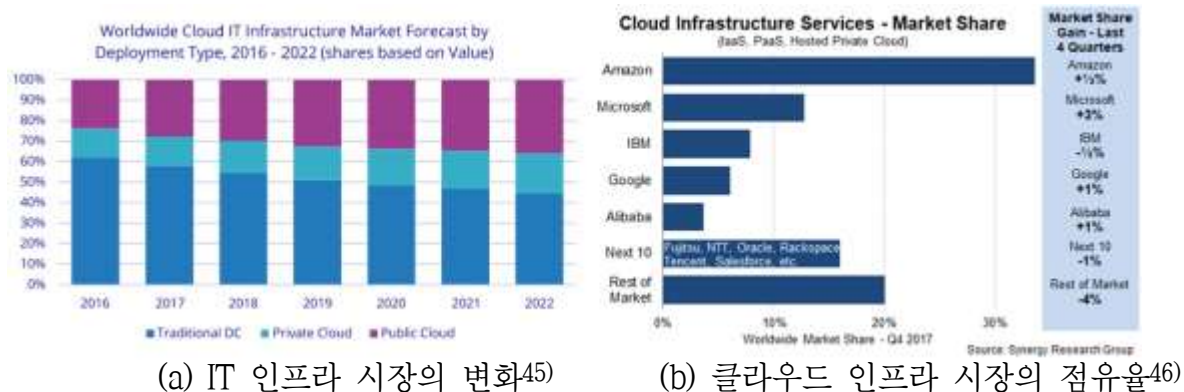
3. 클라우드 분야

44) 위키피디아, [https://en.wikipedia.org/wiki/R_\(programming_language\)](https://en.wikipedia.org/wiki/R_(programming_language))

IT 서비스가 탄생한 초기에는 자체적인 인프라 구축이 필요했기 때문에 사업자들은 SW와 하드웨어를 모두 구매하여 인프라를 구축하였다. 하지만, 인터넷 데이터 센터(IDC) 서비스 제공자들이 등장하면서 사업자들은 직접적으로 인프라 구축을 하지 않고 이미 구축된 IDC 하드웨어 자원을 임대하여 IT 서비스를 제공할 수 있게 되었다. 이후 IDC 사업자들은 클라우드를 도입하여 사용자가 원하는 자원의 양을 동적으로 조정할 수 있게 되면서 IDC 자원을 보다 효율적으로 관리할 수 있게 되었고 사용자들은 하드웨어 자원 임대 비용을 절감할 수 있게 되었다. 그리고 클라우드 기술이 발전함에 따라 하드웨어 자원 임대 뿐만 아니라 서비스 구현에 필요한 SW까지도 임대할 수 있게 되었다.

이러한 클라우드 기술의 진화는 IT 인프라 시장의 판도를 변화시키고 있으며 점차적으로 클라우드 시장의 비중을 늘려가고 있다. 클라우드 자원을 외부에서 임대해서 사용하는 공공 클라우드 서비스와 클라우드 서비스를 자체적으로 구축하는 사설 클라우드 시장 모두 증가하고 있다. 특히 공공 클라우드 서비스를 제공하는 아마존, MS, 구글 등의 시장 규모도 지속적으로 증가하고 있다. IT 분야에서 중요한 클라우드 기술도 대표적인 공개SW들이 많이 활용되고 있다. 본 보고서에서는 오픈스택, 클라우드 파운드리, 도커, 쿠버네티스 등에 대해서 살펴보도록 한다.

[그림 3-12] IT 인프라 시장에서의 클라우드 비중 및 주요 클라우드 사업자들



오픈스택은 리눅스 이후 가장 주목 받는 공개SW 프로젝트 중 하나이며 2010년 7월 Rackspace와 NASA의 주도로 시작된 프로젝트를 기반으로 다양한 형태의 클라우드 인프라

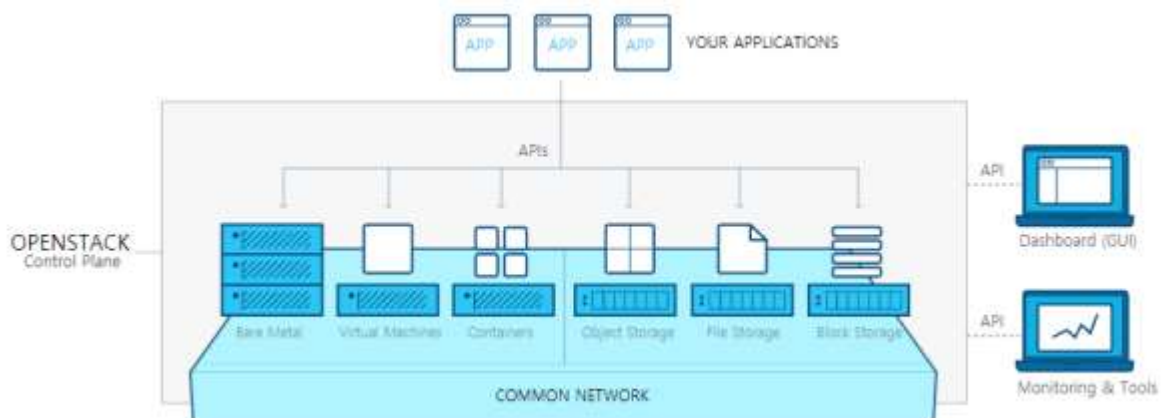
45) David H Deans(2018), Why the cloud IT infrastructure market is set to reach \$52.3bn in 2018, CloudTech.
 46) Mike Wheatley(2018), Cloud infrastructure giants Amazon, Microsoft, Google, and Alibaba all gain market share, siliconANGLE.

를 손쉽게 구축할 수 있는 클라우드 플랫폼을 개발하기 위한 프로젝트이다. 컴퓨팅, 저장장치, 네트워크 자원으로 구성된 대규모 자원들을 제어 및 관리하는 클라우드 플랫폼으로 웹 기반의 사용자 인터페이스를 통해 관리자가 손쉽게 자원을 제어하고 서비스를 관리할 수 있도록 하며, 서비스 이용자는 가상 인프라 서비스를 제공받고 동적으로 관리할 수 있는 기능을 포함하고 있다.

오픈스택의 성장으로 오픈스택 재단이 2013년도에 설립되었다. 해당 재단에는 Suse, 에릭슨, 화웨이, 델, 도이치텔레콤, AT&T, 인텔, 후지쯔, 레드햇 등 컴퓨터 장비 제조사, SW 기업 및 통신 사업자등 다양한 기업들이 회원사로 참여하고 있다. 오픈스택⁴⁷⁾은 2018년 8월까지 10,764명의 기여자들에 의해 942,125번의 커밋되었고 코드 라인 수는 9,105,385 라인에 이르고 있다. 대부분 Python언어로 작성되었으며 COCOMO 모델에 의하면 2,830년의 노력이 들어간 것으로 추정되면 아파치 라이선스를 따르고 있다.

오픈스택은 클라우드 플랫폼으로 주요 기능들을 각각 독립된 하부 프로젝트로 구분되어 개발되고 있으며 초기에는 컴퓨팅 서비스인 Nova, 객체 저장 서비스인 Swift, 이미지 관리 서비스인 Glance의 3가지 하부 프로젝트로 시작되었으나 지속적으로 새로운 프로젝트가 추가되면서 영역을 확장해 나가고 있다. 일례로 컨테이너 서비스를 위한 ZUN, 블록 저장 서비스를 위한 CINDER, 네트워킹 서비스를 위한 NEUTRON, 로드 밸런싱 기능을 위한 OCTAVIA 등 많은 하부 프로젝트들이 분리되거나 추가되었다⁴⁸⁾.

[그림 3-13] 오픈스택의 구성 요소



출처: 오픈스택(<https://www.openstack.org>)

47) <https://www.openhub.net/p/openstack>

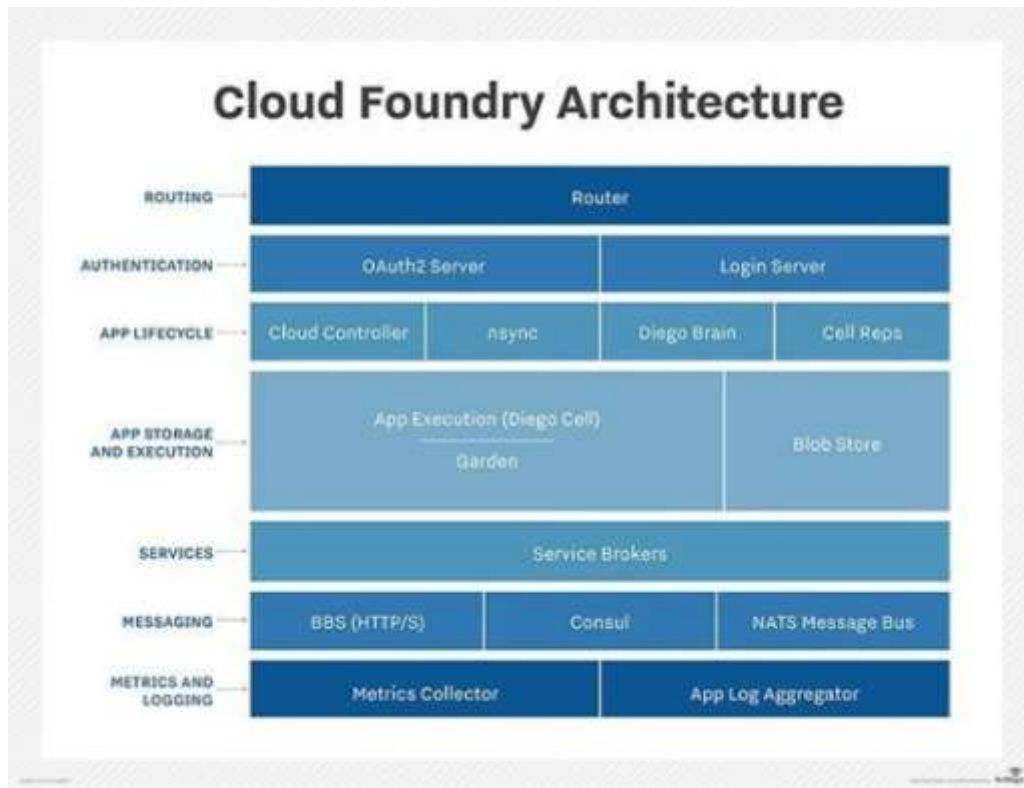
48) OpenStack, OpenStack Components.

오픈스택은 공공 클라우드 서비스와 사설 클라우드 서비스 구축에 많이 활용되고 있다. 도이치텔레콤, 차이나텔레콤, 텔레포니카, 오렌지 등의 통신사들의 클라우드 서비스 구축을 포함한 다양한 공공 클라우드 서비스 구축에 활용되고 있으며, 타타통신, IBM, 우분투, 차이나 모바일, 텔레포니카 등의 사설 클라우드 구축에도 활용되고 있다. 이외에도 월마트, 타켓, T-Mobile, 야후, 케세이 퍼시픽, 에딘버그 대학 등 다양한 활용 사례들을 보유하고 있는 클라우드 분야의 대표 공개SW이다⁴⁹⁾.

클라우드 파운드리(Cloud Foundry)는 원래 VMware에서 개발된 멀티 클라우드 플랫폼을 관리하기 위한 공개SW이다. 2015년 클라우드 파운드리 재단(Cloud Foundry Foundation)이 설립되면서 관련 프로젝트들을 관리하고 있으며 리눅스 재단의 협력 프로젝트로 진행되고 있다. 클라우드 파운드리 재단은 클라우드 파운드리 SW(소스 코드 및 모든 관련 상표) 관련 지적 재산을 보유하면서 중립적인 역할을 수행하고 있다. 그리고 피보탈, Comcast, SAP, IBM, 구글, 델 등의 업체들이 클라우드 파운드리 재단 활동에 참여하고 있다.

49) OpenStack Marketplace, <https://www.openstack.org/marketplace/>

[그림 3-14] 클라우드 파운드리 구조



출처: TechTarget

클라우드 파운드리의 구성 요소는 플랫폼 내에서 데이터 전달을 위한 Routing 기능, 사용자 인증 관리를 위한 Authentication 기능, 클라우드 상의 응용 프로그램 관리를 위한 Application Lifecycle 기능, 데이터 저장 기능을 위한 Application Storage and Execution 기능, 응용 프로그램과 특정 서비스(e.g. 데이터베이스)와 연결을 위한 Service Broker 기능, 각각의 가상 머신들 간의 통신을 위한 Messaging 기능 등 클라우드 구축에 필요한 다양한 기능들을 오픈소스 형태로 개발되고 있다⁵⁰⁾.

클라우드 파운드리(Cloud Foundry)는⁵¹⁾ 2018년 8월까지 2,848명의 기여자들에 의해 104,222번의 커밋 되었으며, 코드 라인 수는 4,214,059 라인에 이르고 있고 대부분 Go언어로 작성되었다. COCOMO 모델에 의하면 1,253년의 노력이 들어간 것으로 추정되면 저장소는

50) TechTarget, <https://searchcloudcomputing.techtarget.com/definition/Cloud-Foundry>

51) <https://www.openhub.net/p/cloudfoundry>

깃허브⁵²⁾를 사용하고 있고 아파치 라이선스를 따르고 있다.

클라우드 파운드리⁵³⁾의 주요 활용 사례로는 IBM, 피보탈, SAP, Suse, 화웨이 등의 기업들이 클라우드 파운드리 재단으로부터 인증된 배포판을 통한 사업으로 추진하고 있다. 그리고, 구글, 아마존 AWS, IBM, MS 애저, 화웨이 등 공공 클라우드 서비스를 제공하는 사업자들도 클라우드 파운드리를 활용하고 있다. 이외에도 약 200개에 가까운 서비스들이 클라우드 파운드리와 협력하고 있고 약 14개의 업체에서 클라우드 파운드리를 활용한 훈련 프로그램들을 운영하고 있다⁵⁴⁾.

2013년 3월에 공개된 도커(Docker)는 컨테이너라고 불리는 운영체제 수준의 가상화를 지원하기 위한 공개SW로 리눅스 응용 프로그램들을 컨테이너 안에 배치하는 작업을 편리하게 해줄 수 있다. 이를 위해 SW 실행에 필요한 코드, 런타임, 시스템 도구, 시스템 라이브러리 등을 모두 컨테이너화 시킴으로써 실행 환경에 관계없이 동일하게 동작시킬 수 있는 장점을 가지고 있다.

도커 컨테이너는 리눅스 커널이 제공하는 기능들 위에 올라가기 때문에 일반적인 가상 머신과는 달리 별도의 운영 체제를 컨테이너에 포함하지 않는다. 그리고 리눅스 커널의 기능을 직접 활용하여 컴퓨터 자원들(CPU, 메모리, 블록 입출력, 네트워크 등)을 격리함으로써 운영 체제 위의 응용 프로그램들을 서로 구분할 수 있도록 한다. 이러한 특징들 때문에 도커는 다른 가상화 기술 보다 가볍게 동작할 수 있고 하나의 서버에서 여러 컨테이너들을 보다 효율적으로 구동시킬 수 있게 된다.

도커⁵⁴⁾는 2018년 8월까지 11,382명의 기여자들에 의해 257,984번의 커밋이 있었으며 코드 라인 수는 9,236,254 라인에 이르고 있고 대부분 Go언어로 작성되었다. COCOMO 모델에 의하면 2,898년의 노력이 들어간 것으로 추정되며 저장소는 깃허브⁵⁵⁾를 사용하고 있고 아파치 라이선스를 따르고 있다.

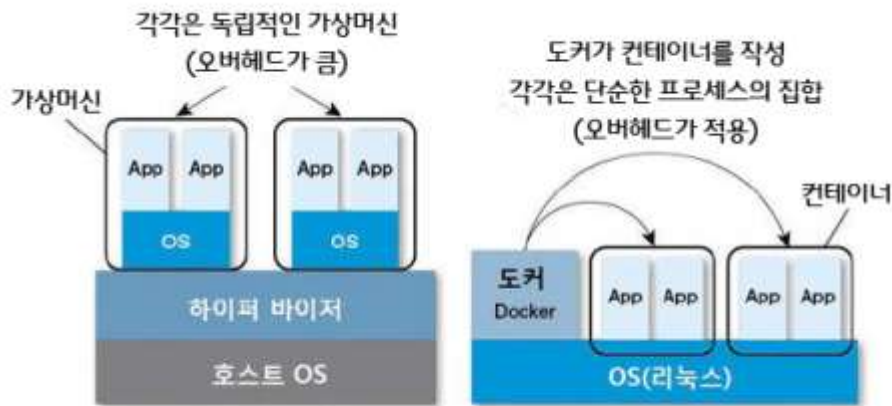
52) <https://github.com/cloudfoundry>

53) Gateway to the Cloud Foundry Ecosystem, <https://www.cloudfoundry.org/thefoundry/>

54) <https://www.openhub.net/p/docker>

55) <https://github.com/docker/swarm>

[그림 3-15] 일반적인 가상화 기술과 도커의 차이



출처: SK하이닉스 블로그, 도커 기술 소개

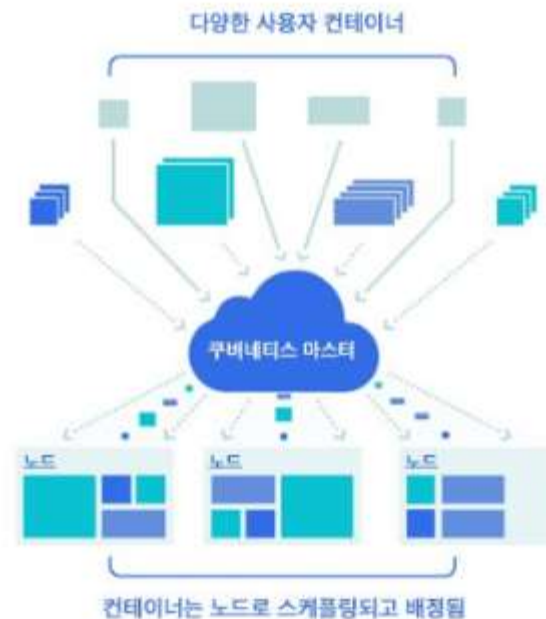
최근 도커는 IBM, 레드햇, 구글, MS 등 20여개의 기업들이 주축으로 오픈소스 컨테이너 표준을 위한 리눅스 재단의 프로젝트인 ‘오픈 컨테이너 프로젝트(Open Container Project)’에 컨테이너 이미지 관련 기능을 위한 참고 소스로 활용되고 있다. 그리고 대표적인 공공 클라우드 서비스인 아마존 AWS, MS 애저, 구글 클라우드 서비스들에서 도커를 활용할 수 있도록 지원되고 있으며, Expeida, Paypal, VISA 등 많은 기업들도 도커를 활용하고 있다⁵⁶⁾.

쿠버네티스(Kubernetes)는 컨테이너화된 응용프로그램의 배포, 확장 및 관리를 자동화하는 공개SW로 구글이 오랜 기간 컨테이너를 운영한 경험과 공개SW 커뮤니티에서 제공하는 좋은 아이디어와 결합되어 개발되었다. 주요 역할은 클라우드에 올라온 컨테이너화 된 수많은 응용프로그램들에게 레이블을 부여하여 컨테이너 작업의 부하와 서비스들을 관리하는 기능을 제공함으로써 쉽게 관리할 수 있도록 해준다.

쿠버네티스의 주요 기능들을 자세히 살펴보면 우선 서비스의 리소스 요구 사항 및 기타 제약 조건에 따라 컨테이너를 자동으로 배치할 수 있다. 두 번째로 서비스 중인 컨테이너가 종료될 경우 자동으로 복구할 수 있다. 세 번째는 수동 설정 또는 CPU 사용량에 따라 자동으로 컨테이너 확장이 가능하다. 네 번째로 컨테이너의 IP 주소 또는 DNS를 이용하여 로드 밸런싱이 가능하다. 마지막으로 응용 프로그램 상태를 모니터링 하여 인스턴스가 종료되지 않도록 응용프로그램을 배치할 수 있다.

56) CloudFoundary, Gateway to the Cloud Foundry Ecosystem

[그림 3-16] 쿠버네티스의 역할



출처: 네이버 포스트, 쿠버네티스

이러한 기능 장점들 뿐만 아니라 쿠버네티스는 벤더나 플랫폼에 종속되지 않는다는 장점을 가지고 있다. 이미 아마존 AWS, MS 애저, 구글 클라우드 같은 대부분의 공공 클라우드 서비스들에서 사용이 가능하며 오픈스택으로 구축한 사설 클라우드에서도 사용할 수 있다. 따라서 공공 클라우드와 사설 클라우드를 결합해서 사용하는 하이브리드 클라우드 시스템에서 가상화 기능을 제공할 때 중요한 역할을 담당하고 있다.

쿠버네티스⁵⁷⁾는 2018년 8월까지 2,235명의 기여자들에 의해 70,766번의 커밋을 되었으며 코드 라인 수는 1,583,639 라인에 이르고 있다. 대부분 GO 언어로 작성되었으며 COCOMO 모델에 의하면 448년의 노력이 들어간 것으로 추정되며 아파치 라이선스를 따르고 있다.

쿠버네티스의 활용 사례들을 살펴보면 공공 클라우드, 사설 클라우드, 하이브리드 클라우드 고려 없이 널리 사용되고 있다. 이미 구글 클라우드, 아마존 AWS, 구글 클라우드 뿐만 아니라 다수의 공공 클라우드 서비스에서 적용되어 사용자의 편의성을 향상시키고 있으며, box, Comcast, ebay, 화웨이, IBM, ING 등 클라우드를 활용하는 많은 기업들도 최신 기술인 쿠버네티스를 적극 활용하고 있다⁵⁸⁾.

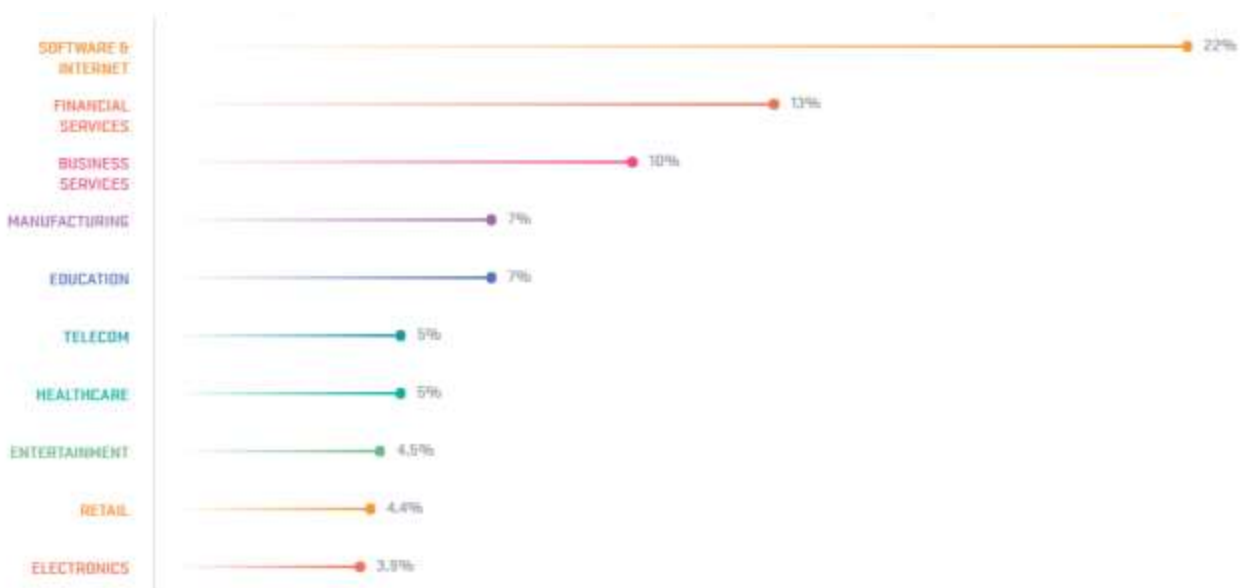
57) <https://www.openhub.net/p/kubernetes>

58) Kubernetes, 쿠버네티스 사용자 사례 연구, <https://kubernetes.io/ko/case-studies/>

제2절 주요 안전 산업 분야

최근 공개SW 활용 범위는 SW산업에만 머무르지 않고 확장되고 있는 추세이다. 2017년에 발표한 깃허브 동향 자료에서도 공개SW가 금융 서비스 분야, 비즈니스 서비스 분야, 제조 분야, 교육 분야 등 다양한 분야에서 활용되고 있음을 보여주고 있다. 특히 BearingPoint가 공개한 FOSS Management Study에서 밝힌 설문 조사에서는 자동차 생태계의 25개 회사를 대상으로 한 조사에서 기업의 85%가 공개SW와 연관되어 있다고 밝혔다⁵⁹⁾. 따라서 본 보고서에서는 SW안전 분야에서 공개SW 활용 사례를 파악하고자 우선적으로 인터넷 검색을 통한 문헌 조사로 대표 해외 사례들을 자동차 분야와 기타 분야로 나누어서 소개한다.

[그림 3-17] 2017년 깃허브 동향에 포함된 공개SW 활용 분야



출처: 깃허브, Octoverse 2017

1. 자동차 분야

인공지능, 빅데이터, 클라우드, 사물인터넷 기술들이 자동차 산업에서도 활용되기 시작하면서 공개SW의 중요성이 늘어나고 있다. 처음에는 IV(In-Vehicle Infotainment) 분야를 중심으로 자동차 산업에서 공개SW가 활용되기 시작하였고 최근에는 자율주행 차량 개발을 위

59) Blackduck(2012), BearingPoint, FOSS Management Study.

해 인공지능 기술이 활용되면서 자동차 산업에서 공개SW가 중요해지고 있다. 또한 자율주행과 관련된 SW안전의 문제를 공개SW 개발 방식을 활용해 해결하고자 하는 움직임도 생기고 있다. 본 보고서에서는 자동차 분야의 공개SW 활용 사례로써 바이두(Baidu)의 아폴로(Apollo) 프로젝트, Voyage 프로젝트, AGL(Automotive Grade Linux) 프로젝트, GENIVI 프로젝트들을 중심으로 설명한다.

사례 1: 바이두 아폴로(Baidu Apollo)

바이두는 2000년에 중국 베이징에서 설립된 인터넷 서비스 업체로 중국 내수 시장을 기반으로 세계 2위 인터넷 검색 서비스를 제공하고 있는 기업이다⁶⁰⁾. 바이두는 2017년 4월에 아폴로(Apollo) 1.0 버전을 공개하면서 자율주행차량 공개SW 프로젝트를 출범시켰다. 2018년 11월 기준 127개사가 바이두 아폴로 프로젝트에 참가하고 있으며 현대자동차도 2018년 6월에 아폴로 프로젝트에 참여하기로 발표했다⁶¹⁾. 아폴로 프로젝트는 공개SW 개발 방식을 활용한 자율주행 관련 오픈 소프트웨어 플랫폼으로 클라우드 기술 활용 및 하드웨어, 차량 인증 등의 100여개의 하부 프로젝트들을 포함하고 있다.

아폴로 프로젝트가 공개SW 프로젝트로써 가지는 의의는 크게 3가지로 볼 수 있다. 첫째, 자동차 분야의 공개SW가 기존의 IVI분야를 넘어 안전(Safety) 영역이 속하는 자율주행을 위한 프로젝트란 것이다⁶²⁾. 둘째, 협업을 위해 공개SW를 활용하고 있으며 지도 데이터의 30%는 타사의 데이터를 사용하고 있으며 공개SW 생태계를 적극 활용하고 있다는 것이다⁶³⁾. 셋째, 링컨MKZ 모델에 시범 적용된 자율주행 기술은 차폭 등 일부 데이터 수정만 거치면 다른 차종에 이식할 수 있을 정도로 유연하며 높은 완성도를 보이는 공개SW라는 점이다⁶⁴⁾.

바이두가 아폴로 프로젝트에서 공개SW 개발 방식을 활용하는 이유는 크게 3가지로 첫 번째는 협업으로 파트너들과 협력을 통해 신뢰성 있는 차량용 자동화 시스템을 개발하기 위해서이다. 두 번째는 고해상도 지도 서비스, 자율 주행 시뮬레이션 엔진, 심층 학습 알고리즘 분야의 혁신의 가속화를 위해서 이다. 마지막은 데이터를 빠르게 모으기 위해 참여자 수를 빠르게 증가시킴으로써 생태계를 구축하기 위함이다. 이러한 목적으로 바이두는 인공지능 같은 SW 개발에 집중하고 하드웨어는 파트너사에서 개발하는 형태로 프로젝트를 운영하고 있다. [그림 3-18]은 바이두 아폴로의 과거와 미래에 대한 이정표로 2021년에 고속

60) CHINA INTERNET WATCH (2017), China search engine market share in Apr 2017

61) Baidu Apollo, <http://apollo.auto/>

62) embedded (2017), Will China and Baidu's Apollo accelerate autonomous transportation?

63) AUTOMOTIVE Electronics (2018), 바이두 자율주행차 프로젝트 핵심은 'HAD맵과 AI'

64) technode (2017), Baidu launches their open platform for autonomous cars-and we got to test it

도로에서 완전 자율주행차를 목표로 하고 있다.

[그림 3-18] 바이두 아폴로 이정표



출처: Baidu Apollo

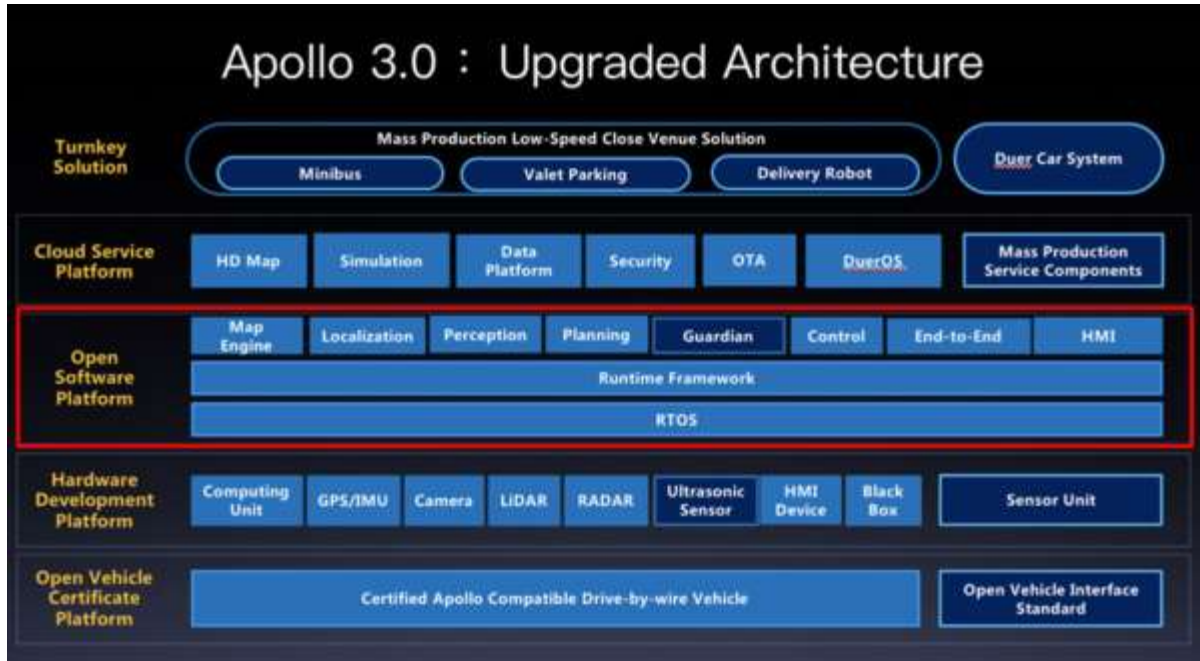
바이두는 2018년 7월 미국 자동차공학회(SAE) 기준 레벨4 수준의 아폴로3.0 버전을 공개했다⁶⁵⁾. 레벨4 수준은 보편적으로 운전자의 개입이 필요 없지만 만일의 상황을 대비하여 운전자가 탑승하여야 하는 자율주행 수준이다. 아폴로3.0 플랫폼은 [그림 3-19]처럼 실시간 운영체제(RTOS), 런타임 프레임워크, IVI 엔진, HMI, 가디언(Guardian) 등 으로 구성되어 있다. 가디언 기능은 안전(Safety)에 직접적으로 관여하고 있다⁶⁶⁾.

가디언은 아폴로 3.0의 주요 기능중 하나로 비상 상황에서 차량을 조심스럽게 멈추게 하여 차량의 안전을 보장하는 기능을 의미한다. 이 기능을 위해서는 모니터링이 핵심이며 항공 산업에서의 헬스 모니터와 매우 유사하다. 모니터는 HW와 SW로부터 발생하는 치명적인 에러 신호를 받게 되면 SW에 오동작이 있는지 판단한 후에 가디언과 운전자에게 음향 및 시야 경고를 보내게 된다. 그 후 운전자가 차량을 제어 권한을 받게 된다.

65) Pandaily (2018), AI Feast at Baidu Create 2018: Level 4 Autonomous Bus, Apollo 3.0, DuerOS 3.0

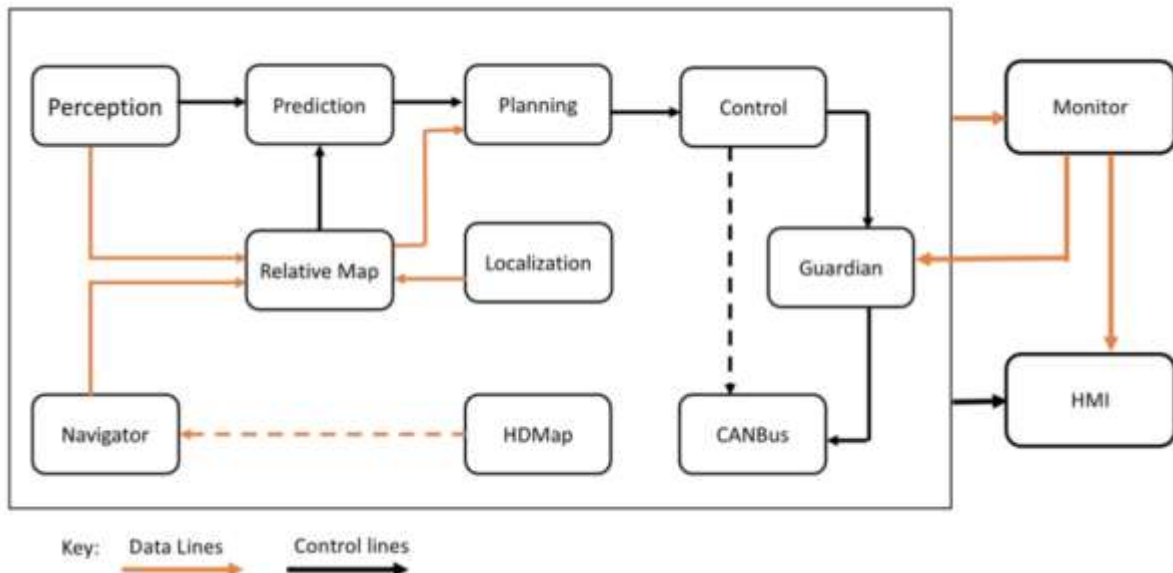
66) Medium Corp (2018), Apollo3.0:Entering the New Era of Autonomous Driving

[그림 3-19] 아폴로 3.0 플랫폼



출처 : Medium Corp. 2018년 7월 11일 기사 - Apollo3.0:Entering the New Era of Autonomous Driving

[그림 3-20] 아폴로 3.0에서의 가디언 관련 SW 구조



출처 : Medium Corp. 2018년 7월 11일 기사 - Apollo3.0:Entering the New Era of Autonomous Driving

그리고 아폴로 3.0은 더욱 다양하고 신뢰성있는 인식 기능들을 제공하고 있다. 우선 가장 가까운 경로상 차량을 탐색하여 차선 탐색만으로 신뢰할 수 없는 차선 유지 및 차량 흐름

유지를 효율적으로 할 수 있다. 그리고 실질적인 센서 환경을 반영하여 다양한 동적 변화에 대응할 수 있도록 라이다(LiDAR), 레이더(Radar) 및 카메라 데이터와 정보들을 비동기로 융합하는 기능을 제공하고 있다. 또한 초음파 센서를 활용한 자동 비상 브레이크(AEB, Automated Emergency Brake) 기능 제공 및 전체 차선을 탐색하기 위한 기능 등을 제공하고 있다.

또한 바이두는 아폴로 프로젝트를 위해 HAD(Highly Automated Driving) 지도 데이터 수집에 주력하고 있다. 아폴로 프로젝트에서 HAD 지도 데이터의 30%는 협력사들이 제공하는 데이터라고 밝혀져 있다. 이에 대해 바이두의 인공지능 자동차부문 총괄 기획자인 양귀(Yang Guo)는 이와 같은 협력이 아폴로를 오픈소스 플랫폼으로 만든 이유이며 협력사들이 아폴로 프로젝트에 참여하는 요인이라고 밝혔다. 그만큼 자율주행 지도 데이터는 매우 중요한 요소라고 할 수 있다. 바이두의 오픈소스 지도 데이터의 집합인 Apollo Scape⁶⁷⁾는 이미 미국의 대표적 오픈소스 데이터 세트인 'Kitti'와 'cityscapes'보다 10배 이상의 규모를 가지고 있다고 알려져 있다.⁶⁸⁾

[그림 3-21] Apollo scape 예시 화면



출처 : Medium Corp, Baidu Apollo Releases Massive Self-driving Dataset; Teams Up with Berkeley DeepDrive

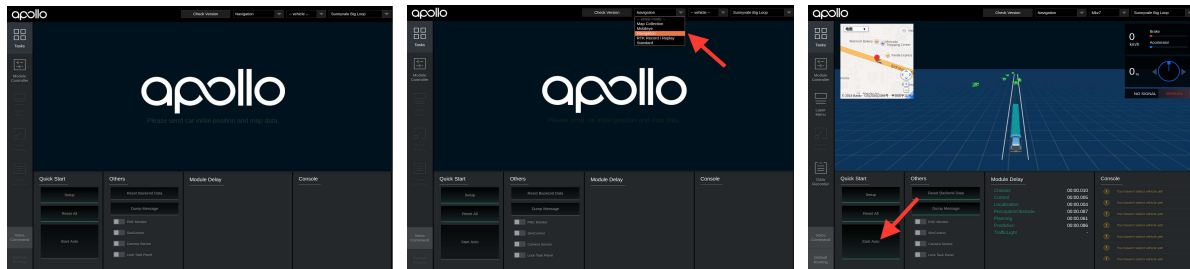
Apollo Scape는 [그림 3-21]처럼 비디오 이미지를 픽셀 단위로 나누어 자전거와 자동차 등 26개의 항목으로 정의하고 있다. 이를 통해 자율주행 차량은 도로, 자동차, 사람, 표지판 등을 서로 다르게 식별하고 있으며, 단순히 기능 제공에서 벗어나 자율 주행의 안전성에 대한

67) 아폴로 스케이프 공식 홈페이지. <http://apolloscape.auto/>

68) Medium Corp (2018), Baidu Apollo Releases Massive Self-driving Dataset; Teams Up with Berkeley DeepDrive

테스트를 위한 기능도 개발이 이루어지고 있다. [그림 3-22]는 깃허브에서 개발되었던 아폴로 2.5 버전의 자율 주행 테스트 기능을 보여주고 있다. 이를 통해서 다양한 자율주행을 위한 다양한 환경을 제공하고 시뮬레이션 할 수 있는 환경을 제공하고 있다.

[그림 3-22] Github를 통해 공개되어진 Apollo 2.5 버전 자율 테스트



출처 : <https://github.com/ApolloAuto/ApolloAuto.github.io>

바이두의 아폴로 프로젝트의 가장 중요한 의미는 공개SW가 안전 분야에서도 활용될 수 있다는 것이다. 아폴로 프로젝트는 차량용 SW 개발을 위해 공개SW 개발 방식을 활용하는 프로젝트로 차량 안전성 제공을 위해 HAD 지도의 안전감지 기능 같은 안전성을 고려하고 있다. 예를 들어, HAD 지도에서 시각적으로 운전자에게 물웅덩이, 강가, 교통 체증 등의 정보를 안내하고 있다. 이러한 기능으로 운전자들은 안전과 관련된 정보를 사전에 인지함으로써 안전 운전을 통해 사고의 가능성을 낮출 수 있다. 또한 차량 고장시 운전자의 대처 방안에 안내해 주는 기능을 통해 고속도로 같은 안전 사고 가능성이 높은 장소에서는 매우 유익할 수 있다⁶⁹⁾.

이러한 안전에 대한 고려는 아폴로 3.0에서 더욱 부각되고 있으며 아예 보호자를 의미하는 가디언(Guardian)이라는 새로운 기능을 정의하고 비상 상황에서 조심스럽게 차량을 정지시켜 운전자의 안전을 보장하고 있다. 특히 가디언 기능은 감시 시스템인 모니터 기능과 연계되어 HW와 SW에서 발생할 수 있는 치명적 오류를 감시하고 이에 대한 대응방안을 안내할 수 있다⁷⁰⁾.

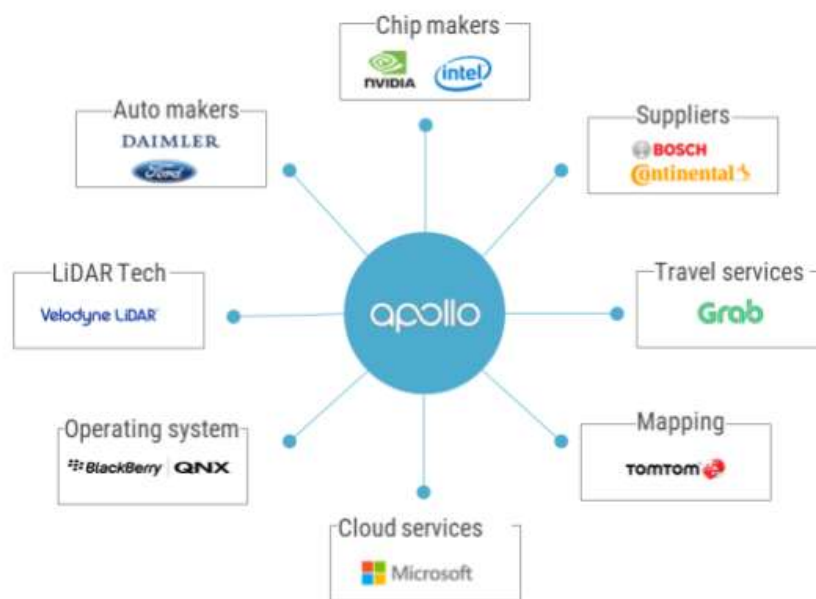
바이두는 아폴로 프로젝트에서 공개SW 개발 방식을 활용함으로써 단기간에 많은 협력사를 모집할 수 있었다. 그 결과 2018년 11월 기준으로 127개의 협력 및 회사들과 제휴를

69) 우버가 운영하는 자율주행차량의 사고 이후 안전분야에서의 주제는 훨씬 두드러졌고, 실제로 아폴로 3.0에서는 이러한 안전문제를 반영해 설계되어졌다.

70) Medium Corp (2018), Apollo3.0:Entering the New Era of Autonomous Driving

맺고 있으며 자동차 제조, SW개발, PC 제조, 카쉐어링 등 자동차 분야의 다양한 회사들이 프로젝트에 참여하고 있다. 이는 구글의 안드로이드 같은 공개SW를 활용한 생태계 구축 방법이 자동차 분야에서도 일부 효과를 보여주고 있다는 점에서 안전 분야의 공개SW 활용의 중요한 사례라고 할 수 있다⁷¹⁾⁷²⁾.

[그림 3-23] 바이두 아폴로 프로젝트 생태계



출처 : CB INSIGHTS (2018), Android of the Auto Industry? How Baidu May Race Ahead Of Google, Tesla, And Others In Autonomous Vehicles

사례 2: Open Autonomous Safety

미국 온라인 교육업체인 Udacity에서 2017년 분사한 자율주행 택시 업체인 Voyage는 공개 SW 프로젝트인 OAS(Open Autonomous Safety)⁷³⁾를 진행하고 하고 있다. OAS는 자율차량을 안전하게 테스트하기 위한 SW 플랫폼을 개발하기 위한 것이다. 수많은 신생 자율주행차량 업체들이 시간과 비용 문제로 개별적으로 안전 프로세스를 정의하기 어려운 현실에서 자율

71) 바이두 아폴로 공식 홈페이지. <http://apollo.auto/>

72) CB INSIGHTS (2018), Android of the Auto Industry? How Baidu May Race Ahead Of Google, Tesla, And Others In Autonomous Vehicles

73) VOYAGE. <https://oas.voyage.auto/>

주행의 차량의 안전 테스트를 위한 공개SW 기반 라이브러리를 제공하여 누구나 쉽게 테스트할 수 있하려고 한다.

OAS는 2018년 3월에 발생한 우버의 자율주행차량 사망사고와 2018년 5월에 발생한 테슬라의 자율주행차량 사망사고 이후에 깃허브에 오픈소스로 공개되었다. 그 이유는 공개를 통해 더 많은 피드백과 버그 등을 해결하고 많은 시나리오와 데이터를 수집하여 효율적인 자율주행차량의 안전 기능 테스트 프레임워크 구축하기 위해서 이다⁷⁴⁾.

[그림 3-24] Open Autonomous Safety의 주요 기능들



출처 : Open Autonomous Safety

이미 프로젝트 명에서 Safety(안전성)이 들어갈 정도로 안전을 위한 시나리오 테스트, 기능 안전(Functional Safety), 자율성 평가(Autonomy Assessment), 테스트 환경 구축(Test Toolkit) 등의 기능을 제공해준다. 기능 안전 관련 기능은 ISO 26262 표준의 자율주행차량 관련 안전에 대하여 고려하고 있다. 시나리오 테스트는 교외 환경에 중점으로 수백 가지 상황에 어떻게 대응해야 하는지가 자세하게 입력할 수 있다. 이 시나리오 중 하나는 2018년 3월 우버의 자율주행차량 사고가 포함되어 있다.

사례 3: Automotive Grade Linux

AGL(Automotive Grade Linux)는 리눅스 재단이 2012년 9월 설립한 차량용 임베디드 리눅

74) VOYAGE GitHub. <https://github.com/voyage>

스 개발 프로젝트로 차량용 SW를 위한 운영체제와 프레임워크를 개발하기 위해 설립되었다. 리눅스를 기반으로 공개SW 개발 방식을 활용하고 있는 AGL은 엔지니어와 구성원들이 정보를 공유하며, 피드백을 통해 발전하고 있다.⁷⁵⁾ 현재 닛산, 혼다, 메르세데스-벤츠, JLR, 파나소닉, 덴소, NTT 데이터 등 자동차 산업의 유명 회사 110개 가 참여하고 있다. 2018년 1월 CES에서 도요타와 아마존이 AGL을 공식 지원한다고 발표했다. 특히 도요타는 AGL에 있어 가장 적극적이며 캠리에 AGL을 시범 적용하고 있다⁷⁶⁾.

[그림 3-25] Autonomous Grade Linux의 주요 기능들



출처: Introduction to Autonomous Grade Linux

AGL의 시작은 IVI 기능 개발이었으나 이후 지속적으로 프로젝트를 확장하면서 계기판, 헤드 업 디스플레이, 텔레매틱스, 고급 운전자 지원 시스템(ADAS) 및 자율주행을 포함한 자동차 관련 모든 소프트웨어를 다루는 것으로 하고 있다.⁷⁷⁾⁷⁸⁾ 자동차 산업 전반에 걸쳐 하

75) LINUX FOUNDATION(2015). Automotive Grade Linux Requirements Specification

76) Automotive Grade Linux. <https://www.automotivelinux.org/>

77) LINUX FOUNDATION(2018), Introduction to Automotive Grade Linux

나의 SW 플랫폼 개발하고 차량 개발에 필요한 시작점의 70~80%까지 개발하는 것으로 목표로 하고 있다. 따라서 공개SW들의 좋은 기능들을 결합함으로써 차량용 SW의 파편화를 막고 개발사, 부품 제공사 등이 하나의 플랫폼을 이용하는 생태계를 구축하려고 한다.

[그림 3-26] Automotive Grade Linux 의 엔터테인먼트 콘솔 예시



출처 : PC World (2016), Automotive Grade Linux: An open-source platform for the entire car industry

AGL은 기존의 자동차 회사들이 참여하는 공개SW 프로젝트들과 달리 안전에 대한 논의 그룹인, 기능 안전 전문가 그룹을 운영하고 있다⁷⁹⁾. 이 그룹은 자동차의 기능 안전에 필요한 다양한 HW, SW 및 프로세스를 개발하는 것에 중점을 두고 있다. 이와 같이 안전이 중요해지는 이유 중에 하나는 ADAS(Advanced Driver Assistance System)의 발전에서 결합되어진 LKAS(차선 유지 지원 시스템), APAS(자동 주차 지원 시스템)와 같은 최신 기술들이 SW를 기반으로 하고 있으며 기능 안전(Functional Safety)이 자동차 산업에서 중요해지고 있기 때문이다.

이를 위해 AGL은 기능 안전을 위해 ISO 26262 표준을 이미 고려하고 있다. 그리고 AGL의 협력사 중 하나인 Audiokinetic 사(社)는 IVI 시스템이 안전 관련 경보를 운전자에게 전달할 수 있는 SW(SW명 : Wwise Automotive)를 이미 개발하여 자동차의 기능 안전을 위해 시각과 청각을 활용할 수 있는 디스플레이⁸⁰⁾를 사용하고 있다.

78) PC World (2016), Automotive Grade Linux: An open-source platform for the entire car industry

79) AGL 기능적 안전 전문가 그룹 소개. https://wiki.automotivelinux.org/functional_safety_expert_group

80) 공간적 청각 디스플레이(Spatial auditory display:SAD)란, 여러 가지 활동 및 지시사항을 특정한 위치와 연관시키는 청각적 단서를 제공함으로써, 운전자가 “직관적으로” 음성 및 지시사항의 출처, 위험의 위치와 그 상대적 궤도를 식별할 수 있게 해준다. 또 애플리케이션 프롬프트에 대한 반응시간과 도로 위의 위험한 장애물과 사건에 대한 반응시간도 단축시킬 수 있다. AUTOMOTIVE Electronics. 2014년 1월호 기사 - QNX, 공간적 청각 디스플레이 HMI

[그림 3-27] Wwise automotive의 작동 예시



출처 : Audiokinetic (2018), Integrated Automotive Audio Management

사례 4: GENIVI

GENIVI는 AGL과 유사하게 자동차 산업에서 공개SW 개발 방식을 활용하여 차량용 멀티 미디어 플랫폼을 표준화하기 위해 2009년에 설립된 단체이다⁸¹⁾. 차량 제조사 및 부품 공급 사들이 GENIVI 개발 플랫폼을 활용하여 빠르게 IVI 제품 및 차량을 개발하고 및 시험할 수 있도록 공개SW를 배포하고 있다⁸²⁾. 특히 GENIVI는 회사별로 특화된 IVI 관련 제품을 개발 할 수 있도록 핵심 모듈 및 기능들만 제공하기 때문에 관련 코드의 80%는 리눅스, 15%는 일부 수정코드 이며 나머지 5%는 GENIVI 특화 코드로 구성되어 있다. 2018년 기준으로 약 113개 이상의 회원⁸³⁾을 보유하고 있으며, 국내에서는 ETRI, 삼성전자, 현대자동차, LG전자 등이 회원으로 가입하고 있다.

GENIVI가 탄생한 배경은 모바일 기기들과의 호환성이 중요해지면서 이를 위한 공동의 노력이 필요했기 때문이다. 실제로 공개SW 개발 방식을 활용하고 있는 모바일 OS는 매년 새로운 기능들이 추가되고 있고 개선 속도가 차량용 시스템에 비해 매우 빠르기 때문에 차량용 플랫폼 제공업체는 모바일 OS의 개선 속도를 따라잡기 위해서 많은 노력을 들여야 했다. 따라서 자동차 제조사들도 이러한 문제점을 해결하고 차량용 인포테인먼트 개발 비용을 절감하기 위해서 IVI 기능의 공통 플랫폼을 협업으로 개발하려고 GENIVI를 출범시켰다⁸⁴⁾.

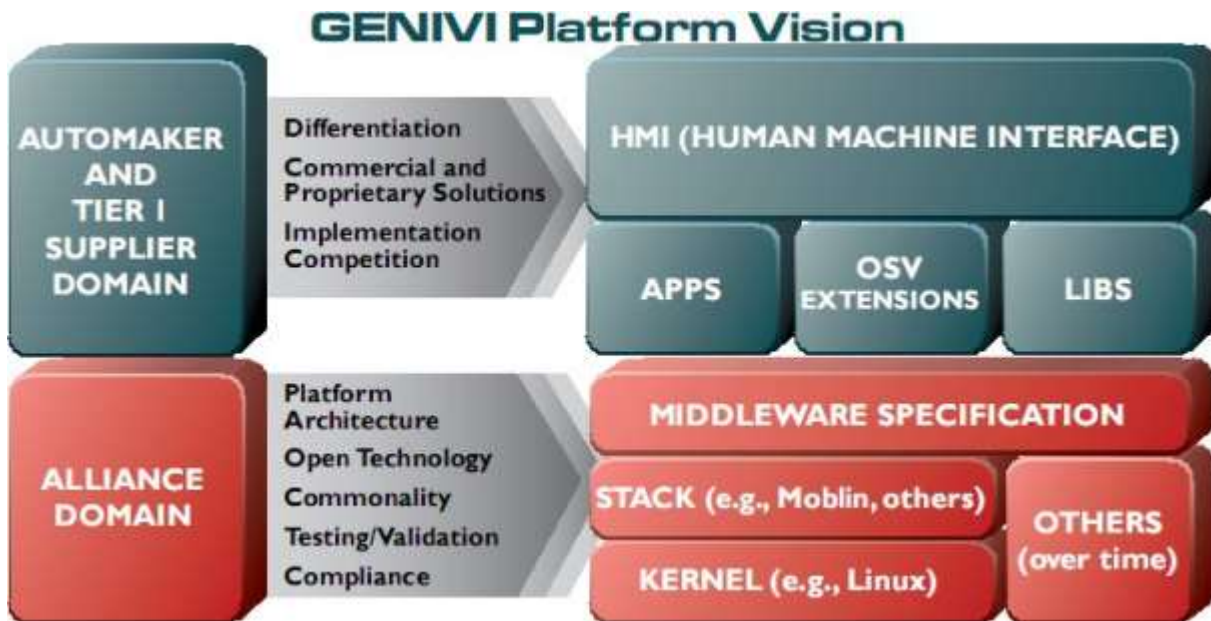
81) GENIVI(2014). BMW Case Study.

82) FOSS(2016). GENIVI Development Platform(GDP)

83) 제니비 회원사. <https://www.genivi.org/genivi-members>

84) GENIVI(2013). THE ROAD AHEAD Open Source IVI GENIVI

[그림 3-28] GENIVI 플랫폼 비전

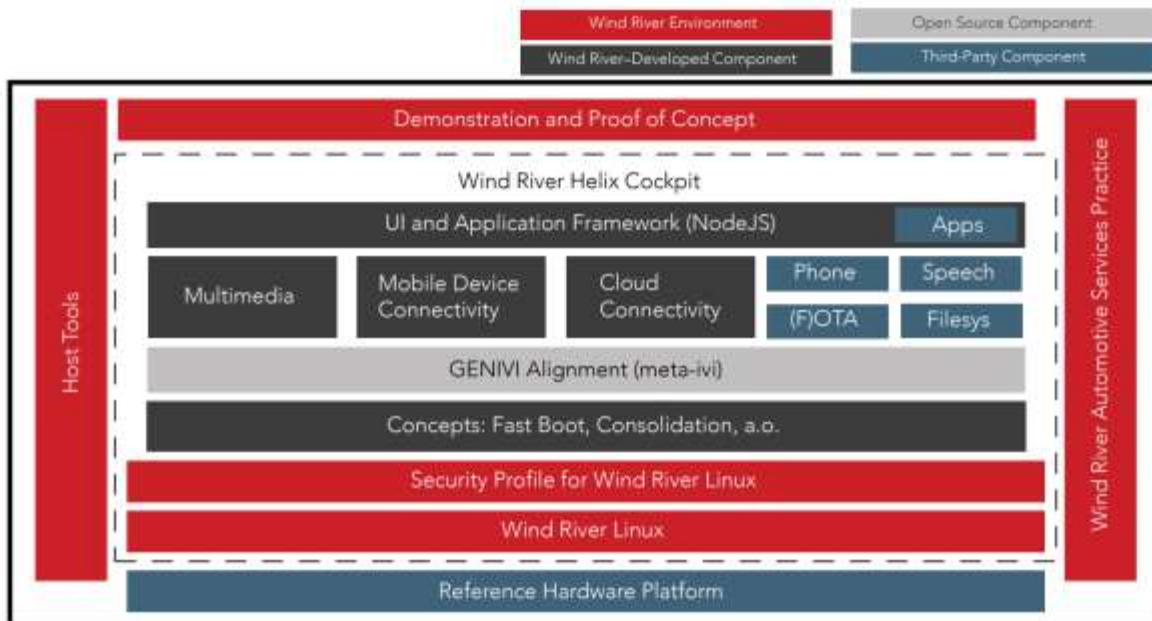


출처 : CNXSoft

다수의 회사들이 협력하여 플랫폼을 개발하는 공개SW 개발 방식의 GENIVI는 과거보다 최신 기술의 적용 속도 측면에서 일반 상용 플랫폼보다 유리하게 되었으며, 호환성 이슈에서도 개선되는 효과를 가져올 수 있었다. BMW와 PSA 등의 일부 자동차 업체들은 2013년부터 GENIVI 표준을 만족하는 제품만을 납품받고 있기 때문에 Wind river, Bosch, Magneti ma-relli, Tomtom 등 유럽 자동차 부품 제조사들은 GENIVI 표준을 따르는 제품을 이미 출시하고 있다.

Wind River 사는 1981년에 미국에서 설립된 임베디드 시스템 소프트웨어 개발 회사인데 2009년에 GENIVI의 창립 멤버로 참가하였고 인포테인먼트 및 디지털 클러스터 분야에서 리눅스 기반 GENIVI 및 AGL 호환 플랫폼으로 Helix Cockpit을 개발하였다. 이 과정에서 GENIVI의 오픈소스 모듈이 적극 활용되었고 안전과 보안이 강화한 자체 플랫폼을 개발할 수 있었다. 이를 통해 타 회사들과 안정적인 협력이 가능했으며 으며 모바일폰 등과 호환성을 향상시킬 수 있었다.

[그림 3-29] Wind River Helix Cockpit의 구조



출처 : Wind River

2. 기타 산업 분야

이번 문헌 조사에서는 철도, 항공 등 여러 산업 분야에서의 공개SW 활동들을 조사하였는데, 인터넷 조사라는 한계점으로 인하여 자동차 산업만큼의 충분히 자세한 사례들을 찾기가 쉽지 않았다. 따라서 다양한 안전 산업 분야 중에 대표적인 사례에 대하여 철도 및 일부 산업에서의 공개SW 활동 사례들에 대하여 소개한다.

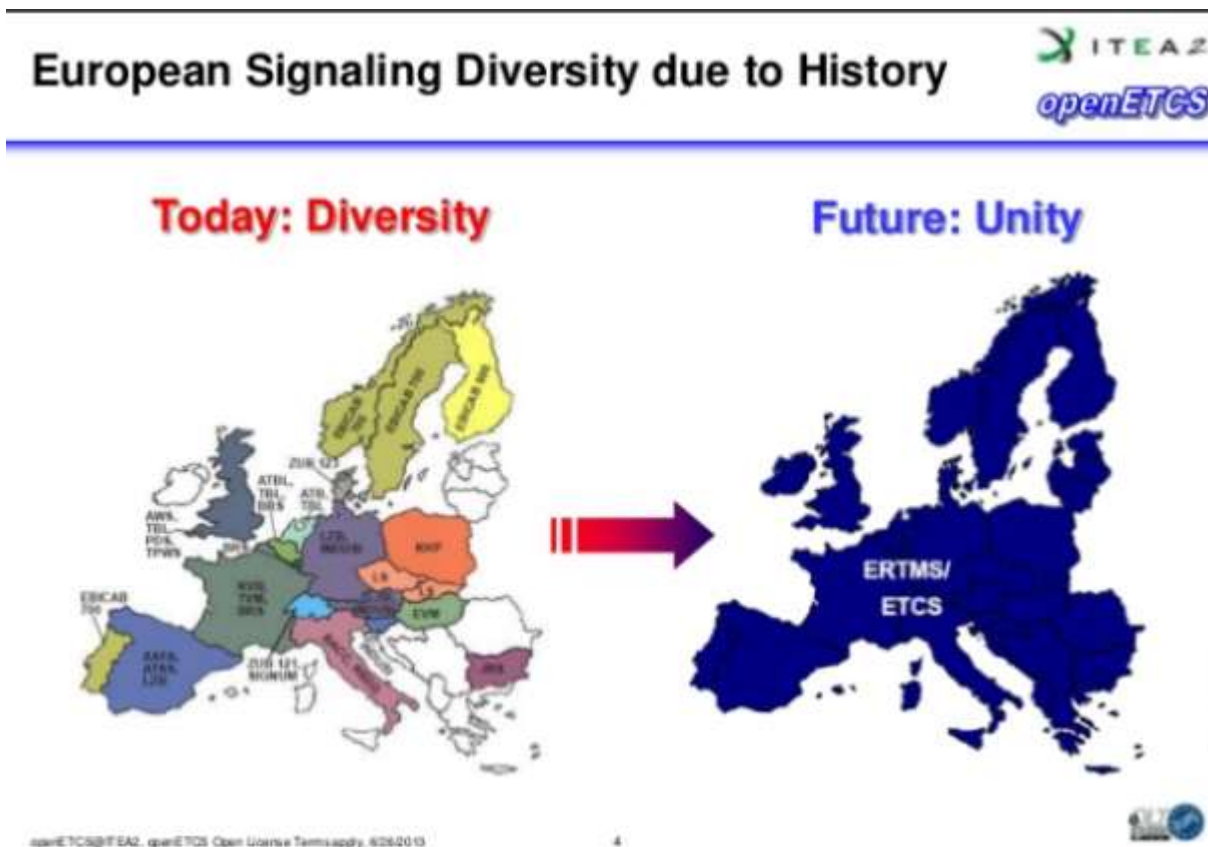
사례 1: openETCS

openETCS는 ‘ETCS(European Train Control System, 유럽 철도 제어 시스템)의 차량 탑재 장비에 들어가는 소프트웨어 프레임워크를 공개SW 개발 방식으로 개발하기 위한 프로젝트이다. 단순히 비용절감 효과를 얻기 위한 것이 아니라 검증, 구현, 유지, 보수 까지 공개SW 개발 방식의 효과를 얻기 위한 철도 분야에서 공개SW 개발 방식을 활용한 사례라고 할 수 있다. 원래 openETCS는 유레카 클러스터의 ITEA2(Information Technology for European Advancement)에 제안된 프로젝트의 이름으로 2012년 7월에 시작하여 2015년 12월까지 총 3년 5개월 동안 진행되었다.

유럽 철도의 역사는 150년이 넘기 때문에 그동안 국가별로 독립적으로 발전하여 왔다. 여

러 국가의 철도 시스템을 통합하는 과정에서 상호 운용성 이슈가 발생하게 되었다. 이러한 문제를 해결하고자 openETCS는 HW와 SW 사양, 인터페이스 정의, 설계 도구, 검증 및 검증 절차 및 내장형 제어 SW를 포함한 모든 단계에서 공개 표준(Open Standard)를 활용하려고 하였다.

[그림 3-30] openETCS의 목표



출처: SlideShare: openETCS ITEA2 2013 Review Overview

openETCS 프로젝트에는 독일, 프랑스, 이탈리아, 벨기에, 네덜란드, 스페인, 영국 등의 30개 기업들이 참여하였으며 특히 독일 기업 12개, 프랑스 기업은 10개 참여하여 프로젝트를 주도하였다. 참여 기업들 중에서는 독일국영철도사, 프랑스국영철도사, 알스톰 등 유럽의 주요 철도 기업들이 포함되어 있다⁸⁵⁾.

85) openETCS(2015). Open Proofs Methodology for the European Train Control Onboard System, openETCS co-summit

[그림 3-31] 프로젝트 참가 기업



출처 : openETCS(2015). Open Proofs Methodology for the European Train Control Onboard System, openETCS co-summit

또한, openECTS는 ETCS 시스템의 공식 모델, ETCS SW의 소스 코드 및 개발 및 검증에 필요한 도구들을 제공하려고 하였다. openETCS에서 제공하는 도구들을 활용하면 ETCS 사양을 활용한 제어 장치용 소스코드를 생성할 수 있게 되어 있다. 이를 위해 openETCS는 공개SW 개발 방식을 활용한 생태계를 구축하려고 하였다. 특히 공개SW 개발 방식을 활용하여 기존의 상호 운용성 문제를 해결하고, 제조업체, 인프라 관리자 및 철도 사업자를 모두 지원하고 검증 활동을 중립적으로 할 수 있도록 하였다.

이러한 사례는 철도 분야에 공개SW 개념을 도입하려는 시도으로써 유의미한 사례라고 할 수 있다. 특히 남북 협력을 통해 대한민국의 철도를 중국 및 러시아를 통해 유럽과 연결하려고 하는 노력이 있는데 이를 위해서는 국내 철도 운행에서도 유럽의 철도 제어 시스템에 대한 고려가 필요하다. 더군다나 철도 분야의 국제 규격으로 통용되고 있는 EU의 철도기준인 TSI(상호운용성 기술사양서: Technical Specifications for Interoperability), EN(유럽규격: European Norm), UIC Leaflets(Specification developed by the International Union of Railways)등을 만족하기 위해서는 openETCS에 대한 연구가 필요할 수 있다.

그리고 openETCS를 통해 ALSTOM과 GE는 2020년부터 인수, 설치, 테스트 및 시운전 비용을 10 ~ 20% 절감할 것으로 예상되면 openETCS 기능의 SW 유지 관리 비용도 50% 정도 절감할 것으로 예상하고 있다. 독일 철도 분야 기업인 도이체 반(Deutsche Bahn)은 2020년까지 차량 탑재 장치 당 ETCS의 조달 및 설치비용을 35만 유로에서 10만 유로로 낮추려고 하고 있으며 연간 SW 유지/보수비용도 차량 탑재 장치 당 4천 유로에서 1천 유로로 낮추려고 하고 있다. 또한 서비스 톨 제공 업체는 관련 분야의 전문 지식을 습득으로 경쟁적 우위를 차지할 수 있다고 한다⁸⁶⁾.

사례 2: 항공 분야

항공 분야는 자동차 분야 및 철도 분야와 달리 구체적인 자료를 찾기가 쉽지 않았다. 이는 항공 분야의 특수성(기술 보안 등) 때문에 외부에 자료 공개가 부족한 것으로 생각된다. 따라서 자료의 구체적이지 않지만 일부 기업들에서 공개SW가 활용되는 대략적인 내용에 대해서만 소개한다.

이에 대한 첫 번째 소개는 대표적인 민간항공기 및 군용항공기 개발 업체인 록히드 마틴이다. 록히드마틴은 미국 메릴랜드 주 베데스다에 본사를 두고 있는 첨단 기술 시스템, 제품 및 서비스의 연구, 설계, 개발, 제조, 통합 및 유지 관리에 주로 종사하는 글로벌 보안 및 항공 우주 회사이다.

록히드마틴은 2015년 StreamFlow라는 빅데이터 분석을 위한 SW 플랫폼을 공개하였다⁸⁷⁾. 이 SW는 데이터 스트림을 실시간으로 처리하게 해주는 공개SW 기반 계산 시스템으로 록히드 마틴 사(社)의 깃허브에서 다운받을 수 있고 아파치 라이선스를 활용하고 있다. Streamflow는 웹 인터페이스를 가지고 있으며, 드래그앤드랍 방식을 활용해서 쉽게 토폴로지를 구성하며, 대시보드를 통해 동작 상태 및 성능을 쉽게 관찰할 수 있다⁸⁸⁾.

Laika Boss는 록히드마틴에서 개발한 멀웨어 탐지용 공개SW로 깃허브⁸⁹⁾에서 다운로드 가능하다. 이 SW는 록히드마틴에서 2012년부터 사내 보안부서에서 록히드마틴의 자산과 인프라를 보호하기 위해 사용되었다. Laika Boss는 좋은 성능 때문에 록히드마틴의 다른 부서에도 사용하도록 권장되었고 2015년에 공개하였다. Laika Boss의 장점은 각 파일들을 세분화하여 트래픽 전체를 관찰하는 것이 아니라 세분화한 요소들을 각각 분석함으로써 부하가

86) openETCS(2015). Exploitable Results by Third Parties. page 2.

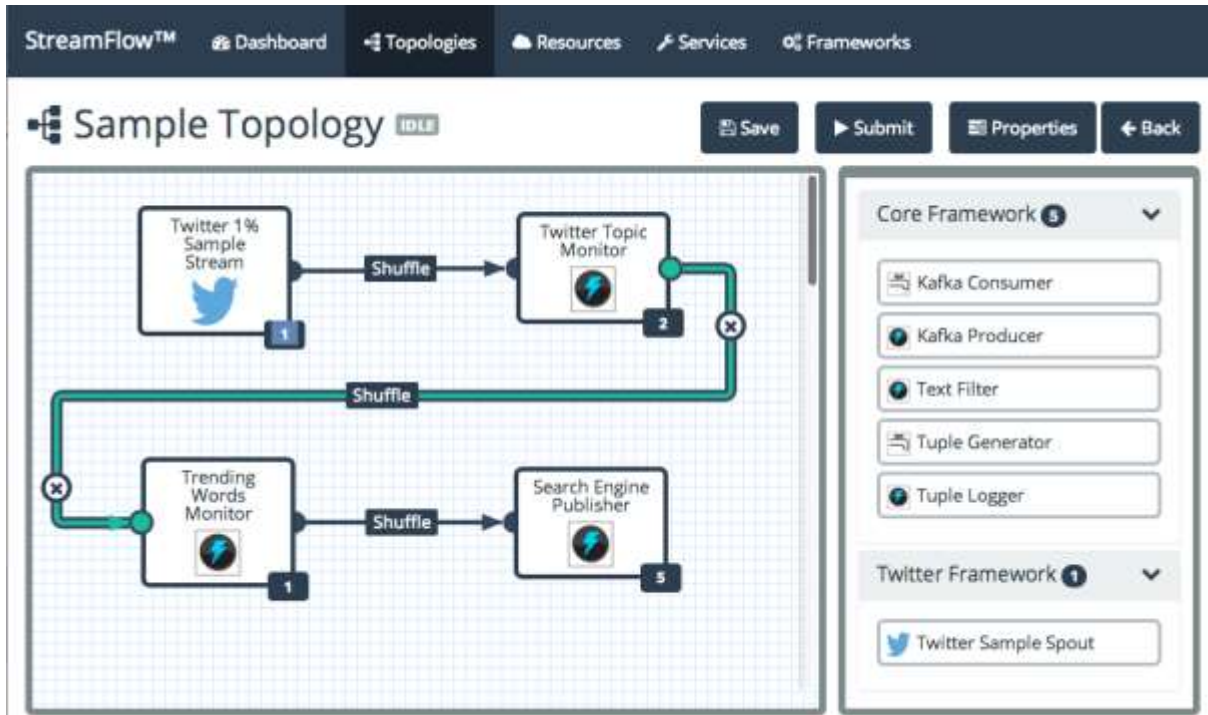
87) Military Embedded Systems(2015), Open source software platform released by Lockheed Martin

88) 깃허브, <https://github.com/lmco/streamflow>

89) 깃허브, <https://github.com/lmco/laikaboss>

적다는 것이 장점이다. 또한, 공개SW로 공개한 이후에 개발자들의 피드백이 활성화되어 공개SW로 전환한 후에 Laika Boss의 모듈은 20개에서 100개로 증가하였다.

[그림 3-32] Streamflow 화면

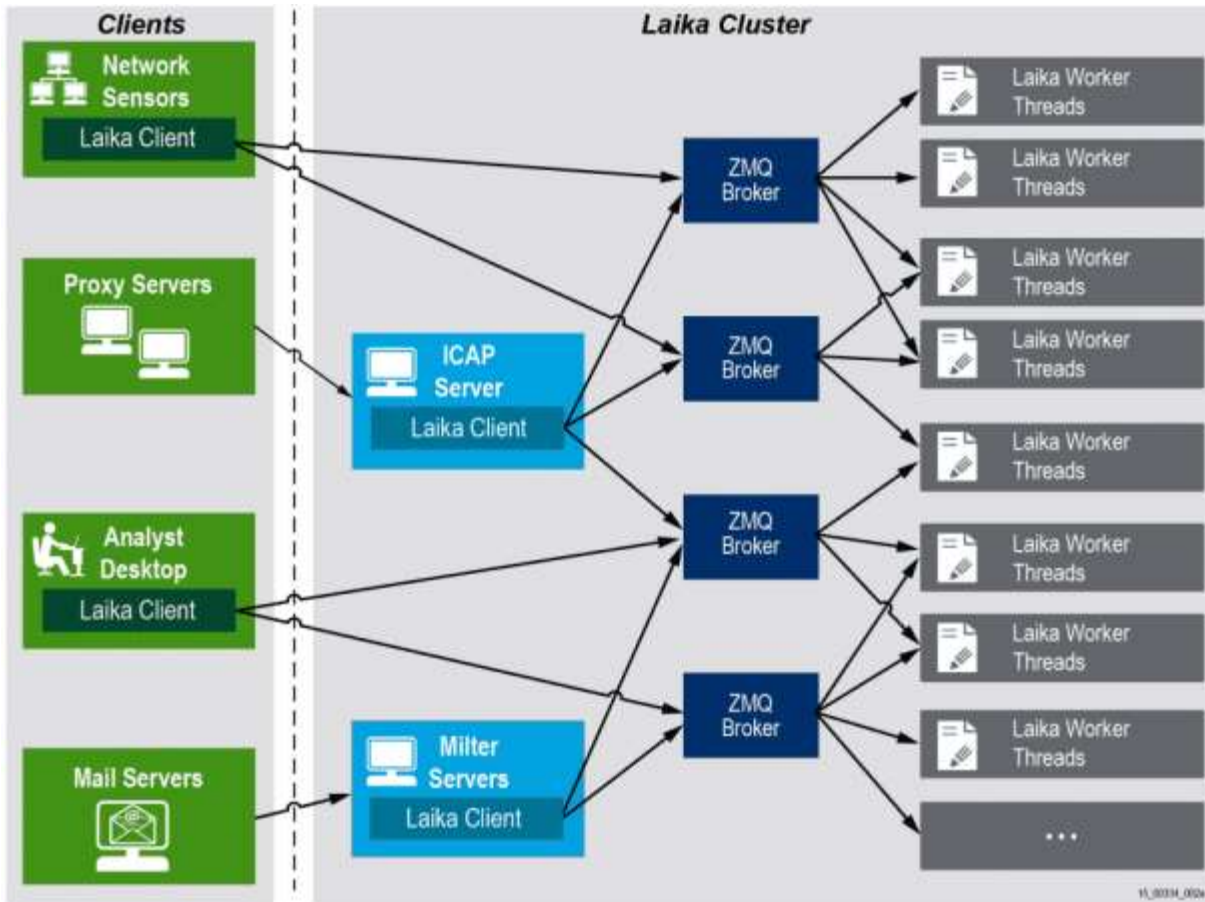


출처: 깃허브, streamflow

Open-DO 이니셔티브는 항공 분야의 안전 중요 SW의 해결되지 않는 이슈를 협력과 개방형 프레임워크를 통해 해결하기 만들어진 단체이다. 이 단체의 목적은 첫째, 안전 중요 SW 개발을 위해 품질을 갖춘 오픈소스 도구와 인증할 수 있는 구성요소들의 사용성을 보장하기 위해서 업체와 개발자간 협력을 위해서 설립되었다. 그리고 둘째, 안전 중요 SW의 개발의 진입 난이도를 낮추고 연구를 촉진하기 위해 학계에서 특정 안전 중요 SW의 개발과 수업 자료로 활용할 수 있도록 하고 공개SW 분야와 안전 중요 SW 분야의 활성화에 기여하려고 하고 있다.

이러한 목적 아래 공개SW와 안전 중요 개발자, 도구 제공자, 항공 산업 리더 및 학계 인증기관들의 참여를 통해 안전 중요 SW의 애자일 개발법, DO-178B를 활용한 안전 중요 SW의 분석 도구, DO-178B와 관련된 SW 설정 관리 절차 등 11개의 프로젝트들을 진행하였지만 공개SW로써 큰 영향력을 발휘하지 못하였다.

[그림 3-33] Laika BOSS의 동작



출처: Penetration Testing, Laika BOSS: Object Scanning System

미항공우주국(NASA, 나사)는 미국의 비 군사적인 우주개발을 총괄하며 종합적인 우주계획을 수립하기 위해 항공우주 활동 기획·지도·실시, 항공우주 비행체를 이용한 과학적 측정과 관측 실시 및 준비 작업 등을 수행하고 있다. 그리고 자체적으로 오픈소스 협약(NASA Open Source Agreement)을 정의하고 나사에서 개발한 여러 SW들을 공개하고 있다. 나사의 공개SW 홈페이지⁹⁰⁾에 의하면 항공기술을 포함한 환경, 데이터 처리 등 총 454개의 공개SW 프로젝트들을 공개하고 있다.

제3절 요약 및 시사점

해외에서 공개SW를 활용하는 범위가 점차적으로 넓어지고 있다. 초기 공개SW는 운영체

90) NASA Code Open <https://code.nasa.gov/>

제, 데이터베이스, 인터넷 액세스 서버 등 전통적인 SW산업 분야에서 공개SW 개발 방식을 활용하였다. 이후 인터넷의 발달과 함께 구글 안드로이드, 우버, 에어비엔비 등 새로운 분야에서 공개SW 개발 방식을 활용하여 플랫폼을 구축하여 서비스를 제공해주는 기업들이 늘어나게 되었다. 이들 분야는 과거에는 SW 산업을 분류되지 않았던 통신 산업, 운송 산업, 숙박 산업 등 비SW산업이었다. 또한, SW산업에서도 아마존, 페이스북 등 IT 인프라 및 IT 서비스의 판도를 바꾸는 기업들도 공개SW 개발 방식을 활용하고 있을 정도로 다양한 분야에서 공개SW 개발 방식을 활용한 새로운 비즈니스 진화를 이루어내었다.

그리고 최근에는 제4차 산업혁명의 핵심 기술이라고 할 수 있는 인공지능, 빅데이터, 클라우드 분야의 혁신 SW들이 공개SW 개발 방식을 이용하여 개발되고 있다. 인공지능 분야의 대표 기술인 심층학습 분야의 경우 텐서플로우, Accord.Net, CNTK, Mahout, MXNet 같은 대표적인 심층학습 프레임워크들이 공개SW 개발 방식을 활용하여 개발되고 있다. 빅데이터 분야도 대표적인 플랫폼인 하둡과 대표 통계처리 도구인 R이 공개SW 개발 방식을 활용하여 개발되고 있다. 또한, 클라우드 분야 역시 오픈스택, 클라우드 파운드리 재단 같이 공개SW로 개발된 플랫폼들이 재단으로 성장하였고, 도커, 쿠버네티스 같은 혁신 SW들이 공개SW 개발 방식으로 새로이 개발되고 있다.

이와 같이 제4차 산업혁명의 핵심 기술 분야에서 공개SW 개발 방식이 널리 사용되는 근본적인 이유는 공개SW 개발 방식의 장점들 때문이다. 첫째, 공유·협업을 통한 생태계 구축에 유리하다. 개발하는 신기술 SW를 공개하고 공유함으로써 새로운 신기술을 사용하고자 하는 개발자들이 쉽게 접근하도록 하고 지속적으로 최신 기술들을 제공함으로써 이들 개발자들이 신기술을 활용하는데 해당 공개SW를 활용할 수 있게 한다. 이로써 공개SW 기반 신기술들을 활용하는 개발자를 확보하고 해당 기술의 저변을 확장시킴으로써 신기술 생태계의 주도권을 확보할 수 있게 된다.

둘째, 신기술의 기술 혁신의 가속화와 완결성을 보다 빠르게 확보할 수 있다. 새로운 기술의 경우 품질의 문제가 발생할 가능성이 높게 된다. 새로이 개발하는 HW 제품의 경우 일정 수준의 품질을 안정적으로 유지하기 위해 품질 관리 절차를 수행하고 해당 제품을 시범적으로 사용하게 하여 결함을 발견하고 이를 개선하는 방안을 수행하고 있다. 새로운 SW 역시 개발된 기술에 대한 검증 절차가 필요하다. HW와 달리 복잡한 SW의 경우 다양한 경우의 수를 실질적으로 테스트하기 쉽지 않게 된다. 하지만, 공개SW 개발 방식을 활용할 경우 손쉽게 신SW를 테스트해줄 수 있는 개발자들을 확보할 수 있는 장점이 있으며 새로운 기능 추가에 대한 다양한 의견을 보다 많은 개발자들로부터 수집할 수 있는 장점이 있다.

공개SW 개발 방식은 SW 신기술 분야의 기술 혁신을 주도하기 위해 널리 활용되고 있을

뿐만 아니라 타 산업 분야에서도 공개SW 개발 방식을 활용하려는 시도가 생기고 있다. 대표적인 분야는 자동차 분야로 IVI 분야만 아니라 최근에는 자율주행 기능 개발을 위해 공개SW 개발 방식의 활용이 늘어나고 있다. 또한, 유럽에서는 철도 분야에서 상호 호환성을 제공하기 위해서 공개SW 개발 방식을 활용하였으며 록히드 마틴, 보잉 등 일부 항공회사에서도 공개SW 개발 방식을 도입한 것으로 알려져 있다.

자동차 분야의 경우 자동차 제조사들이 차량용 멀티미디어 플랫폼을 표준화하여 제품 및 차량 개발을 위한 비용 절감하기 위해 2009년 GENIVI를 설립하였다. 이를 위해 핵심 기능을 담당하는 임베디드 SW의 운영체제를 개발하기 위해 리눅스를 활용하였으며 공통 기능을 위해 공개SW 개발 방식을 도입하여 GENIVI 플랫폼을 개발하였다. 자동차 제조사 및 부품 공급업체들은 표준화된 GENIVI 플랫폼을 기반으로 추가 기능을 개발하거나 UI(User Interface)를 차별화하여 빠르게 제품을 개발할 수 있게 되었다.

자동차 분야에서 공개SW 개발 방식의 효율성을 향상시키기 위해 자동차 회사들은 리눅스 재단과의 협력의 필요성이 제기되었고 리눅스 재단과 협력하여 2012년 AGL 프로젝트가 시작되었다. AGL도 초기에는 IVI 기능 개발에 중점을 두었으나 점차적으로 기능을 추가하여 점차적으로 안전과 밀접한 분야들로 확장을 하였고, 지금은 ISO 26262에서 제공되는 기능 안전을 고려하기 위한 전문가 그룹을 만들었으며 ADAS(Advanced Driver Assistance System) 기능을 위한 안전을 논의하고 있고, 자율 주행을 위한 SW 개발에서 안전을 논의하고 있다.

최근 중국의 인터넷 서비스 업체인 바이두는 자율주행 차량 개발을 위해 공개SW 개발 방식을 활용한 아폴로 프로젝트를 적극적으로 추진하고 있다. 아폴로 프로젝트는 2017년 4월에 처음 공개되었으며 2018년 7월에 3.0 버전이 추가로 공개되었다. 바이두는 아폴로 프로젝트에서 다양한 자동차 산업들의 기업들과 협력하기 위해 공개SW 개발 방식을 활용하고 있으며 현재 100개 이상의 기업들이 해당 프로젝트에 참여하고 있다. 자율주행의 안전을 위해서 비상 상황에서 운전자에게 경고를 보내고 제어 권한을 운전자에게 돌려주기 위한 가디언 기능을 개발하고 차선 유지 및 차선 흐름 유지를 위한 기능의 성능을 개선하여 안전성을 확보하려고 노력하고 있다.

이외에도 도요타의 자율 주행 시뮬레이터 기부 및 Voyage의 OAS 등 자동차 분야에서 공개SW 개발 방식을 활용하는 사례가 늘어나고 있다. 그 배경에는 자율주행이라는 최첨단 기술 개발 비용 절감 및 생태계 구축을 위한 목적이 있다. 하나의 회사에서 기능 개발에 소요되는 비용 절감과 해당 기술을 활용할 수 있는 기업들을 개발 단계에서 확보함으로써 사업의 안정성을 확보하기 위한 생태계 구축 목적이 있다. 특히 OAS는 자율주행에서의 다양한 안전 위험을 대비하기 위한 시험을 위한 참여자의 피드백과 버그 해결을 위한 시나리오와

데이터를 수집하기 위해서이다. 즉 SW를 공개하여 많은 사람들이 사용하게 함으로써 SW 신뢰성을 확보하기 노력의 일환으로 공개SW 개발 방식이 활용되고 있는 것이다.

자동차 분야 이외에도 철도 및 항공 분야에서도 공개SW가 활용되는 사례에 대한 조사를 수행하였으나 자세한 정보를 확보하지 못하였다. 철도와 항공 분야의 경우 자동차 분야와 달리 시장 참여자들이 다양하지 못하기 때문이라고 할 수 있다. 물론, 유럽의 OpenETCS, 록히드마틴, 나사 같은 일부 사례가 있지만 이들은 해당 분야에서 특수한 사례로 보인다.

OpenETCS는 유럽의 여러 국가들 사이의 철도 교통 체계의 호환성 제공을 위해 시작한 프로젝트이며 해당 프로젝트에는 독일, 프랑스, 영국 등 유럽 철도 강국의 기업들이 참여하였다. 록히드 마틴은 자체 솔루션의 일부를 외부에 공개하고 기능 개선을 위해 외부의 리소스를 활용하기 위해 공개SW 개발 방식을 도입하였다. 나사는 미국의 열린 정부 정책에 동조하여 자체 프로젝트의 일부를 외부에 공개하고 있으며 SW 개발 목적에 따라 공개 범위를 다양하게 함으로써 공개SW 개발 방식을 적절히 활용하고 있다.

이 연구에서 수행한 안전 산업 분야에서 공개SW 활용 조사 범위가 자동차, 철도, 항공 분야에 한정되어 있고 인터넷 검색을 통한 문헌 조사라는 한계점이 있다. 그럼에도 불구하고 이들 분야에서 이미 공개SW 개발 방식이 활용되고 있는 것을 보면 공개SW 개발 방식이 다양한 분야의 SW개발에 활용되고 있는 것으로 추정할 수 있다. 공개SW 개발 방식의 장점은 SW산업만이 아닌 전 산업에서 활용될 수 있으리라 생각된다. 또한, OpenETCS와 나사의 사례를 보면 공공 목적의 프로젝트에서도 충분히 공개SW 개발 방식이 활용될 수 있으리라 보인다. 이는 과거와 달리 SW개발 프로젝트의 규모가 커졌기 때문에 여러 회사들이 참여하는 프로젝트의 경우 공개 범위에서 차이가 있을 뿐 공개SW 개발 방식과 유사하기 때문이라 생각된다.

제4장 국내 소프트웨어안전 분야 공개소프트웨어 현황

자동차 분야를 중심으로 해외에서는 안전 산업 분야에서도 공개SW 개발 방식을 활용하는 사례가 늘어나고 있다. 이는 공개SW 개발 방식의 장점인 비용 절감과 기술 혁신의 가속화 때문이다. 하지만 기존 공개SW 현황 조사들은 리눅스, 데이터베이스, 웹 서버 등의 기존 SW 분야를 대상으로 한 조사들은 있었지만 안전 분야라고 할 수 있는 자동차, 철도, 항공 등의 제조업 분야에서의 조사들은 전무한 실정이다.

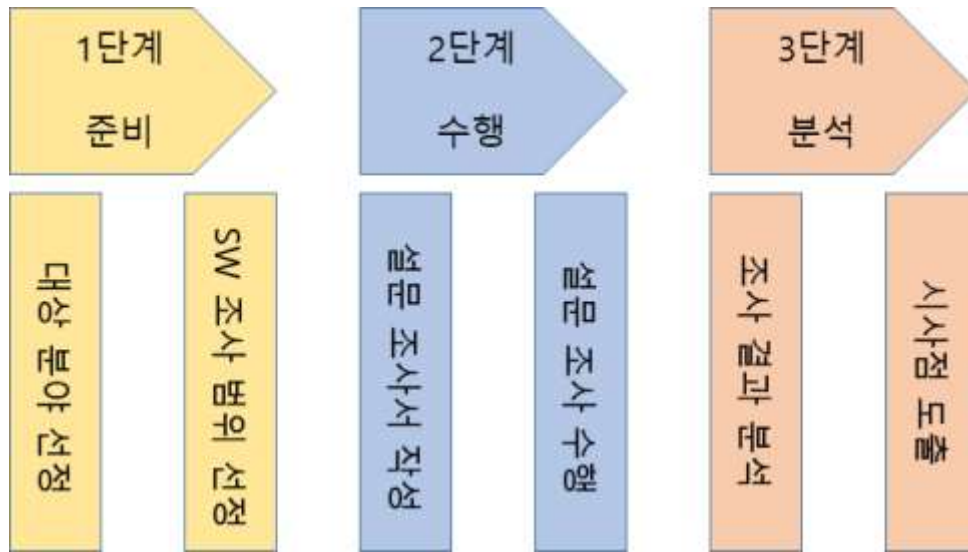
따라서 본 조사는 본격적인 전수 조사에 앞선 탐색 조사 형태로 SW안전 분야에서의 공개 SW 활용 현황에 대한 파악하기 위한 목적으로 안전 분야 기업들의 공개SW 사용 현황뿐만 아니라 공개SW에 대한 인식, 이해도 및 활용 등 실제 현황에 대한 설문 조사를 실시하였다. 조사 결과에 대한 심층 분석을 통해 국내 안전 분야의 공개SW 활성화를 위한 시사점 및 정책적 제언을 통해 향후 공개SW 활성화를 위한 정책 마련의 기초 자료를 도출하고자 한다.

제1절 설문 조사 수행 방법

국내 안전 분야 공개SW 현황 조사를 위해 구조화된 설문지를 바탕으로 국내 안전 분야 기업에 종사하는 현업 담당자들을 대상으로 설문조사를 수행하였다. 안전 분야의 공개SW 현황 조사는 이전 사례가 없는 조사이기에 현황 파악을 위한 탐색 조사로써 국내 안전 분야의 일부 업체를 대상으로 [그림 4-1]처럼 설문 조사를 수행하였다.

우선 1 단계에서는 설문 조사 준비 단계로 대상 분야 선정을 위하여 설문 조사 대상 분야와 대상 업체 수 결정하고 조사할 SW 범주 및 SW 현황 조사 범위를 결정하기 위하여 안전 분야의 SW 범주 및 조사 범위를 선정하였다. 그리고 2단계에서는 설문 조사 수행을 위해서 1단계에서 설정한 조사 범주 및 범위에 맞는 설문 조사 질의서를 작성하고 선정된 조사 대상 업체를 선정 후 설문 조사를 진행하였다. 마지막으로 3단계에서는 조사된 설문 결과를 바탕으로 SW 범주별 비교 분석 및 공개SW 활용 현황에 대한 분석을 진행하였고 이를 바탕으로 국내 SW안전 분야 공개SW 활용 활성화를 위한 시사점을 도출하였다.

[그림 4-1] 현황 조사 수행 절차



<표 4-1> 안전 관련 주요 산업

구분	주요 산업 분야	상세 산업
주요 국가기반 시설	에너지	전기
		석유
		가스
	교통수송	철도 (지하철)
		항공
		항만
		도로
	보건의료	의료서비스
		혈액
민간 부문 주요 산업	원자력	
	방위산업(무기체계)	
	석유화학	
	1차금속제조	
	기계장치제조	엔진
		터빈
	전기 장비, 기구 및 구성요소 교통수단	항공우주제품
		전기 장비
		자동차

조사 대상 분야 선정에 있어 <표 4-1> 다양한 안전 분야⁹¹⁾ 중 국내 산업의 영향력 측면에

91) 산업 전반에 걸쳐 초연결, 지능화, 무인화로 인해 SW의 비중이 점차 증가하고 있으며, 제조업이 주를 이루던 안전

서 교통이라는 주제를 통해 ‘자동차, 철도, 항공’을 선정하고 국내 항공 산업의 활성화 정도를 고려해 항공과 국방을 하나의 분야로 선정하였다. 그리고 대상 분야별 기업의 규모에 관계없이 10개의 업체를 선정하였다.

안전 분야의 SW 범주 및 설문 조사의 범위를 결정하기 위해 SW 안전성을 확보하기 위해 SW 개발 과정에서 수행해야 하는 SW 용도에 대해 <표 4-2>처럼 정리하였다. SW안전성을 위해서는 SW안전 활동, 방법, 목표 및 목적 등이 포함된 표준을 준수해서 SW를 개발해야 하기 때문에 SW 안전 관련 표준의 각 프로세스 단계에는 목적, 입력/출력물, 활동 내용 및 목표 등이 기술되어 있다. 따라서 안전 분야에서 사용하는 SW는 <표 4-2>에서 정리하였듯이 개발 프로세스 단계별로 활용되는 SW, 개발 프로세스 지원 SW, 실제 공급하는 시스템에 포함된 제어 SW로 구분할 수 있다.

<표 4-2> 개발에서 활용되는 SW 용도

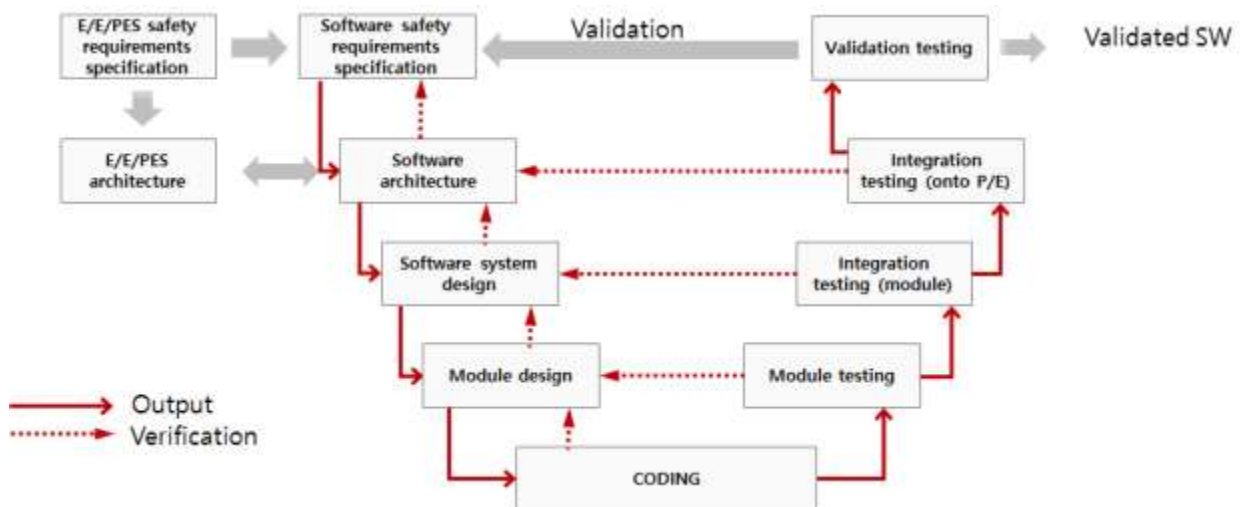
구분	SW 용도
개발 프로세스 단계별	안전성 분석
	요구사항 관리
	설계
	구현
	테스트 (정적 분석, 단위/통합 시험, 시스템 시험 등)
개발 프로세스 지원	형상관리, 프로젝트 관리, 이슈/위험 관리 등
제어 SW	시스템의 지능화, 무인화 등

개발 프로세스 단계별 SW는 [그림 4-2] 같은 SW안전성 확보를 위한 IEC 61508의 개발 절차에 따라 안전성 분석에서 테스트까지 크게 5단계로 구분할 수 있다. 테스트는 세부적으로 정적 분석, 단위/통합 시험, 시스템 시험으로 구분 가능하다. 안전성 분석 활동은 안전 분야 업체들이 보편적으로 수행하는 활동은 아닌 수요처에서 수행하는 활동이다. 개발 프로세스 지원 SW로는 ALM(Application Lifecycle Management)에 포함된 프로젝트 관리, 형상 관리, 이슈/위험 관리 도구 등의 다양한 SW가 사용되고 있다. 또한, 기존 기계, 장치 중심의 시스템들의 지능화, 무인화 등으로 인해 안전 분야 업체들이 개발 및 생산하는 제품에도 제어

산업분야에서도 SW가 HW 비중을 넘어서고 있다. 이로 인해 안전 산업의 많은 분야들이 SW 안전에 대한 활동, 표준, 규제가 생성되고 있으며, 전 산업에 공통적으로 적용 가능한 IEC 61508을 기반으로 각 산업 분야별 안전 표준을 제정하여 준수하고 있다.

SW를 사용하고 있다. 이에 이번 설문 조사에서는 개발 프로세스 단계별 지원 SW, 개발 프로세스 지원 SW, 제어 SW를 대상으로 SW 범주를 한정하고 이 중 안전성 분석 활동은 안전 분야 업체들의 보편적인 활동이 아니기 때문에 안전성 분석 지원 SW는 조사 대상에서 제외하였다.

[그림 4-2] IEC 61508 SW 개발 모델



출처: 네이버 블로그, IT 융합 산업의 HW 및 SW 안전 표준 IEC 610508 개요 참조

안전 분야의 SW 종류 및 SW현황 조사 범위를 기반으로 설문 조사 질의서를 작성하였으며, 조사 항목은 산업 분야별, 활용 SW별, 공개SW를 바탕으로 현황 조사 설문지에 대한 분석, 공개SW 활용 관련 데이터 수집을 통해 기본정보, 국내/외 제도, 규제 지침 및 사내지침, 사용현황, 공개SW 이해도, 공개SW 활용현황 5개로 구분하여 [그림 4-3]처럼 설문 조사 분야를 구성하였다. 설문 조사 항목은 정량적으로 측정하여 통계화할 수 있는 내용은 객관식으로 구성하고, 대상 기업/개인의 입장, 특징/특성 등이 포함되어 응답이 다양하게 도출될 수 있는 사항은 주관식으로 구성해 상세한 답변이 도출될 수 있도록 구성하였다.

산업 분야별 조사 항목에 대해서는 기업 및 설문 담당자의 기본 정보와 각 분야의 국내/외 SW 규제 및 지침 여부 및 SW 관련 내용을 파악할 수 있도록 준비하였다. SW 관련하여 SW 사용 현황, 공개 SW 이해도 및 공개SW 활용 현황으로 질의 항목을 재구성하여 질의서를 작성하였다. SW 사용 현황은 SW안전성(신뢰성) 활동 시 SW 사용 여부 및 대상 SW를 파악하기 위해 질의 항목을 작성하였으며, 공개 SW 이해도에서는 공개SW 개념, 라이선스 정책의 이해도 및 공개SW 관련 커뮤니티나, 사내 활동 내용에 대한 질의서를 작성하였다.

공개SW 활용 현황 측면에서는 사용 여부, 목적 및 방법과, 공개SW에 대한 장/단점 및 활용 확대의 의견을 조사할 수 있도록 질의서를 준비하였다. <표 4-3>은 설문 조사 내용에 대한 요약이다.

〈표 4-3〉 설문 조사 항목 요약

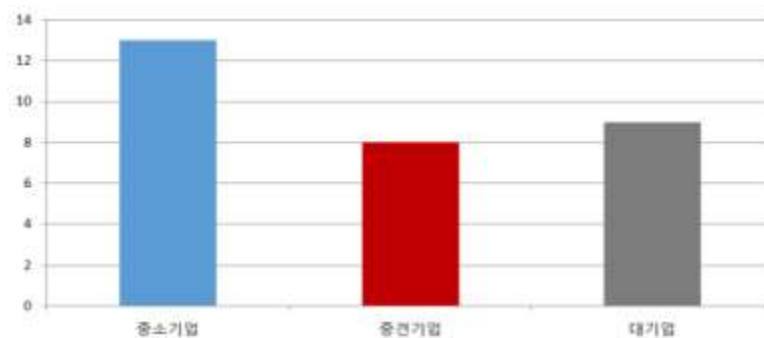
No	구분	항목
1	기본정보	- 기업 규모 - 종사 분야 - 종사 기간 - 담당 업무 - SW 개발 언어 및 유형
2	국내/외 제도, 규제 지침 및 사내지침	- 표준, 규제, 사내지침 존재 유무 - 준수 항목 내용 - 공개SW 부서 존재 유무 - 공개SW에 대한 기업 입장
3	SW 사용현황	- 사내 안전성/신뢰성 활동 유무 - SW 사용 현황 및 관련 내용
4	공개SW 이해도	- 공개SW 이해도 - 공개SW 관련 경험 - 공개SW 라이선스 이해도 - 공개SW 관련 활동 - 공개SW 관리 활동 필요성
5	공개SW 활용 현황	- 공개SW 사용 여부 - 공개SW 사용 방식 및 용도 - 공개SW 사용 이유 - 사용 공개SW 라이선스 정보 - 공개SW 활성화를 위한 요소 - 공개SW 홍보활동의 필요 유무 - 공개SW 활용을 위한 지원 요구사항

제2절 설문 조사 결과

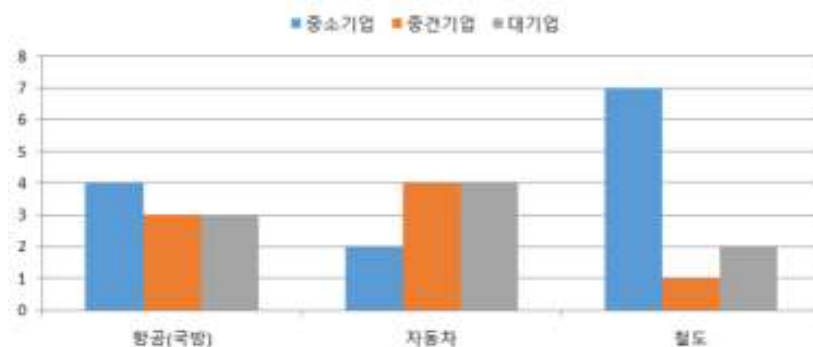
1. 조사 대상 기본 정보

국내 안전 분야의 공개SW 현황을 파악하기 위해 자동차, 철도, 항공(국방) 분야의 각 10개 업체를 대상으로 설문 인터뷰 조사를 실시하였다. 조사 대상 기업들의 규모는 [그림 4-3]처럼 중소기업 13개 업체, 중견기업 8개 업체, 대기업 9개 업체 대상으로 진행하였다. 각 분야별로는 [그림 4-4]같이 항공(국방) 분야는 중소기업 4개, 중견기업 3개, 대기업 3개 업체, 자동차 분야에서 중소기업 2개, 중견기업 4개, 대기업 4개 업체, 철도 분야에서 중소기업 7개, 중견기업 1개, 대기업 2개 업체를 대상으로 조사를 실시하였다.

[그림 4-3] 조사 대상 업체 규모



[그림 4-4] 분야별 조사 대상 업체 규모

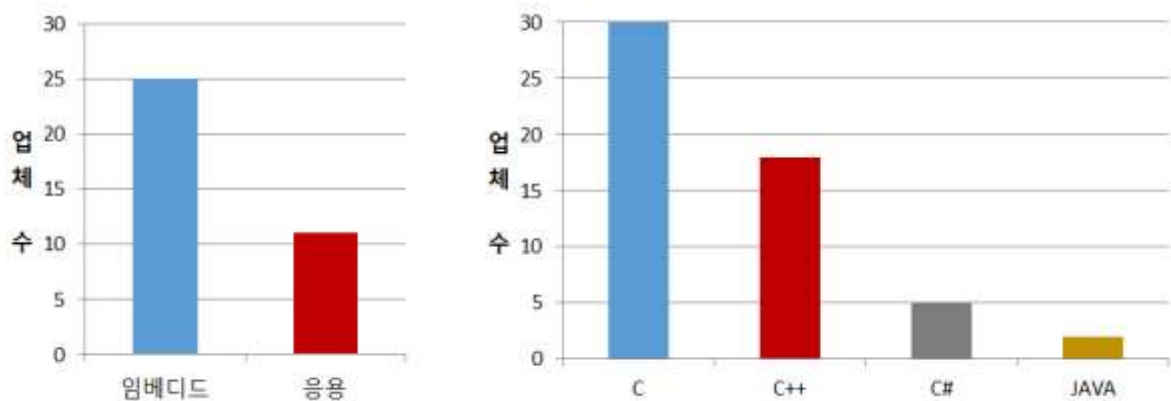


[그림 4-5]는 기업들이 개발하는 SW의 유형과 사용 개발 언어에 대한 조사 결과를 나타낸 것으로 대상 기업 중 25개 기업이 임베디드 SW를 개발하고 있으며, 11개 기업이 응용 SW

를 개발하는 것으로 조사되었으며 이 중 6개 업체는 임베디드 SW와 응용 SW를 모두 개발하는 것으로 조사되었다. 안전 분야의 특성상 응용SW보다 임베디드 SW 형태로 제품에 적용되는 개발이 많다는 것을 알 수 있다.

개발 언어 측면에서는 모든 업체에서 C 언어를 기본적으로 사용하고 있는 것으로 조사되었고, Java 언어의 사용 비율이 가장 낮게 나타났다. 그 이유는 임베디드 시스템⁹²⁾에 적용되는 SW개발 업체가 많았기 때문에 임베디드SW 개발에 널리 활용되는 C/C++ 언어 사용 비중이 높게 나타난 것으로 해석할 수 있으며 Java의 경우는 응용 프로그램 개발이 많기 때문에 이번 조사에서 적게 나온 것으로 해석할 수 있다.

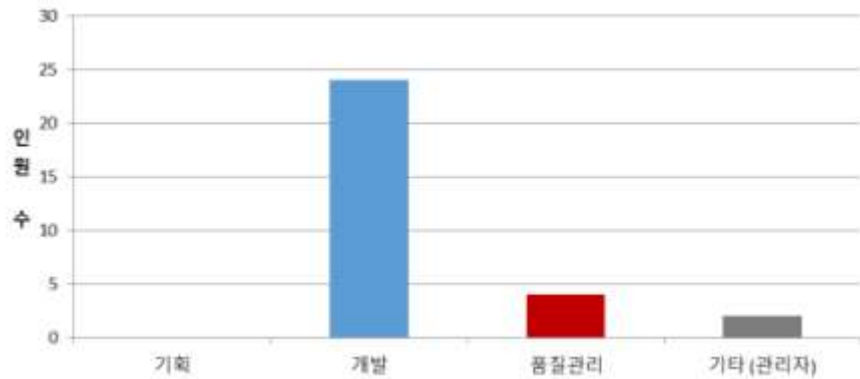
[그림 4-5] 설문 대상 업체의 개발 SW 유형 및 사용 개발 언어(복수 응답)



설문 조사 대상자의 담당 업무는 [그림 4-6]처럼 개발 부서 종사자가 24명(80%)으로 가장 많았으며, 품질 담당이 4명(13.3%), 기타(개발부서 관리자) 2명(6.7%)이 응답하고 기획 업무 종사자는 선정되지 않았다. 이는 각 업체에서 SW 관련 현업에 종사하는 인력들 위주로 선정되었기 때문이다. 따라서 업체의 공개SW 현황 파악에 적절한 대상으로 확인되었으며 이번 조사가 전수조사 성격이 아닌 것을 감안하면 일부 품질 담당과 관리자들을 포함함으로써 SW안전 산업에 종사하는 다양한 업무 특성을 감안할 수 있었다.

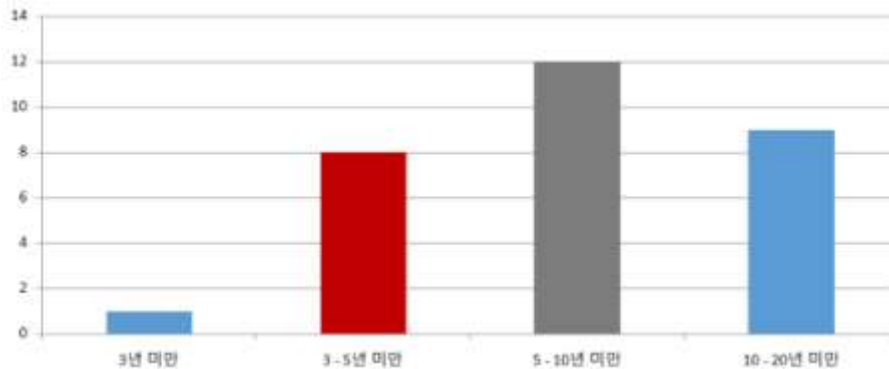
92) 임베디드 시스템 : 기계나 기타 제어가 필요한 시스템에 대해 제어를 위한 특정 기능을 수행하는 SW가 포함된 시스템

[그림 4-6] 설문 조사 대상자 담당 업무



[그림 4-7]은 인터뷰 대상자의 근무 기간을 나타낸 것이며, 5~10년 미만이 12명(40%)으로 가장 많았고, 10~20년 미만이 9명(30%), 3~5년 미만이 8명(26.7%), 3년 미만이 1명(3.3%)으로 조사되었다. 설문 대상자는 대부분이 해당 기업에서 3년 이상을 근무한 경력자들이어서 SW 안전 산업 분야의 공개SW 현황을 파악할 수 있는 인력들로 판단되며 본 설문 조사에 적합한 대상자로 확인되었다.

[그림 4-7] 설문 조사 대상자 근무 기간

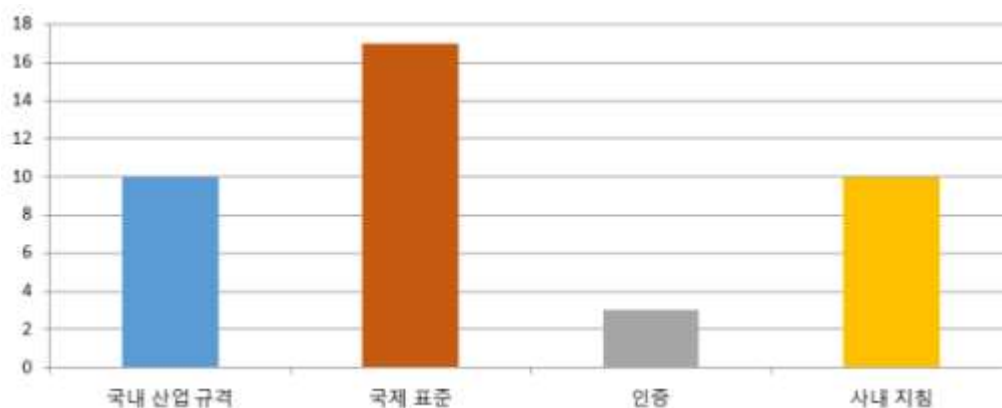


2. 분야별 SW 관련 지침 현황

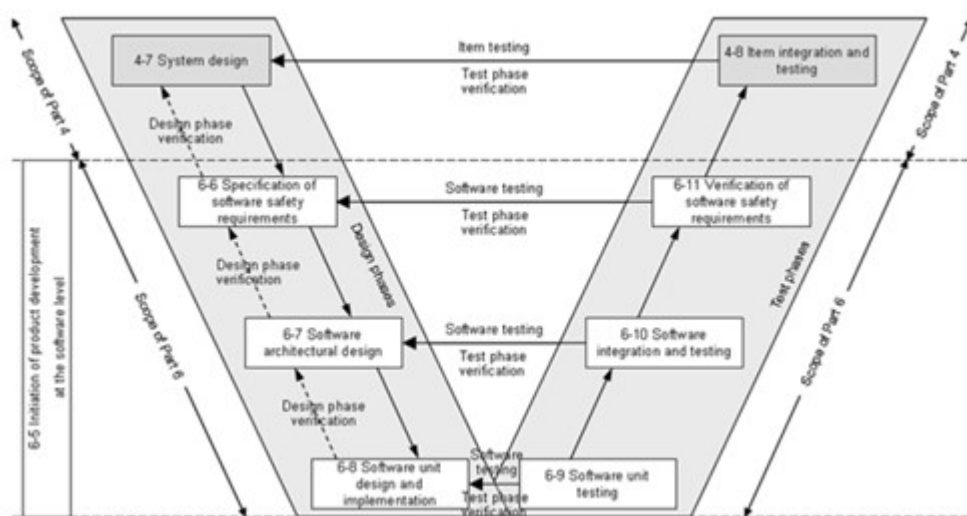
안전 분야의 기업을 대상으로 국내/외 사업 수행 시 준수해야 하는 지침(표준, 규제, 사내 지침 등)의 존재 여부에 대해 조사하였다. [그림 4-8]은 이에 대한 결과로 국제 표준을 준수하는 기업이 17곳으로 가장 많았으며, 사내 지침과 국내 산업 규격이 각 10개 기업, 인증(프

로세스 지침)이 3개 기업으로 조사되었다. 관련된 지침이 없는 5개 기업들을 제외한 다수의 기업들이 안전 관련 표준 및 지침을 따르고 있는 것으로 파악되었다.

[그림 4-8] 안전 분야 적용 표준 및 지침 준수 현황 (복수 응답)



[그림 4-9] ISO 26262의 SW 개발 프로세스



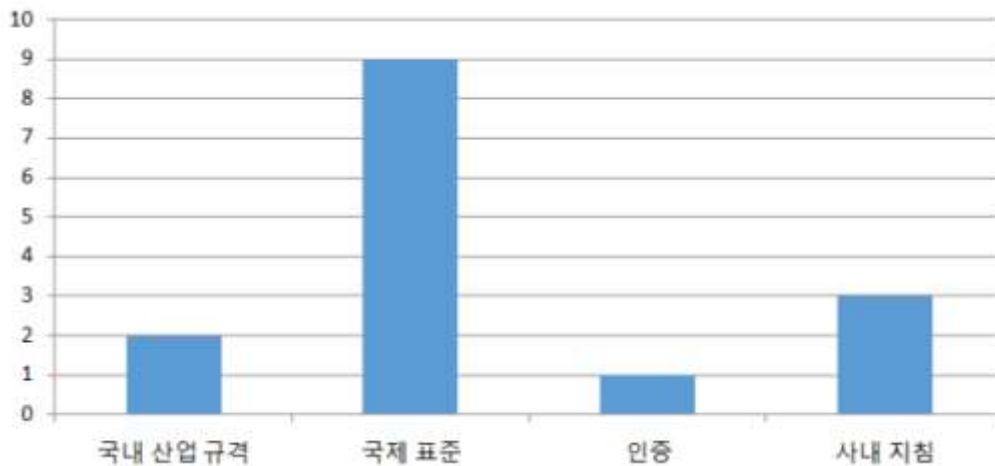
출처: SW공학포털, ISO 26262 자동차 기능 안전을 위한 MISRA 코딩 규칙 적용 방법

자동차 분야는 대표적인 안전 분야로 자동차 기능안전 국제 규격인 ISO 26262⁹³⁾가 있다. 자동차 분야의 기업들은 자동차 시스템 개발 시 기능안전성 확보를 위해 ISO 26262를 기반으로 개발하거나, ISO 26262을 기반으로 자체 지침을 수립하고 이를 준수하여 개발을 진행

93) ISO 26262 : “Road vehicles – Functional safety“, 자동차 기능안전 국제 규격

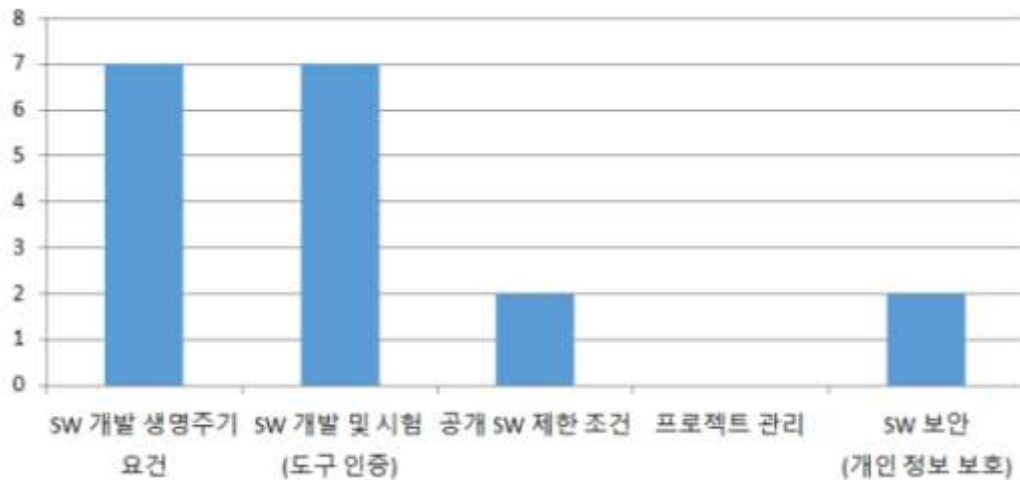
하고 있다. [그림 4-9]는 ISO 26262에서 제시하는 SW 개발 프로세스를 나타낸 것이며, 표준에는 각 단계별 수행 활동, 입/출력물, 완료 기준 등이 제시되어 있다.

[그림 4-10] 자동차 분야 적용 표준 및 지침 (복수 응답)



자동차 분야 기업들이 적용하는 표준 및 지침에 대해 조사한 결과는 [그림 4-10]처럼 자동차 안전 국제 표준인 ISO 26262를 준수하는 기업이 9곳으로 가장 많았으며, 국내 산업 규격으로는 국내 자동차 OEM사의 SW 지침을 따르는 기업 2곳과 A-SPICE 프로세스 인증과 사내 자체적인 지침을 따르는 기업이 각각 1개, 3개로 나타났다. 결과적으로 모든 자동차 분야 기업들은 ISO 26262 자체를 준수하거나 아니면 이에 파생된 SW 지침을 따르는 것으로 파악되었다. 그 이유는 자동차 산업 전반적으로 SW안전에 대하여 고려하고 있다는 것을 파악할 수 있다.

[그림 4-11] 자동차 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)



[그림 4-11]은 자동차 분야에서 현재 적용하고 있는 표준, 지침 등에 포함된 SW 관련 내용을 조사한 결과로, SW 개발 및 시험과 SW 개발 생명주기 요건이 포함되어 있다는 기업이 각 7개로 가장 많았으며, 공개 SW 제한 조건과 SW 보안이 포함되어 있다고 대답한 기업이 각 2개로 조사되었다. 공개SW 관련 제한 조건으로는 GPL 계열 라이선스에 따른 코드 공개 조항을 회피하기 위한 지침으로 파악되었다.

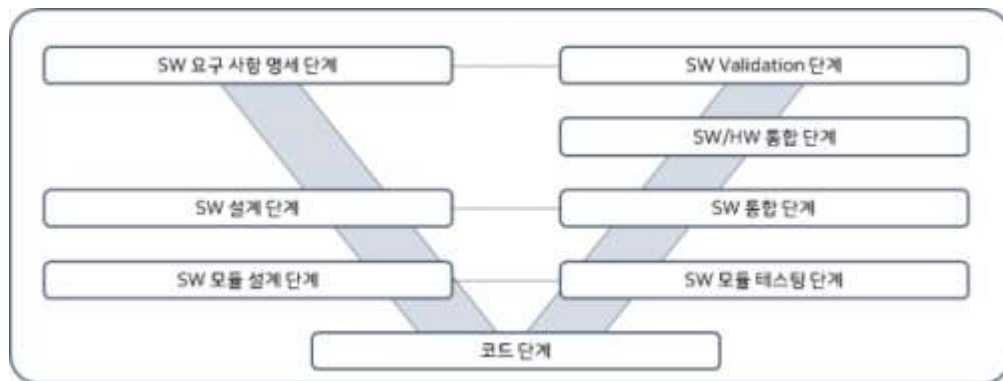
철도 분야의 대표적인 SW 지침은 국제 표준인 IEC 62279⁹⁴⁾이며, 국내에서는 2012년 개정 철도안전법에 의하여 SW 개발 및 검증 활동을 철도차량 기술기준에 포함하여 SW 형식 승인 제도를 제정하여 고시⁹⁵⁾로 명시하였다. 약 5년간 제도 운영의 유예 기간을 두었으나, 2017년 5월 형식승인 유예 기간이 만료되어 2018년에는 SW 형식 승인 제도를 운영 중에 있다. 또한, 한국철도시설공단의 철도 사업에서는 SW를 포함한 RAMS⁹⁶⁾ 활동이 사업 요건에 포함되어 있다. [그림 4-12]는 IEC 62279의 철도 SW 개발 프로세스를 나타낸 것이다.

94) IEC 62279 : “Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems”, 철도에서 제어와 방호 응용 분야에 사용하는 프로그램 가능한 전자 시스템의 개발에서 요구되는 기술적인 요구 사항과 절차를 기술

95) SW 형식 승인 제도 : 철도차량기술기준 part51-3.2.7에 철도 SW에 대한 기준 명시

96) RAMS : Reliability(신뢰성), Availability(가용성), Maintainability(유지보수성), Safety(안전성)의 약어로써, 4가지 특성을 확보하기 위한 HW/SW 개발 및 품질 활동을 수행해야 함

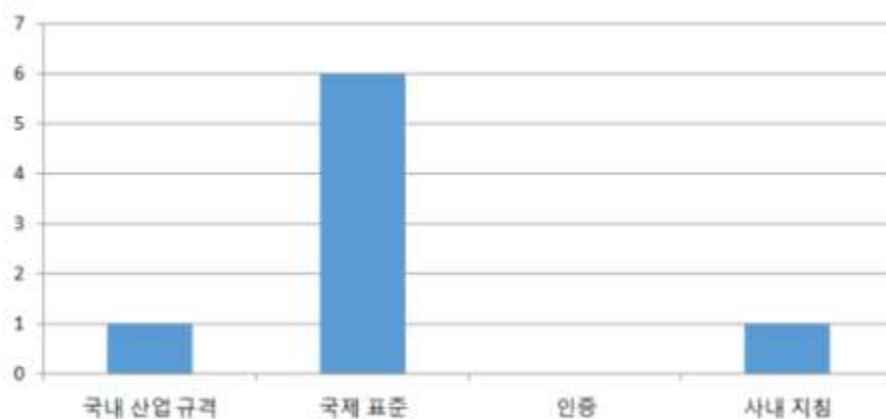
[그림 4-12] IEC 62279의 철도 SW 개발 프로세스



출처: 슈어소프트, IEC 62279 기반 소프트웨어 Safety 프로세스 구축 사례 재구성

철도 분야 기업들이 적용하는 표준이나 지침은 [그림 4-13]처럼 국제 표준인 EN 50128이나 IEC 62279를 준수하는 기업이 6개로 가장 많았으며, 이들 6개의 기업 중에서 추가적인 자체 지침과 철도 형식 승인 제도의 규제를 따르는 기업도 각 1개씩 있었다. 그리고 지침을 따르지 않는 4개의 기업들은 규모가 작은 기업들로 자체적으로 SW안전 활동을 수행할 여력이 부족한 기업들이었다.

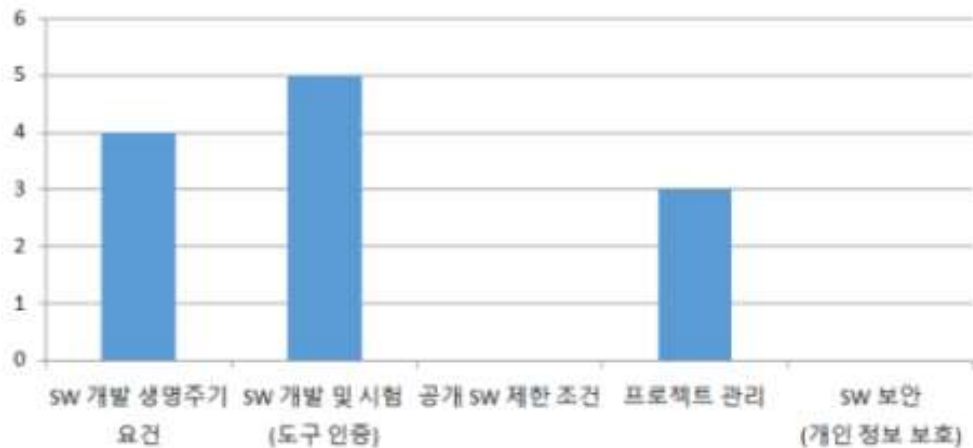
[그림 4-13] 철도 분야 적용 표준 및 지침 (복수 응답)



[그림 4-14]는 철도 분야에서 현재 적용하고 있는 표준, 지침 등에 포함된 SW 관련 내용을 조사한 결과로 철도 분야에서는 SW 개발 및 시험 내용을 포함하고 있다는 기업이 5개로 가장 많았으며, SW 개발 생명주기 요건과 프로젝트 관리가 각각 4개, 3개 기업 순으로 조사되었고, 공개SW 제한 조건과 SW 보안이 포함되어 있다고 답변한 기업은 없었다. 특히

공개SW 관련하여서는 사용 제한보다는 활용 가능한 공개SW 라이선스 관리 및 사용 중인 공개SW 관리와 같은 적극적인 정책을 펴고 있었다.

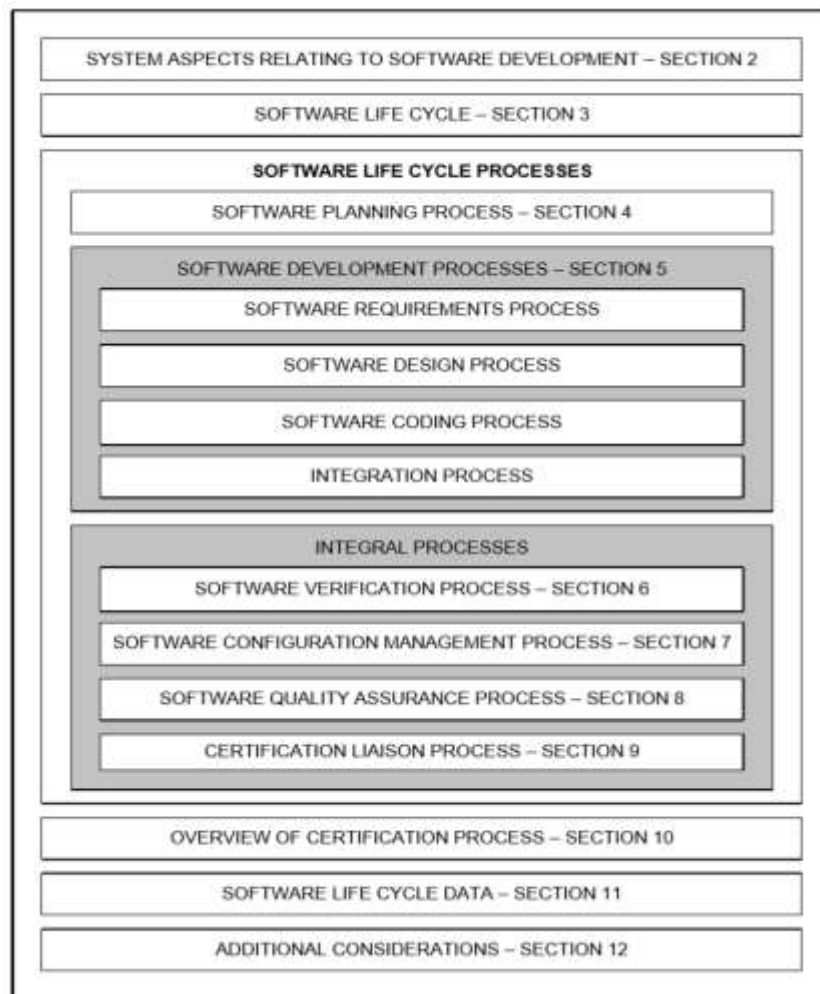
[그림 4-14] 철도 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)



항공 분야는 항공기에 탑재되는 시스템 및 장치의 SW안전 확보를 위한 요구사항을 [그림 4-15]처럼 DO-178C⁹⁷⁾의 항공 SW 개발 프로세스에서 명시하고 있다. 특히 항공기 사고는 자동차, 철도 사고에 비하여 인명 피해의 규모와 심각도가 클 수 있기 때문에 타 분야 보다 더욱 철저하게 관리하고 있는 것으로 알고 있다. 그리고 미국 연방 항공국(FAA), 유럽 항공 안전국(EASA), 민간 항공 관리국(CAA)등의 다양한 항공 관련 요구사항을 만족하기 위해서는 DO-178C 표준을 준수하여 SW안전을 보장할 수 있는 SW 개발을 진행해야 한다.

97) DO-178C : “Software Considerations in Airborne Systems and Equipment Certification” 항공기에 탑재되는 시스템과 장치 인증을 위한 SW 요구사항을 기술

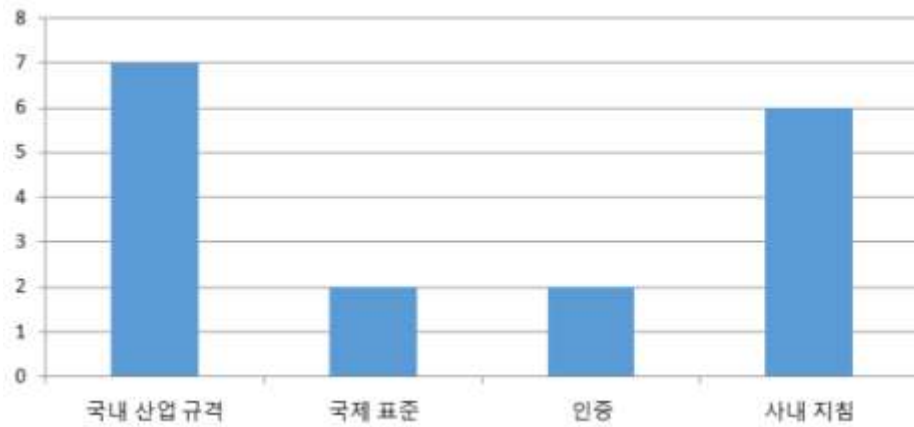
[그림 4-15] DO-178C의 항공 SW 개발 프로세스



출처: 네이버 블로그, DO-178C 소개와 개요 재구성

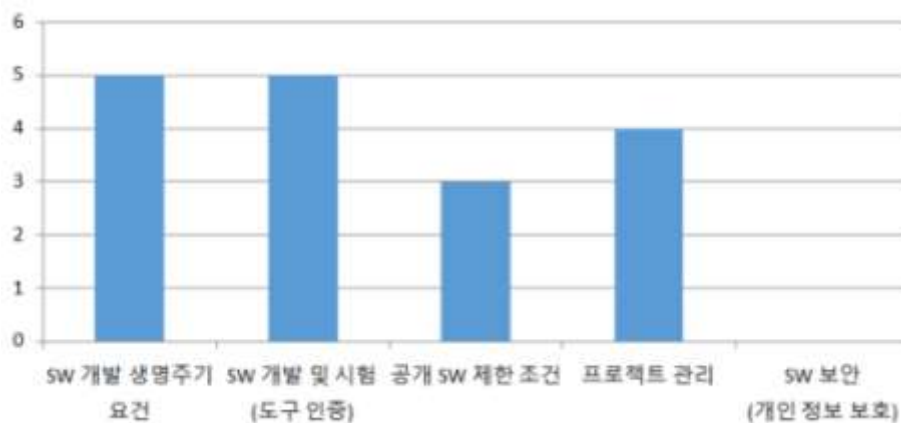
항공(국방) 분야 기업들이 적용하는 표준 및 지침 현황은 [그림 4-16]에서 보여준다. 이번 조사에서는 국내 산업 규격인 방위사업청의 “무기체계 SW 개발 및 관리 매뉴얼”을 준수하는 기업이 7개로 가장 많았고, 2개 기업은 항공과 국방 사업을 겸하는 기업으로 항공 SW 국제 표준인 DO-178B/C를 추가적으로 따르는 것으로 조사되었다. 그 이유는 국내 항공 관련 기업들이 부족하여 국방 관련 기업들이 본 설문 조사에서 많이 참여했기 때문이다. 그리고 CMMI, SP 프로세스 인증을 준수하는 기업 2개와 자체적인 SW 지침 따르는 기업도 6개가 있었다. 관련 지침을 따르지 않는 기업은 1개를 제외한 대부분의 기업들이 SW안전을 위해 국제 표준 또는 자체적인 지침을 따르고 있는 것으로 파악되었다.

[그림 4-16] 항공(국방) 분야 적용 표준 및 지침 (복수 응답)



[그림 4-17]은 항공(국방) 분야에서 현재 적용하고 있는 표준, 지침 등에 포함된 SW 관련 내용을 조사한 결과로 SW 개발 생명주기 요건을 응답한 기업이 5개, SW 개발 및 시험을 응답한 기업이 5개, 공개 SW 제한 조건을 응답한 기업이 3개, 프로젝트 관리를 응답한 기업이 4개로 나왔다. 그리고 표준 및 지침에 SW 보안 내용은 포함되지 않은 것으로 조사되었다.

[그림 4-17] 항공(국방) 분야 적용 표준, 지침 내 SW 관련 내용 (복수 응답)



특히 항공 분야와 같이 조사된 국방 분야는 다른 분야와 달리 방위사업청에서 “무기체계 SW 개발 및 관리 매뉴얼⁹⁸⁾”을 제정하여 [그림 4-18]처럼 무기체계 SW 개발 프로세스에

따라 각 개발 단계에서 준수해야 할 수행 활동과 품질 기준을 명시하고 있다. 이 매뉴얼에는 [그림 4-19]처럼, 공개SW에 대해 소스코드 공개 의무가 있는 공개 SW의 사용 금지 내용을 포함하고 있다. 이는 GPL 계열의 소스코드 공개 의무가 있는 공개SW 활용을 원칙적으로 금지하여 향후 발생할 수 있는 위험을 대비하기 위해서이다. 하지만, 소스 코드 공개 조항이 없는 라이선스의 활용에 대해서는 명확한 조항은 없는 현실이다.

[그림 4-18] 무기체계 SW 획득 절차



출처: 무기체계 SW 개발 및 관리 매뉴얼

[그림 4-19] 무기체계 SW 개발 및 관리 매뉴얼의 공개SW 관련 지침

7. 지식재산권

가. 사업관리부서장은 정부투자과 공동투자 연구개발 사업에서 「방위사업법시행령」 제 35조 제2항에 의하여 무기체계 소프트웨어 지식재산권의 소유관계를 정한 후 이를 제안요청서, 계약서 및 체계개발동의서에 명시해야 한다.

나. 사업관리부서장은 지식재산권 분쟁을 예방하기 위하여 제안업체가 제안서 작성시 상용소프트웨어, 저작권 등록 소프트웨어, 비공개 소프트웨어 등의 원천기술을 식별하고 확보계획을 명시하도록 제안요청서를 작성 하여야 한다.

다. 사업관리부서장은 사업과 관련된 소프트웨어 (실행파일, 소스코드 등 포함) 지식 재산(특허 또는 저작권 등)의 소유권 또는 실시(사용)권을 확보하기 위하여 계약 특수조건 등에 명시 하여야 한다.

라. 사업관리부서장은 소프트웨어 개발단계에서 소프트웨어(공개 소프트웨어 포함)가 지식재산권 및 라이선스에 저촉되지 않도록 관리한다.

마. 연구개발주관기관은 연구개발 시 소스코드 공개 의무가 있는 공개 소프트웨어 (Open Source)를 사용해서는 안된다.

바. 연구개발주관기관은 소프트웨어에 대한 지식재산권 및 라이선스 저촉여부를 검토 하여 소프트웨어통합시험결과서에 포함한다.

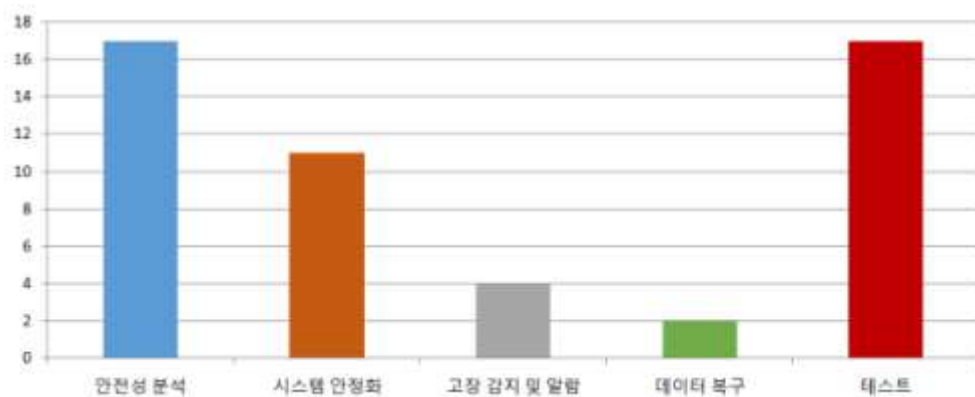
사. 기술지원기관은 소프트웨어통합시험결과서를 토대로 지식재산권 및 라이선스 저촉 여부를 확인한다.

출처: 무기체계 SW 개발 및 관리 매뉴얼

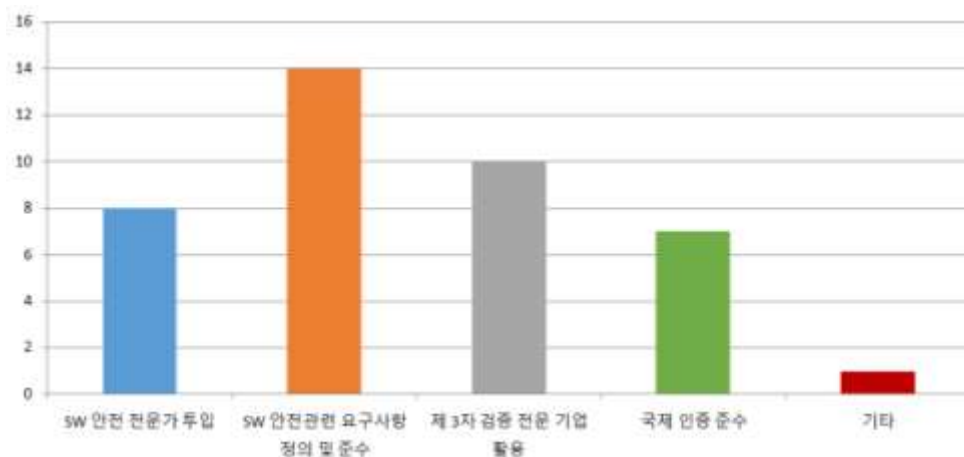
98) 무기체계 SW 개발 및 관리 매뉴얼 : 방위력개선사업으로 획득되는 소프트웨어의 체계적인 개발 및 관리를 위한 프로세스와 산출물 작성표준 등을 규정

안전 분야 기업에서 SW 개발 시 안전성(신뢰성) 확보를 위한 필요사항에 대한 질문에는 5가지의 응답이 있었으며, 중복 의견을 제출한 기업도 있었다. [그림 4-20]은 이에 대한 결과로 안전성 분석과 테스트가 각각 17개 기업으로 가장 높은 응답률을 보여 안전 분야 기업들이 안전성(신뢰성) 확보를 위해 중요하게 고려하는 사항으로 파악되었다. 그 외 시스템 안정화 11개 기업, 고장 감지 및 알람 4개 기업, 데이터 복구 2개 기업의 의견이 조사되었다.

[그림 4-20] SW 안전성(신뢰성) 확보를 위한 필요사항 (복수 응답)



[그림 4-21] SW 안전성(신뢰성) 확보를 위한 수행 활동 (복수 응답)



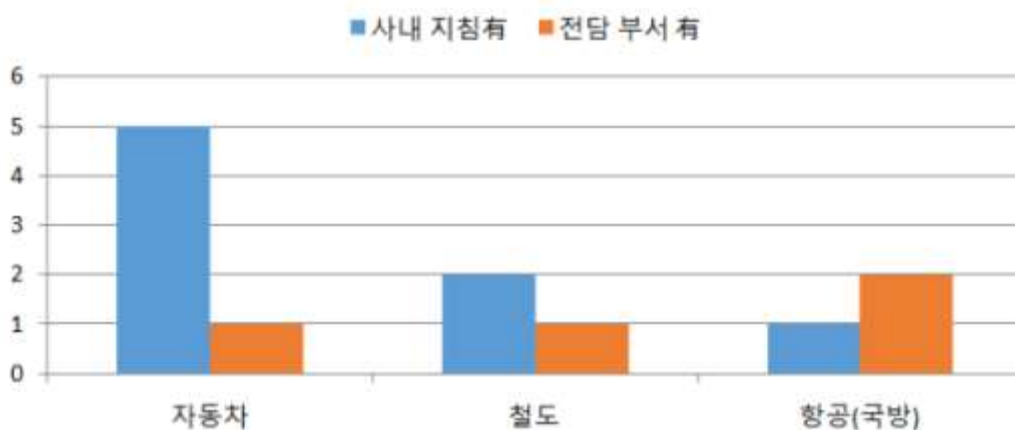
안전 분야 기업들이 안전성(신뢰성) 확보를 위해 수행 활동에 대해 조사를 진행하였으며 [그림 4-22]는 기업들의 복수 응답을 포함한 결과를 나타낸 것이다. 결과를 보면, SW 안전성 관련 수행 활동으로 SW 안전 관련 요구사항 정의 및 준수한다는 답변이 14개 기업으로

가장 많았으며, 제3자 검증 전문 기업 활용이 10개 기업, SW 안전 전문가 투입이 8개 기업, 국제 인증 준수가 7개 기업, 기타 1개 기업의 응답을 보였다.

[그림 4-23]은 분야별 공개SW 관련 사내 지침과 전담 부서가 있는지에 대한 조사 결과로 공개SW 관련 사내 지침의 존재 여부에 대한 질문에 전체 30개 기업 중 8개 기업이 존재한다고 답변하였다. 분야별로 보면 자동차 분야가 5개 기업으로 가장 많았으며 철도와 항공(국방) 분야는 각각 2개, 1개 기업으로 나타났다. 공개SW 관련 전담부서의 존재 여부에 대한 질문에 30개 기업 중에 4개 기업이 전담부서가 있다고 답하였다.

이러한 사실들을 보면 대부분의 안전 분야 기업들 30개 중에서 22개 기업들이 관련 지침이 없으며 26개 기업들은 관련 부서가 없다는 것을 알 수 있다. 이를 토대로 안전 분야 기업들은 공개SW에 대한 적절한 관리가 수행되지 않고 있다는 것을 알 수 있으며 공개SW 사용에 대한 관리는 부서 또는 개발자 자체적으로 수행해야 하기 때문에 관리의 한계가 있을 수밖에 없다.

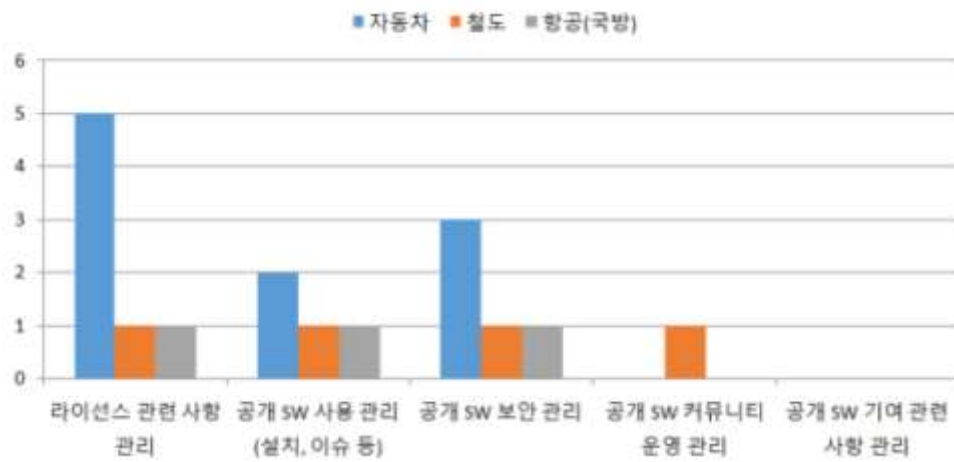
[그림 4-22] 분야별 공개SW 사내 지침 현황



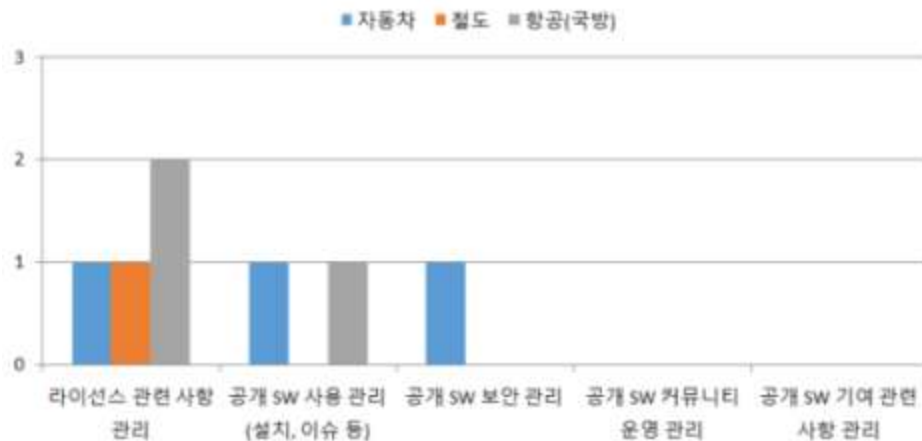
[그림 4-24]와 [그림 4-25]를 보면, 공개SW 관련 지침이 존재하는 기업의 지침 내용에는 사용 가능한 공개SW 라이선스, 공개SW 사용 규정과 보안 관련 내용들을 포함하고 있다고 답변하였으며, 공개SW 전담 부서가 있는 기업에서는 해당 부서가 공개SW 라이선스 관리와 사용 중인 공개SW 관리를 담당한다고 답변하였다. 자동차 분야의 5개 기업들의 지침에서 라이선스 관리를 하고 있고 3개의 기업에서 보안 관리를 하고 있으며 2개의 기업에서 공개SW 사용 관리를 하고 있다고 하였다. 철도 분야의 1개의 기업의 지침에서 공개SW 라이선

스, 사용 관리, 보안 관리, 커뮤니티 관리를 수행한다고 답변하였다. 항공(국방) 분야는 1개 기업에서 지침으로 공개SW 라이선스, 사용 관리, 보안 관리를 하고 있다고 응답하였다. 그리고 항공(국방) 분야의 2개 기업에서 공개SW 관련 전담 부서가 있으며 해당 부서는 “무기체계 SW 개발 및 관리 매뉴얼”의 기준에 따라 사용 금지된 라이선스의 공개SW를 포함하는지에 대한 관리를 담당한다고 답변하였다.

[그림 4-23] 분야별 공개SW 사내 지침 내용 (복수 응답)



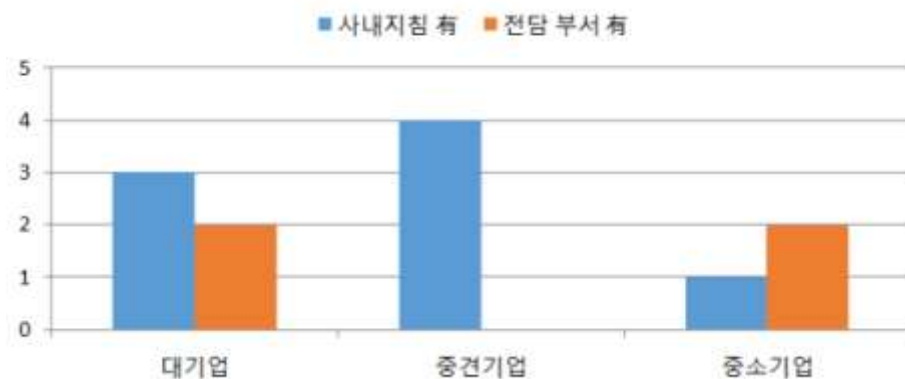
[그림 4-24] 분야별 공개SW 관련 전담 부서 활동 내용 (복수 응답)



공개SW 관련 사내 지침과 전담부서의 존재 여부에 대한 기업 규모별 결과는 [그림 4-26]에 나타내었으며, 공개SW 사내 지침이 있다는 기업은 중견기업 4개, 대기업 3개, 중소기업 1개로 나타나서, 상대적으로 일정 수준의 이상의 기업들 (대기업의 33% 또는 중견기업의 50%) 들이 공개SW에 대한 지침을 가지고 있음을 알 수 있다. 하지만 중소기업 12개 중 1개

의 기업에서만 관련 지침을 가지고 있었다. 해당 조사 결과를 보면 대체적으로 중견급 이상의 기업에서는 공개SW에 대한 위험 대비 측면에서 지침을 일부 가지고 있으나 중소기업에서는 거의 대응을 하고 있지 않다고 볼 수 있다. 그리고 전담 부서는 기업 규모와 상관없이 매우 부족한 현실임을 알 수 있다.

[그림 4-25] 기업 규모별 공개SW 관련 사내 지침 및 전담 부서 현황



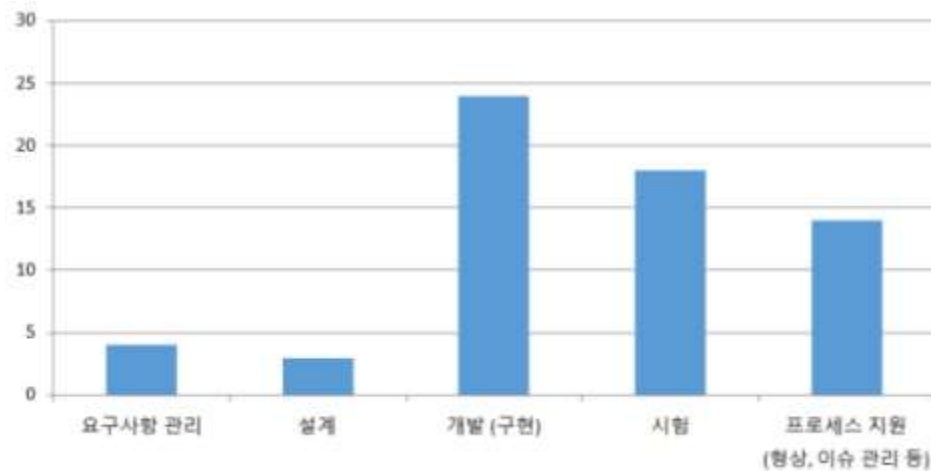
조사 결과를 볼 때, 안전 분야에서는 기업 규모나 산업 분야에 관계없이 대체적으로 기업에서 공개SW에 대한 별도의 정책을 확보하여 운영하지 않는 것으로 보이며, 일부 공개SW 관련 사내 지침이나, 전담부서가 있는 기업에서도 공개SW 라이선스 위반에 대한 관리 감독을 담당하거나, 개발 시 활용 가능한 공개SW 기반 도구 사용 현황 관리 정도를 수행하는 것으로 확인되었다. 라이선스 별로 의무사항이 다르기 때문에 공개SW 라이선스의 위반에 대한 관리가 매우 복잡하나 대부분의 기업에서는 이에 대한 전담 부서도 부재한 상황이었다.

3. 공개SW 포함한 SW 활용 현황

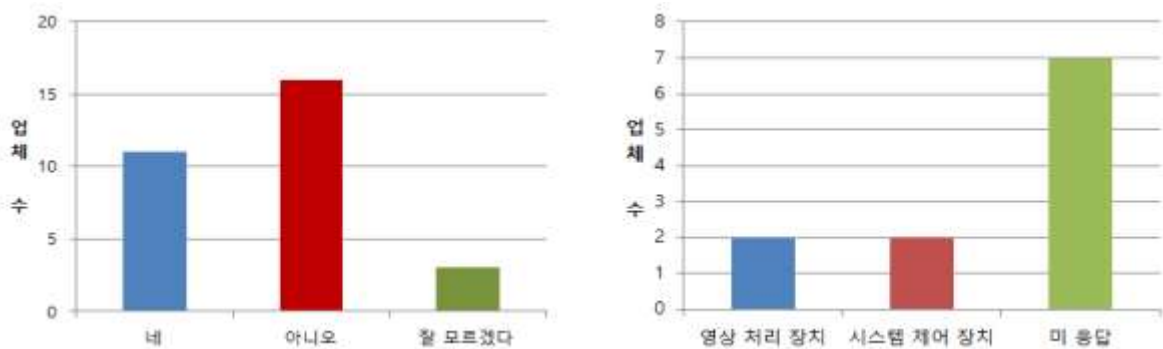
[그림 4-7]의 설문 조사 대상자의 담당 업무에서 개발 업무라고 응답한 24명의 개발자들을 대상으로 SW 활용 용도에 대해 조사를 수행하였다. [그림 4-26]은 이에 대한 결과를 나타낸 것으로 안전 분야에서는 SW 개발에서 개발(구현) 단계가 24개 기업으로 가장 많았으며, 시험 단계에서 시험 도구 SW를 사용하는 기업은 18개, 프로세스 지원 도구 SW 14개 기업으로 조사되었고, 요구사항 및 설계도구 SW를 사용하는 기업이 각각 4개와 3개의 기업

들로 조사되어 많은 기업들이 개발, 시험, 프로세스 지원을 위해 SW를 활용하고 있는 것으로 확인되었다. 해당 결과를 보면 단순 개발뿐만이 아니라 SW개발 전과정에서 SW가 활용됨을 알 수 있다.

[그림 4-26] 안전 분야에서의 SW 활용 용도 (복수 응답)



[그림 4-27] 개발 시스템에서 공개SW 사용 현황

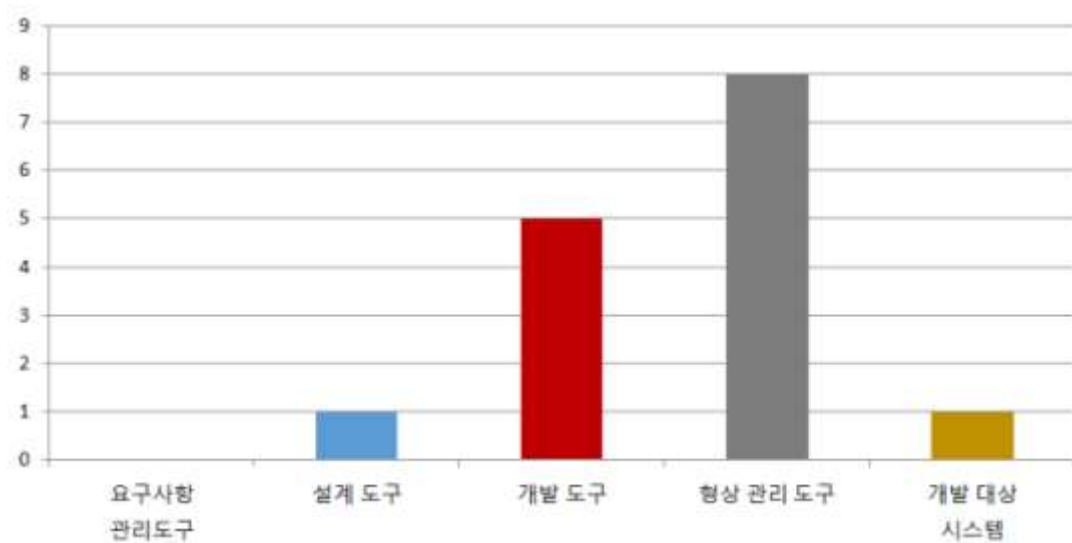


[그림 4-27]은 안전 분야 30개 기업을 대상으로 개발 시스템에서 공개SW의 사용 여부에 대해 조사 수행 결과로 11개(37%) 기업이 공개SW를 사용한다고 답하였으며, 사용하지 않는 기업이 16개(53%), 잘 모름으로 답변한 기업은 3개(10%)로 조사되었다. 공개SW를 활용하여 개발 중인 시스템은 영상처리 장치 2개 기업, 시스템 제어 장치가 2개 기업에서 답하였으며, 미 응답이 7개 기업으로 조사되었다. 이러한 사실은 이미 안전 분야 기업들도 공개SW를 활용하고 있다는 것이다. 그리고 미 응답에 대한 수치가 높게 나오는 것은 업체가 개발

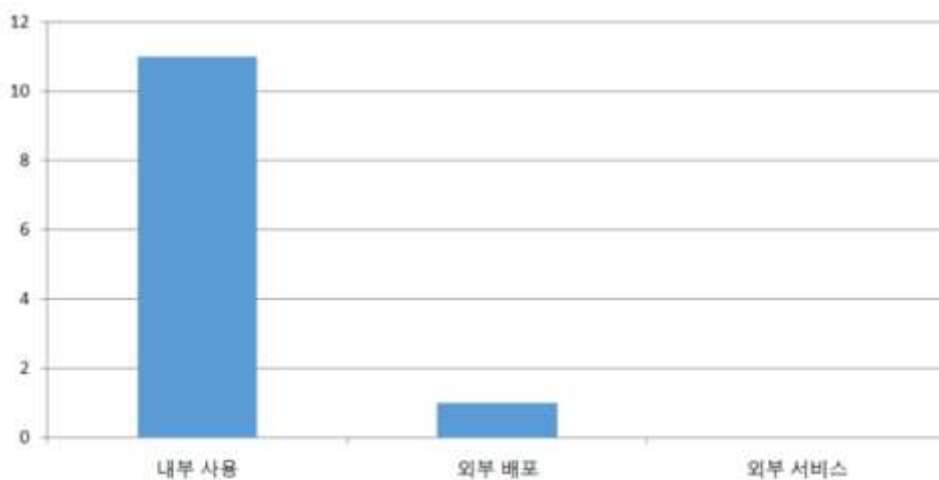
시스템에 관련된 언급을 기피하는 것이 원인이라고 분석된다. 앞으로 제4차 산업혁명이 가속화 될수록 개발 시스템에서 공개SW를 활용하는 비율이 더욱 증가할 것으로 예상된다.

그리고 SW 개발을 위한 공개SW 활용 용도에 대한 질문에 [그림 4-28]처럼 설계 단계 활용 응답이 1개, 개발(구현) 단계 활용 응답이 5개, 프로세스 지원 응답이 8개, 개발 대상 시스템 응답이 1개로 조사되었다.

[그림 4-28] 공개SW 활용 용도 (복수 응답)



[그림 4-29] 공개SW 활용 목적 (복수 응답)



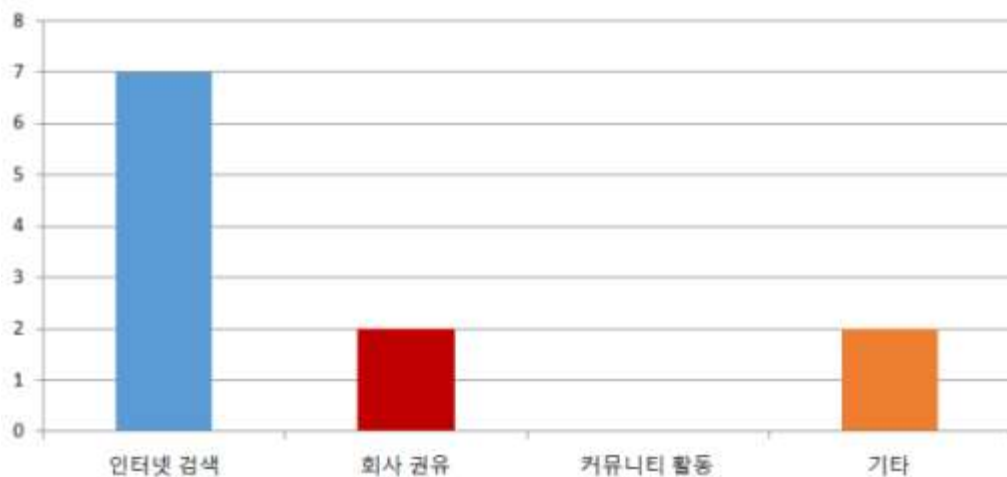
[그림 4-29]는 공개SW 활용 목적에 대한 결과인데, 공개SW를 활용하는 11개 기업 모두가

기업 내부에서 사용하고 있다고 답변하였으며, 이 중 1개 기업이 외부 배포를 목적으로 공개SW를 사용하고 있다고 답변하였다. 이 결과를 토대로 보면 다수의 기업들에서 공개SW의 소스코드 공개 조항에 대한 우려가 있기 때문에 외부 배포에 대해서는 조심하고 있는 것으로 판단된다.

이 결과들을 보면 안전 분야 기업의 공개SW 활용 수준은 낮은 것으로 분석되며, 공개SW에 대한 사용 방법 또한 주로 내부적인 사용 목적으로 사용하는 것으로 조사되었다. 안전 분야의 대부분의 기업들은 공개SW의 소스 코드를 수정하여 새로운 SW를 개발하는 형태가 아닌 개발 프로세스에서 완성된 도구 형태의 SW를 사용하는 것으로 조사되었으며, 특히 프로세스 지원도구의 형상관리 측면으로 가장 많이 사용하는 것으로 확인되었다.

공개SW를 활용하고 있는 11개 기업을 대상으로 공개SW를 알게 된 경로에 대해 조사하였으며, [그림 4-30]처럼 필요에 의한 인터넷 검색에 대한 응답이 7개로 가장 높은 응답률을 보이고 있으며, 그 외 회사 권유를 통한 공개SW 사용에 대한 응답이 2개, 기타 2개로 조사되었는데 기타는 회사 동료의 개인적 권유로 사용하였음을 밝혔다. 안전 분야에서는 조직적인 차원에서 공개SW를 활용하기보다 개인적 필요에 의해 공개SW를 사용하는 것으로 조사 결과에 나타났다.

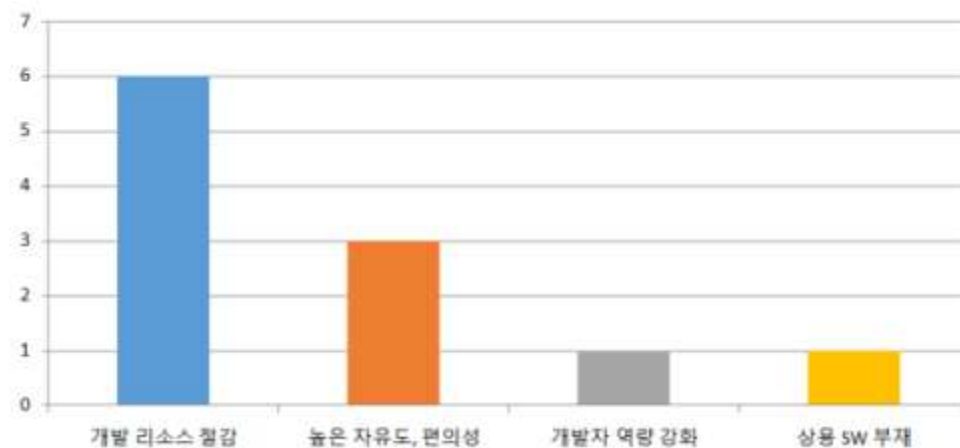
[그림 4-30] 공개SW 인지 경로



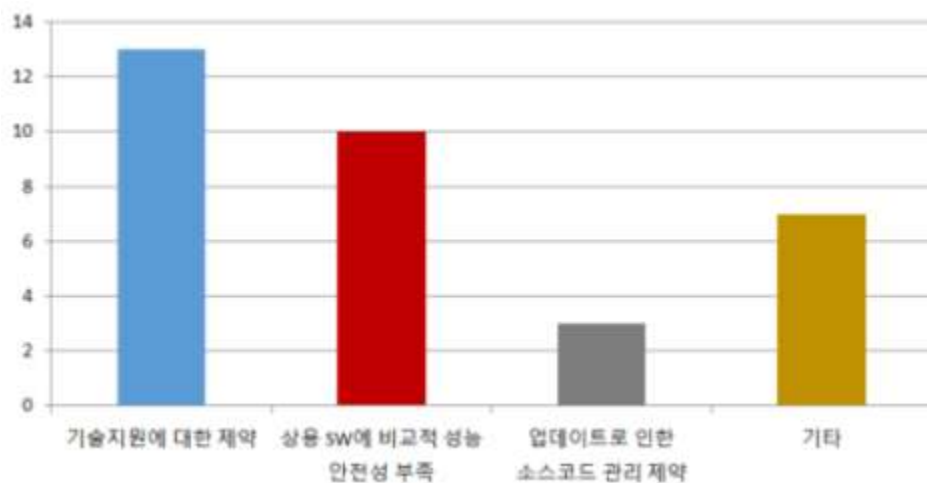
공개SW를 사용 기업 11개를 대상으로 공개SW를 사용하는 이유에 대해 물었을 때 안전 분야 기업들의 종사자들은 [그림 4-31]처럼 개발 리소스 절감이 6개로 가장 많은 응답을 보

였으며 높은 자유도 및 편의성이 3개, 개발자 역량 강화와 상용SW 부재라는 이유가 각각 1개씩 응답하였다. 전체적으로도 절반 이상의 응답자들이 개발 리소스 절감을 목적으로 공개 SW를 사용하는 것으로 나타났다.

[그림 4-31] 공개SW 사용 이유 (복수 응답)



[그림 4-32] 공개SW 사용하지 않는 이유 (복수 응답)



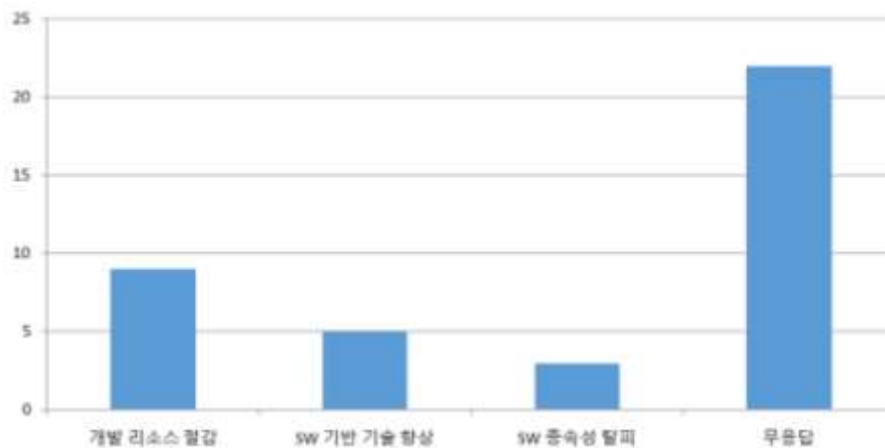
[그림 4-32]은 공개SW를 사용하지 않는 이유에 대한 설문 결과로 기술지원에 대한 제약에 대한 응답이 13개, 상용SW에 비교적 성능 안전성 부족에 대한 응답이 10개, 버전 업데이트로 인한 소스코드 관리 제약에 대한 응답 3개, 기타 응답이 7개로 조사되었다. 기타 응답에는 코드 공개에 대한 리스크, 라이선스 정보 부족, 요구사항에 맞는 SW가 없다는 답변들

이 있었다. 본 설문 결과에 의하면 아직까지는 국내의 안전 분야 기업들은 공개SW를 주도로 활용할 수 있는 상황이 아님을 알 수 있다.

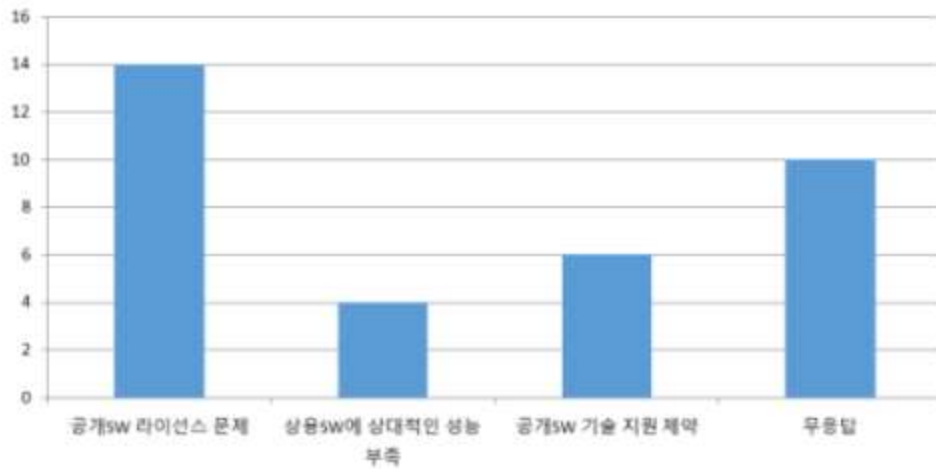
안전 분야 기업들이 공개SW에 대해서 어떻게 생각하고 있는지에 대하여 조사를 수행하였다. 현재 응답자들이 생각하는 공개SW의 장점에 대해서는 [그림 4-33]처럼 30개 설문지 중에서 22개 설문지에서 무응답으로 답변하지 않았으며, 이는 곧 국내 안전 분야 기업들은 공개SW에 대하여 큰 의미를 부여하고 있지 않음을 알 수 있다. 그리고 복수 응답한 결과 개발 리소스 절감이 9개로 가장 많았고, SW기반 기술력 향상이 5개로 2번째로 응답이 많았고 마지막으로 SW 종속성 탈피가 3개로 조사되었다.

[그림 4-34]는 응답자들이 생각하는 공개SW의 단점에 대한 조사 결과로, 30명의 응답자 중에서 10명의 응답자들이 응답하지 않았고 나머지 응답 결과 중 중복 응답에 의한 결과 공개SW 라이선스 항목을 14개로 가장 많았으며 다음으로는 공개SW의 기술 지원 제약이 6개였으며 상용SW에 상대적 성능 부족 이슈가 4개로 나타났다.

[그림 4-33] 응답자들이 생각하는 공개SW 장점 (복수 응답)



[그림 4-34] 응답자들이 생각하는 공개SW 단점 (복수 응답)

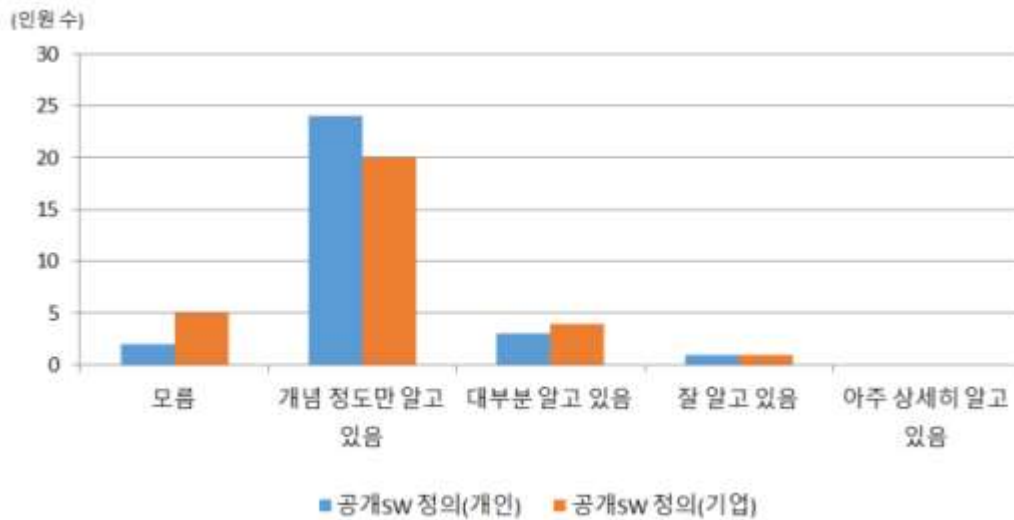


[그림 4-33]와 [그림4-34]의 결과를 종합해보면 국내의 공개SW 사용 목적은 SW 개발 기술력 향상 보다는 SW를 개발하는 리소스를 최소화 하는 것에 중점을 두고 있으며, 아직 국내에서는 공개SW에 대한 정보 및 이해도가 높지 않기 때문에 라이선스 문제를 공개SW의 가장 큰 위협으로 판단하고 있는 것으로 해석된다. 이는 공개SW 활용 관점에서 초보적인 수준에 머물러 있음을 알 수 있다.

4. 공개SW에 대한 이해도

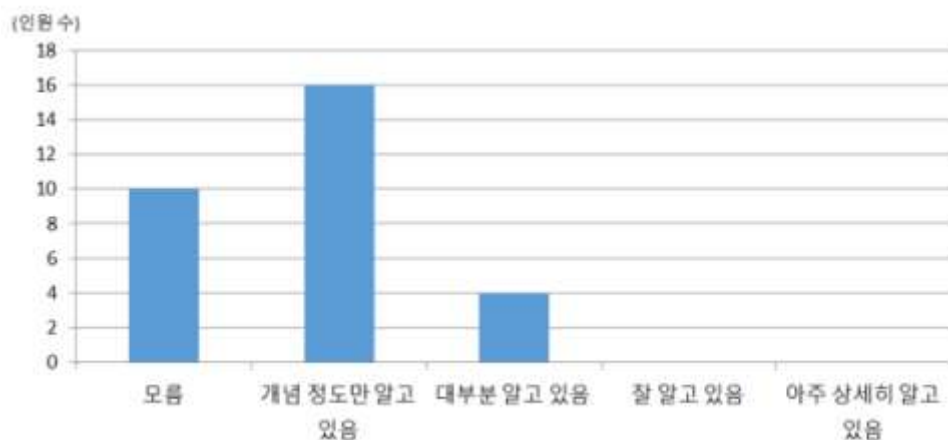
[그림 4-35]은 공개SW의 정의 및 개념에 대한 응답자 개인 및 소속 기업의 이해도를 조사한 결과이며, 설문 응답자 중 24명(약 80%)의 응답자가 공개SW에 대해 개념 정도만 알고 있다고 답변하였고, 기업 차원에서는 20명(약 66%)의 응답자가 개념 정도만 안다고 응답하였다. 그리고 기업 차원에서 공개SW에 모른다고 응답한 응답자가 5명(17%)이었다. 이를 토대로 보면 국내 안전 분야 기업들은 공개SW에 대한 인식이 매우 부족하다는 것을 알 수 있다.

[그림 4-35] 기업 및 설문 응답자의 공개SW 정의 및 개념 이해도

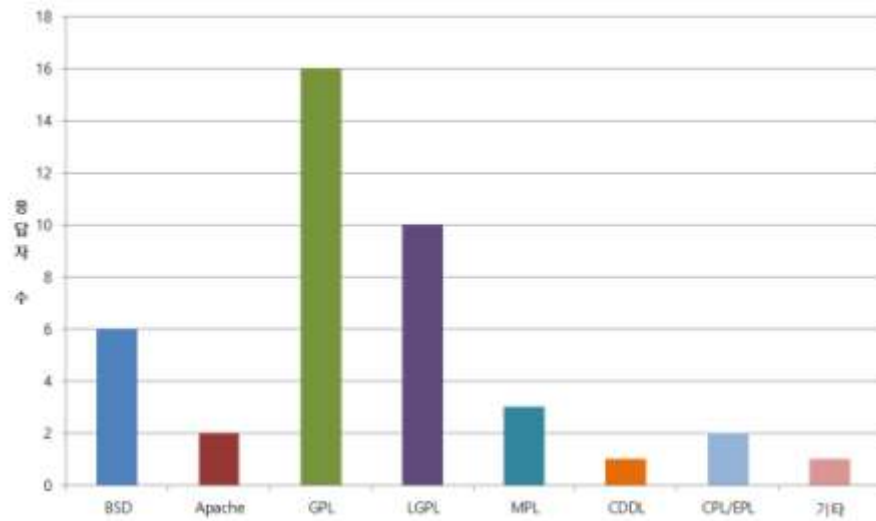


공개SW의 자세한 이해도를 측정하기 위해 공개SW 중요 이슈인 공개SW 라이선스 이해도에 대한 설문 결과는 [그림 4-36]처럼 개념 정도만 알고 있다고 답변한 응답자가 16명(54%)으로 가장 많았다. 모른다고 답변한 응답자가 10명(33%)으로 공개SW의 개념에 대하여 알고 있다고 한 응답자 중에서 일부는 공개SW 라이선스에 대해 모른다고 응답했다는 것을 알 수 있다. 공개SW 라이선스에 대해 대부분 알고 있다고 답변한 응답자가 4(13%)명으로 매우 소수임을 알 수 있다. 앞선 공개SW 이해도에 비해 라이선스 이해도가 낮은 것은 실질적으로 응답자들의 공개SW 이해도가 낮다는 것을 증명한 결과이다.

[그림 4-36] 공개SW 라이선스 이해도



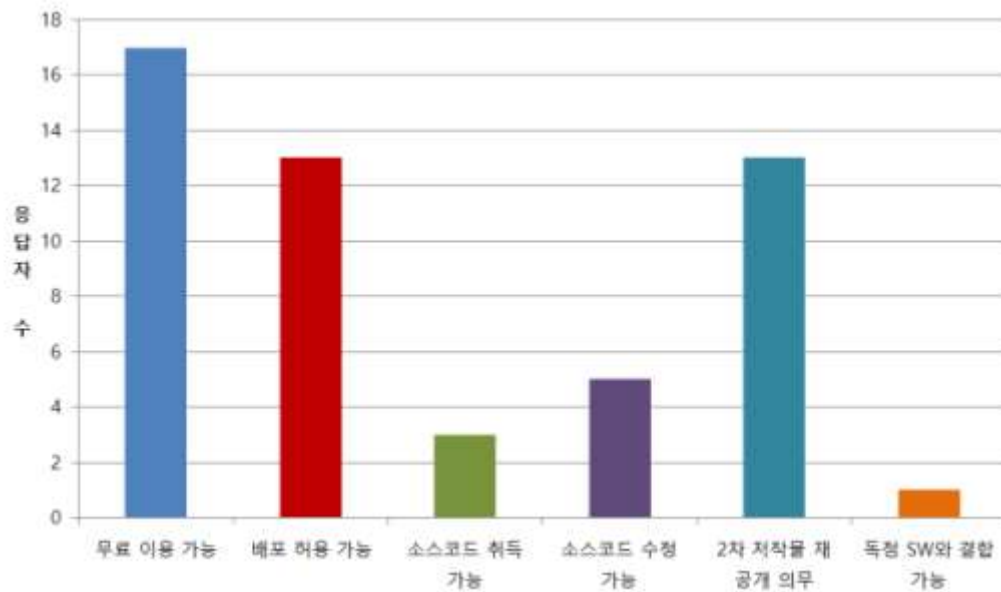
[그림 4-37] 알고 있는 공개SW 라이선스 (복수 응답)



공개SW 라이선스 이해도에 대한 정확한 파악을 위해서 알고 있는 공개SW 라이선스종류에 대한 설문에 대한 결과가 [그림 4-37]이다. 응답자들이 가장 많이 알고 있는 라이선스는 GPL 라이선스가 16명이며 해당 숫자는 공개SW 라이선스 이해도에서 개념 정도만 알고 있다고 답한 16명과 동일하다. 그리고 다른 공개SW 라이선스에 대한 인지도를 보면 LGPL 10명, BSD 10명, MPL 3명, Apache 및 CPL/EPL 각각 2명, CDDL 및 기타가 각각 1명으로 조사되었다. 해당 결과를 보면 대체적으로 공개SW 라이선스에 대한 인지도가 낮음을 알 수 있고 최근 부각되고 있는 허가적 라이선스(BSD, 아파치, MIT)에 대한 인지도도 낮음을 알 수 있다.

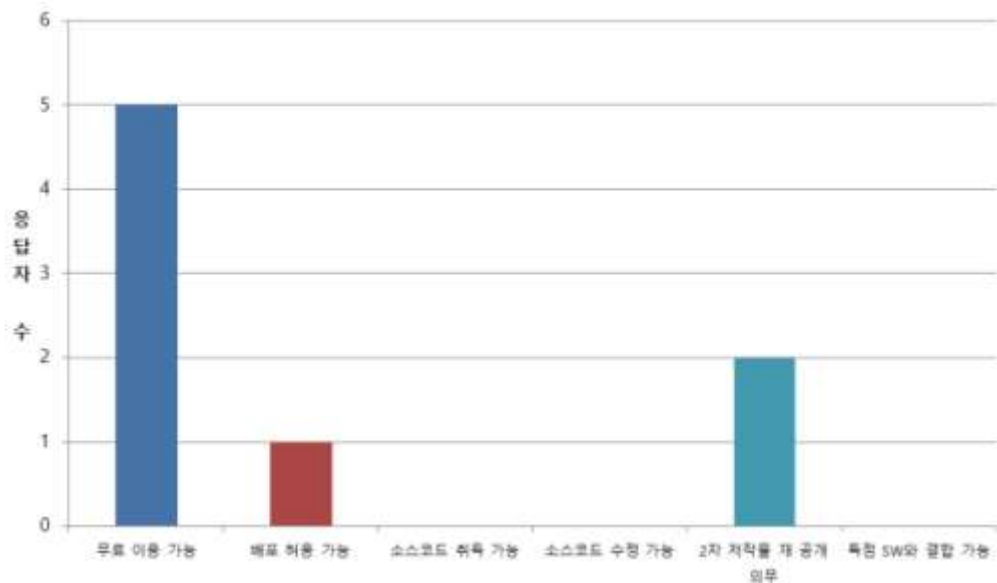
공개SW의 라이선스 주요 정책에 대해 알고 있는 내용에 대한 설문에는 [그림 4-38]처럼 무료 이용 가능, 배포 허용 가능, 2차 저작물 재공개 의무 등의 공개SW 관련 일반적인 사항에 대해서는 알고 있다고 답변한 응답자들이 10명이 넘었다. 하지만, 소스코드 취득 가능, 수정 가능, 독점 SW와의 결합 등의 상세한 라이선스 정책들에 대해서는 인지도는 낮게 나왔다.

[그림 4-38] 라이선스 주요 정책 이해도 (복수 응답)



기업에서 사용하고 있는 공개SW의 라이선스 주요 정책에 대해 알고 있는지 조사해본 결과는 [그림 4-39]과 같다. 무료 이용 가능, 배포 허용 가능, 2차 저작물 재공개 의무의 항목에 대하여 각각 5명, 1명, 2명 만이 알고 있다고 일부 응답하였다. 이를 보면 실질적으로 공개SW를 활용하고 있음에도 공개SW 라이선스에 대해서 잘 모르는 경우가 많음을 알 수 있다.

[그림 4-39] 사용 중인 공개SW의 라이선스 정책 이해도

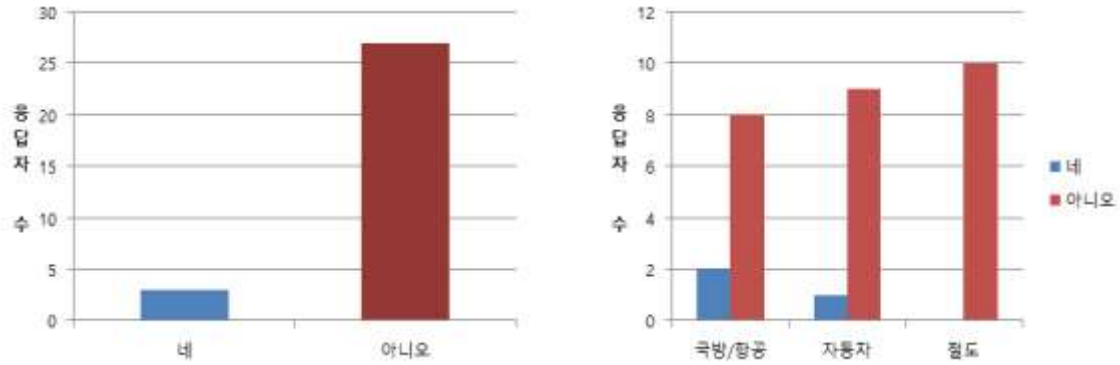


결과적으로 안전 분야의 현업 종사자들은 공개SW의 개념 및 라이선스의 주요 정책에 대해 이해도가 매우 낮은 것으로 조사되었다. 공개SW 사용 현황에서 나타난 것처럼 50%가 넘는 19개 기업이 공개SW를 사용하지 않고 있다. 또한 공개SW 라이선스 등의 현안에 대한 이해도 낮음을 알 수 있었다. 이는 업무 수행에 있어 공개SW와 관련성이 낮기 때문에 공개SW에 대하여 자연스럽게 관심도가 낮아 나타난 결과로 보인다. 그리고 일부 업체에서 공개SW를 활용하고 있음에도 공개SW 라이선스 정책에 대한 낮은 인지도가 보이고 있다는 사실은 안전 분야 기업들의 공개SW 활용에 따른 라이선스 위반으로 인한 경제적 손실의 가능성이 있으며, 공개SW를 활용함으로써 얻을 수 있는 효과에 대해서도 낮은 이해를 가지고 있다는 것을 알 수 있었다.

5. 공개SW 활성화에 필요 사항

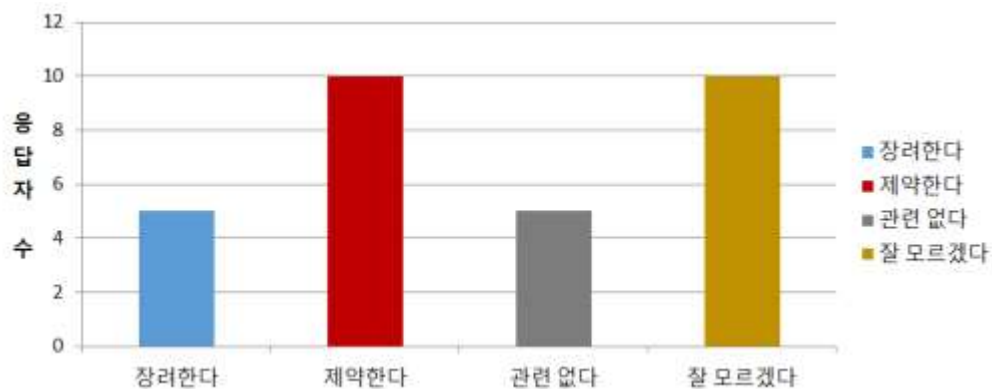
설문 조사에서 공개SW 관련된 교육, 세미나, 발표 등의 경험 여부에 대해 조사한 결과 27명(90%)의 응답자들이 경험이 없다고 답변하였고, 3명(10%)이 경험이 있다고 답변할 정도로 안전 분야에서는 아직까지 공개SW에 대한 관심도가 매우 낮은 것으로 조사되었다. 특히 [그림 4-40] 결과처럼 철도 분야에서 공개SW 관련한 교육(세미나) 경험이 있는 응답자가 한 명도 없었다.

[그림 4-40] 공개SW 교육(세미나) 경험



[그림 4-41] 결과처럼, 안전 분야 기업 중 공개SW 활용을 제약한다는 기업은 전체의 1/3에 해당하는 10개 기업이었으며 단지 5명의 응답자의 기업에서만 공개SW 활용을 장려한다고 답변하였고 약 15명의 응답자들은 관련 없거나 잘 모르겠다고 답변하였다.

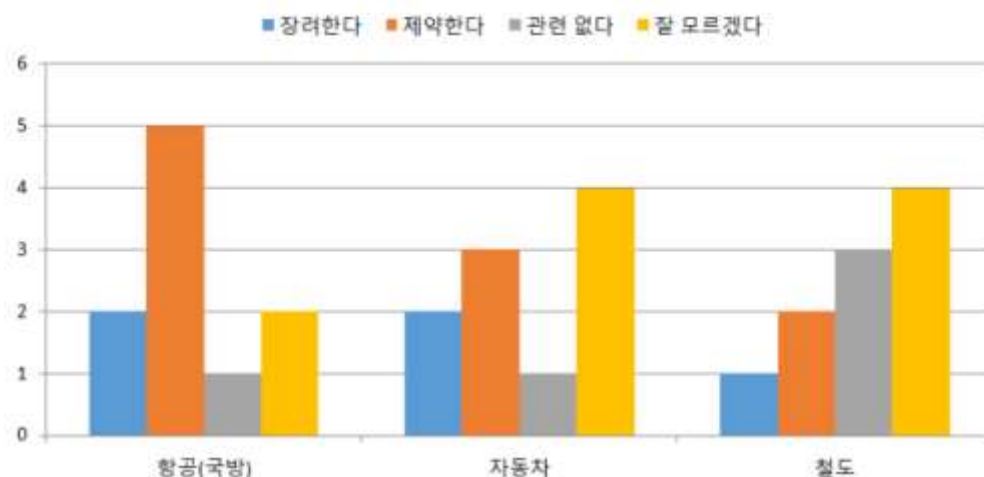
[그림 4-41] 공개SW에 대한 기업 인식



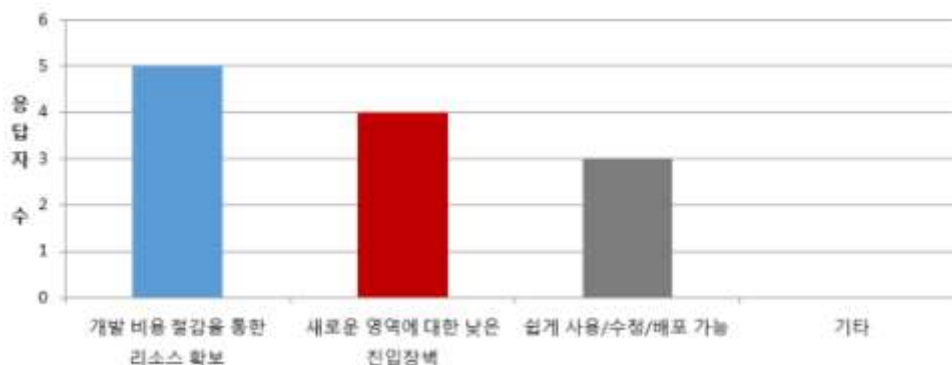
[그림 4-42]는 분야별 기업들의 공개SW 인식 결과를 나타낸 것이다. 항공(국방) 분야는 5개(50%)의 기업에서 제약한다고 가장 많이 응답하였는데, 이는 무기체계 SW 개발 및 관리 매뉴얼의 공개SW 관련 지침에서 공개SW 활용에 대한 부분적인 제약이 있기 때문인 것으로 분석된다. 자동차 분야는 공개SW 활용을 장려하는 기업이 2개(20%), 제약하는 기업이 3개(30%)이며, 관련없다는 응답이 1개, 모르겠다는 응답이 4개(40%)였다. 자동차 분야에서는 아직 공개SW에 대한 기업인식이 뚜렷하지 않는 것으로 분석된다. 철도 분야에서는 공개SW

활용에 대한 기업의 인식에 대해 잘 모르거나, 관련 없다는 응답이 7개 기업에서 조사되었으며, 이는 철도 분야에서 공개SW에 대한 관심도가 매우 낮은 것으로 분석된다.

[그림 4-42] 안전 분야별 공개SW 기업 인식



[그림 4-43] 기업들의 공개SW 활용 장려 이유

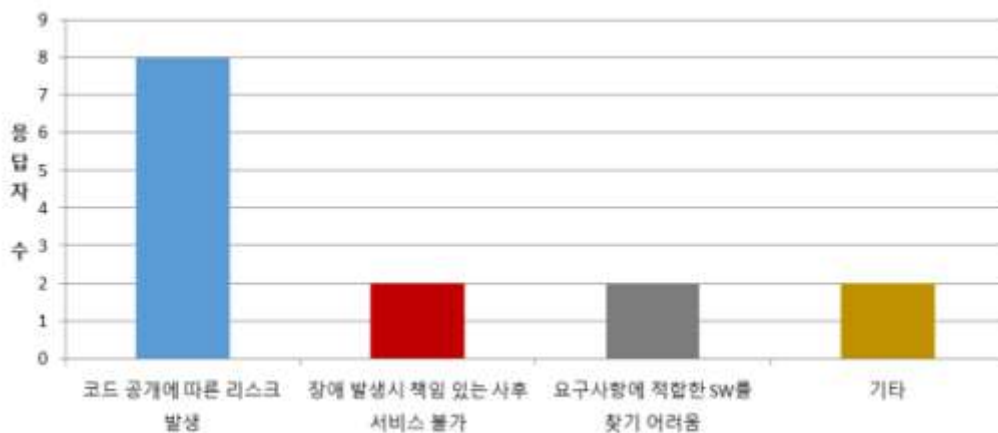


공개SW 활용을 장려하는 기업들을 대상으로 한 장려 이유에 대해서는 [그림 4-43]처럼, 5개 기업 모두에서 개발 비용 절감을 통한 리소스 확보라고 답하였으며, 새로운 영역에 대한 낮은 진입장벽이 4개 기업, 쉽게 사용/수정/배포 가능하다는 이유를 3개 기업에서 답하였다. 이를 통해 공개SW를 장려하고자 하는 기업들은 공개SW의 장점을 이해하고 있으며, 공개SW 사용이 기업의 성장에 도움이 된다고 인식하는 것으로 확인되었다.

반대로 공개SW 활용을 제약한다고 답변한 기업에서는 그에 대한 이유로 코드 공개에 따

른 리스크 발생이 8개로 가장 많았으며 장애 발생시 책임있는 사후 서비스 불가 및 요구사항에 적합한 SW를 찾기 어렵다는 응답이 각각 2개의 기업들로 조사되었다. 기타 응답으로 공개SW 활용에 대한 부분 제약으로 인해 공개SW 라이선스 문제 발생에 따른 리스크라고 답변하여서 실질적으로 공개SW 활용에 따른 위험 회피를 위한 요소가 가장 많이 응답되었다. [그림 4-44]는 해당 결과를 보여준다.

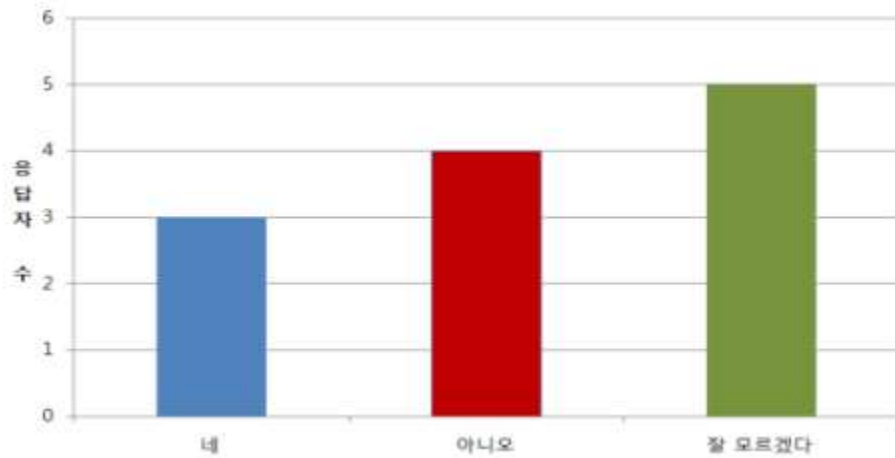
[그림 4-44]기업들의 공개SW 활용 제약 이유



종합적으로 보면 국내에서는 아직까지 코드 공개를 통해 기업의 SW기술을 향상시킬 수 있다는 미래지향적인 인식보다는 핵심 기술 유출로 인한 기업적 손실과 공개SW 라이선스 위반으로 인해 법적 문제 발생 및 비용 손실에 대한 우려가 큰 것으로 분석된다.

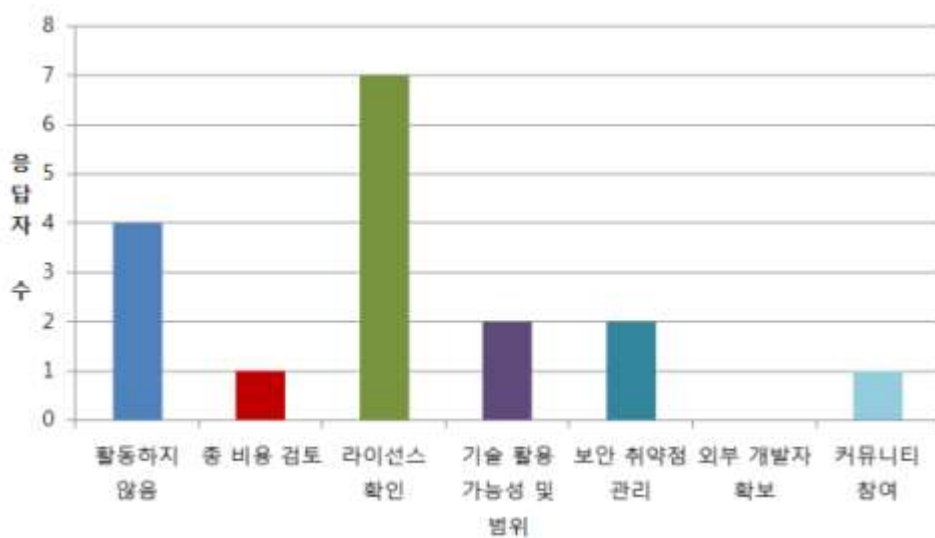
[그림 4-45]는 공개SW 관리 활동을 하고 있는 12개 기업을 대상으로 관리 활동의 적절성 여부를 조사한 결과이다. 결과를 보면, 모른다고 답변한 응답자가 5명(42%)으로 가장 많았으며, 그 외 적절하지 않다고 답변한 응답자가 4명(33%), 적절하다고 답변한 응답자가 3명(25%)으로 조사되었다.

[그림 4-45] 공개SW 관리 활동의 적절성 여부



안전 분야 기업 중 공개SW를 사용하고 있는 11개 기업을 대상으로 공개SW 관리 활동에 대해 복수 답변이 가능하도록 조사를 실시하였으며, 응답으로는 라이선스 확인 활동이 7개 기업으로 가장 많았고, 공개SW 관리 활동을 하지 않는다는 기업도 4개로 나타났다. 그 외에 기술 활용 가능성과 보안 취약점 관리를 한다는 기업이 각 2곳씩 있었으며, 총 비용 검토와 커뮤니티에 참여한다는 기업이 각각 1곳으로 조사되었으며, 결과는 [그림 4-46]과 같다.

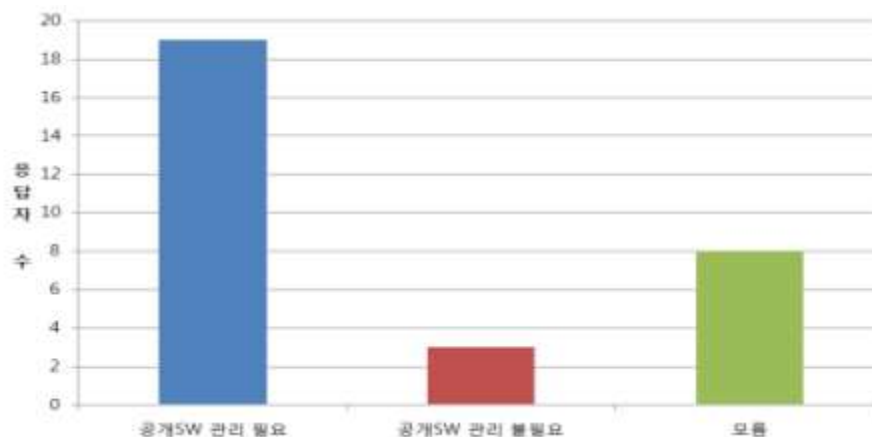
[그림 4-46] 공개SW 관리 활동 (복수 응답)



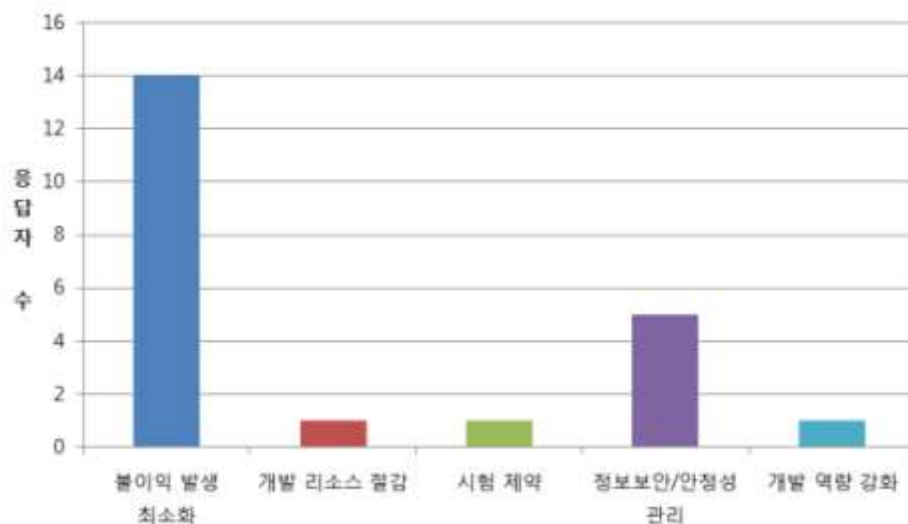
안전 분야 기업들에서 공개SW의 관리가 필요하다고 생각하는지 전체 30개 응답자들에게 물어보았으며, [그림 4-47]처럼 관리가 필요하다는 응답이 19명(63%), 불필요하다는 응답이

3명(10%), 모르겠다는 응답이 8명(27%)으로 조사되어, 현재 공개SW를 활용하고 있는 응답자들이 많지 않았지만 다수의 응답자들은 공개SW 관리 필요성을 인식하고 있는 것으로 조사되었다.

[그림 4-47] 공개SW 관리 필요성



[그림 4-48] 공개SW 관리가 필요한 이유 (복수 응답)

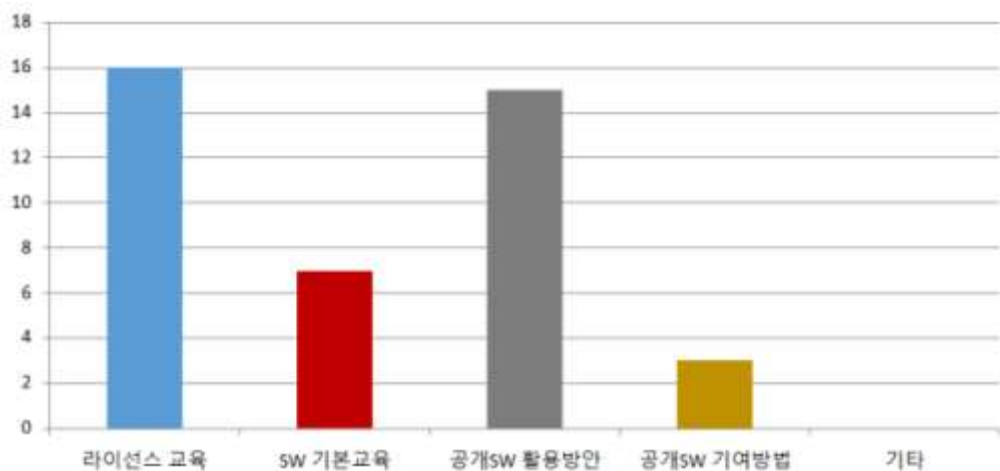


공개SW 관리가 필요하다고 응답한 19명을 대상으로 한 공개SW 관리가 필요한 이유에 대해 복수 응답 결과 [그림 4-48]처럼 14명이 라이선스 문제로 인한 불이익 발생을 방지하기 위해 필요하다고 응답하였다. 따라서 사전에 사용 가능한 공개SW를 식별하고 라이선스 정책을 파악하는 등 공개SW의 관리가 필요하다는 의견을 보이고 있다. 그리고 공개SW의 보

안 취약점에 대한 우려로 인하여 정보보안/안정성 관리가 필요하다는 의견도 5명이었다. 이 설문 결과에 의하면 공개SW를 활용하고 있는 11개 기업들보다 많은 수의 기업들의 응답자들이 공개SW 관리의 필요성을 인식하고 있다는 것을 알 수 있다.

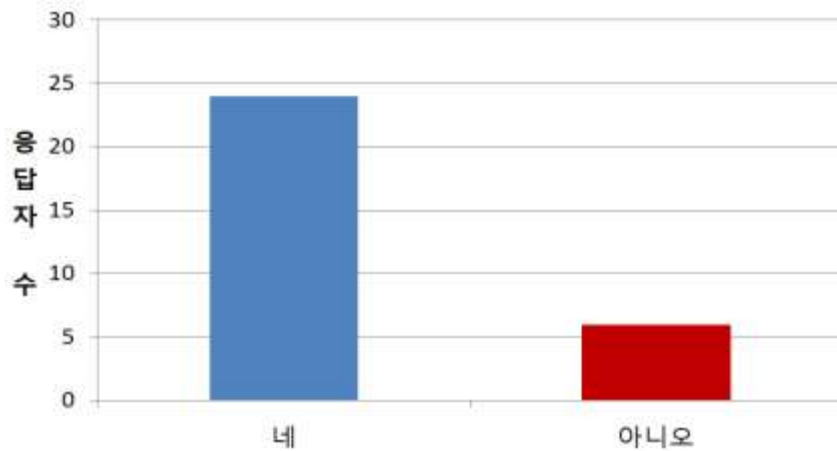
공개SW 활용 활성화를 위한 필요한 지원에 대해 복수 응답 가능한 설문을 하였으며, [그림 4-49]처럼 라이선스 교육이 16명(39%), 공개SW 활용방안이 15명(37%)으로 다수의 응답자들이 답변하였으며, SW 기본교육의 필요하다는 응답이 7명(17%), 공개SW 기여방법에 대한 지원이 필요하다는 응답이 3명(7%)이 나왔다. 이는 안전 분야 공개SW 활용의 확산을 위해서는 공개SW 라이선스 정보 및 공개SW 활용 방안 등의 기본적인 정보의 확산이 우선적으로 필요하다는 것을 보여준다.

[그림 4-49] 공개SW 활성화를 위한 필요 요소 (복수 응답)



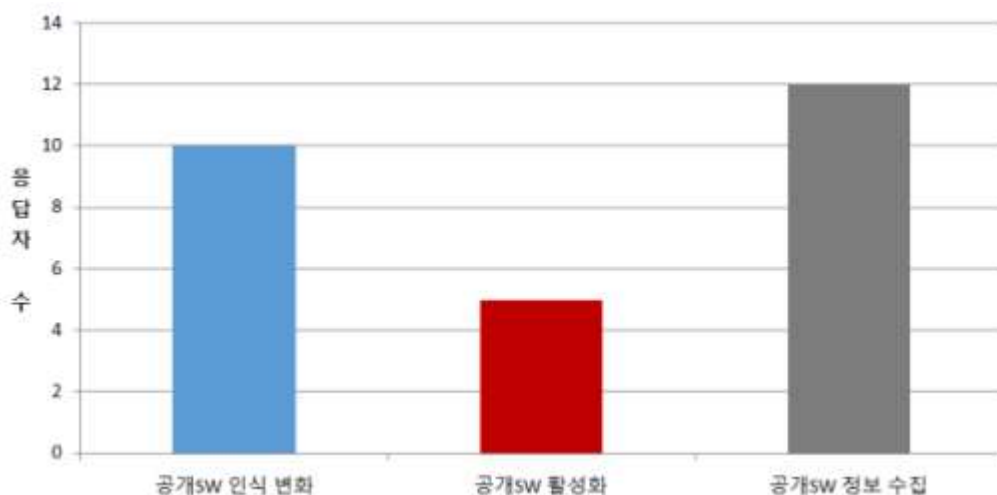
30개 기업들의 종사자들에게 공개SW의 홍보 필요성에 대한 설문에 대해서는 [그림 4-50]처럼, 24명(80%)의 응답자들이 필요하다고 답하였고, 6명(20%)의 응답자들은 불필요하다고 답변하여 전반적으로 공개SW의 홍보 필요성에 대해 매우 긍정적인 결과를 보이고 있다. 이는 설문조사에 응한 응답자들도 공개SW 중요성을 인지하고 있는 것으로 판단된다.

[그림 4-50] 공개SW의 홍보 필요성



24명의 공개SW의 홍보 활동이 필요하다고 답변한 응답자들을 대상으로 홍보가 필요한 이유에 대하여 복수 응답으로 질문한 결과 [그림 4-51]처럼, 공개SW 정보 수집 관점에서 필요하다는 의견이 12명으로 가장 많았으며, 공개SW 인식의 변화가 필요하다는 의견이 10명이었고, 마지막으로 공개SW 활성화를 위해서 홍보가 필요하다는 의견이 5명이었다. 이는 아직까지 국내 안전산업 분야에서는 공개SW의 활동이 활발하게 이루어지지 않고 있기 때문에 이에 대한 정보의 확산이 필요하다는 전반적인 인식이 있기 때문이라고 분석된다.

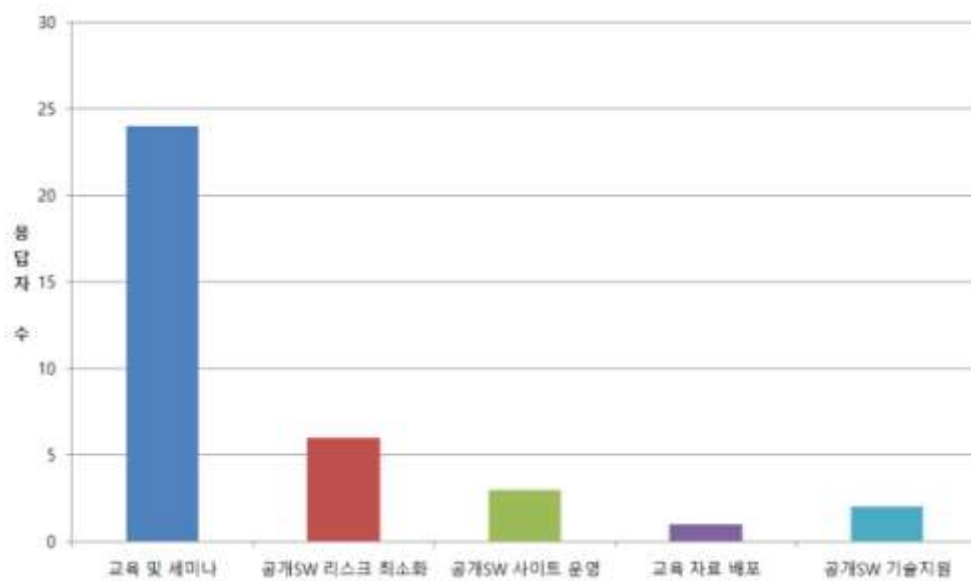
[그림 4-51] 공개SW 홍보의 필요 이유 (복수 응답)



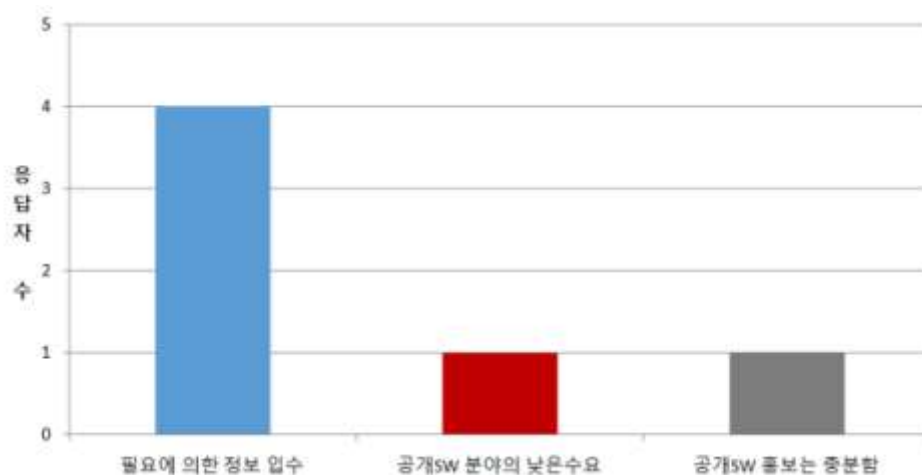
공개SW 홍보의 필요성을 긍정적으로 응답한 24명의 응답자들을 대상으로 한 정부 및 공

공기관의 공개SW 관련 요청 사항에 설문에는 [그림 4-52]처럼 공개SW 관련 교육 및 세미나 요청이 24명으로 가장 많았으며, 그 외 공개SW 리스크 최소화 6명, 공개SW 관련 사이트 운영 3명, 공개SW 기술지원 2명, 교육 자료 배포가 1명으로 조사되었다. 대체적으로 본 설문 대상자들은 공개SW에 대한 정보 제공 및 관련 교육 프로그램 제공에 대한 요청이 있었다.

[그림 4-52] 공개SW 관련 정부/관련기관에 대한 요청사항 (복수 응답)



[그림 4-53] 공개SW 홍보의 불필요 이유 (복수 응답)



공개SW 홍보 활동이 필요없다고 응답한 6명의 응답자들은 그 이유에 대해 [그림 4-53]처럼 필요에 의해 정보를 수집할 수 있다는 의견이 4명으로 가장 많았고, 소수 의견으로 공개SW에 대한 낮은 수요가 1명, 이미 충분한 정보를 획득할 수 있다는 의견이 1명이었다. 공개SW 홍보 활동이 필요없다라는 의견을 가진 6명 중 5명은 이미 공개SW에 대한 충분한 정보를 언제든지 수집할 수 있는 것으로 보아도 될 것 같다. 결국은 공개SW 관련 홍보 활동에 대한 설문 대상자 30명중 오직 1명만이 공개SW에 대한 낮은 수요라고 답변하였다.

제3절 조사 결과 요약 및 시사점

이 보고서의 3장에서는 해외 공개SW 동향에 대한 문헌 조사를 수행하였다. 조사 결과 제4차 산업혁명의 주요 분야라고 할 수 있는 인공지능, 빅데이터, 클라우드 분야의 경우 공개SW가 새로운 기술 진보를 주도하고 있었고 안전 분야에서도 공개SW를 활용하는 사례들이 있음을 알 수 있었다. 특히 자동차 분야의 경우 GENIVI, AGI, 바이두 아폴로 같은 공개SW를 활용하여 제품간 호환성을 향상시키거나 자율주행 같은 신기술 개발이 이루어지고 있었다. 하지만, 철도 및 항공 분야에서는 유의미한 결과를 찾지 못하였다.

이와 같은 해외 동향과 비교하기 위해 국내 안전 분야의 공개SW 현황을 파악하기 위해 3개 분야(자동차, 철도, 항공(국방))을 선정하여 설문 조사를 수행하였다. 설문 조사 내용은 응답자(소속 기업 포함) 정보 및 안전 분야에서의 SW 활용 및 공개SW에 대한 현황을 조사하였다. 하지만, 이번 조사는 전수 조사가 아니었기 때문에 자세한 현황 파악을 위해서는 추가 조사가 필요하다는 한계점이 있다. 그럼에도 불구하고 공개SW 현황에 대한 비SW 산업에서의 동향을 파악한 최초의 조사이었기 때문에 제4차 산업혁명을 대응하기 위한 기초 자료로써 의미있다고 할 수 있다.

이번 조사 결과에 대한 주요 시사점으로 첫째, 국내 안전 분야 기업들의 공개SW에 대한 저변이 넓지 않다는 것을 알 수 있다. 둘째, 국내 안전 분야 종사자들은 공개SW의 필요성을 인지하고 있다는 것을 알 수 있다. 셋째, 국내 안전 분야 기업의 종사자들은 공개SW에 대한 관심도가 있으며 이에 대한 교육 등의 지원 프로그램을 원하고 있다. 넷째, 안전 분야에서 공개SW를 활용을 통한 기술 경쟁력 강화를 위해서는 산업적 활용도가 높은 안전성 분석 및 테스트 SW 분야를 선정해서 국가 차원의 R&D가 필요하다.

해외에서 이미 안전 분야 산업에서도 일부 공개SW가 활용되고 있고 특히 자동차 분야에서 최신 기술인 자율주행 및 IVI 기능 개발을 위해 공개SW가 활용되고 있다. 하지만 국내 관련 산업의 SW 개발자들은 SW개발 과정에 공개SW의 활용을 많이 하고 있지 않고 있으며 국내 SW안전 기업들은 공개SW에 대한 필요성을 인식하지 못하다는 것을 이번 설문 조사에서 파악할 수 있다. 실제로 SW개발자들은 공개SW에 대한 개념 정도만 알고 있으며 실질적인 라이선스 조항에 대해서는 거의 모르고 있었으며 공개SW에 대한 교육 경험이 매우 부족하였다. 또한 기업 측면에서 공개SW 활용에 대한 관리 지침이나 전담 조직을 갖추고 있는 경우가 드물었다.

하지만, 안전 분야 산업의 SW개발자들은 공개SW의 필요성을 인식하고 있다고 할 수 있다. 그 이유는 공개SW를 활용하고 있지 않은 개발자들도 관리 필요성에 대해서는 약 2/3 정도가 필요하다고 인지하고 있었으며 공개SW의 홍보 활동이 필요하다는 의견이 약 80%에 이를 정도로 공개SW에 대한 정보를 제공해주길 원하고 있었다. 이러한 것들로 추정하면 안전 분야 SW 개발자들이 공개SW를 적극적으로 활용하고 있지 않음에도 공개SW를 유익하게 활용할 수 있는 가능성이 있다는 것을 추정할 수 있다.

이번 설문 조사에서 정부가 공개SW의 활성화를 위해서 할 수 있는 첫 번째 지원 정책은 공개SW의 저변 확대를 위한 공개SW 관련 교육이나 세미나라고 할 수 있다. 기존 SW 분야의 경우 다양한 공개SW 교육 및 세미나 프로그램을 통해 어느 정도 인지도가 높지만 안전 분야와 밀접한 제조업 분야에서는 아직까지 공개SW에 대한 인지도가 높지 않기 때문에 우선적으로 공개SW에 대한 교육 및 세미나를 통해 공개SW의 가치 및 활용 가능성에 대한 인식 확산이 필요해 보인다. 특히 제4차 산업혁명 같은 SW 융합 중심의 미래 기술 개발을 위해서라도 공개SW 관련 교육 및 세미나가 필요하다.

마지막으로 공개SW 개발 방식의 장점을 국가 R&D 사업과 연계하면 해당 사업의 결과물의 효율성과 활용성을 향상시킬 수 있다. 안전 분야의 R&D에 공개SW 개발 방식을 도입하게 되면 연구 개발 과정에 많은 SW개발자들이 참여함으로써 결과물의 품질을 향상시킬 수 있고, 결과물을 공개함으로써 R&D 사업에 참여하지 않은 기업들도 결과물을 활용할 수 있게 된다. 만약 안전 분야 R&D 사업에 공개SW 개발 방식을 도입하게 되면 국내 안전 분야 기업들의 SW 활용도가 높은 안전성 분석 SW 또는 테스트 SW 개발에 도입을 하게 된다면 다른 분야의 SW들보다 보다 활용도가 높을 것으로 생각된다.

제5장 소프트웨어안전 분야 공개소프트웨어 활성화 방안

이전 장에서 안전 분야에서 해외 공개SW 동향과 국내 공개SW 현황 파악을 위한 설문 조사 결과를 소개했다. 이들 조사 결과를 기반으로 이번 장에서는 공개SW를 활용한 국내 안전 분야 기술 경쟁력 강화를 위한 방안을 제안한다. 우선적으로 국내 안전 분야에서 공개SW 활용 방안을 제안하고 다음으로 국내 안전 분야의 공개SW 활성화 방안을 제안한다.

제1절 공개소프트웨어 활용 방안

국내 안전 분야의 제품 개발에서 SW의 비중은 점차적으로 커지고 있기 때문에 안전 확보를 위한 SW기술력은 점점 중요해지고 있다. 따라서 국내 안전 분야의 산업 경쟁력 강화를 위한 하나의 방안으로써 SW 경쟁력을 강화시켜야 할 필요성도 증가하고 있다. 안전 분야의 SW 경쟁력 강화는 해외의존도가 높은 국내 안전 산업의 기술 자립도와 해외 시장 진출 가능성을 높일 수 있는 기반을 제공해 줄 것이다. 또한 안전 산업의 SW 경쟁력 강화는 인공지능이 널리 활용될 것으로 예상되는 제4차 산업혁명의 대응 차원에서 미래 사회 안전 확보를 위한 필수불가결한 요소라고 할 수 있다.

[그림 5-1] 공개SW 개발 방식의 장점들



출처: 과학기술정보통신부, 공개소프트웨어 연구개발 수행 가이드라인

이들 분야의 연구 개발에 공개SW 개발 방식을 도입한다면 더욱 효율적으로 연구개발물을 활용할 수 있을 것이다. 과기정통부는 2018년 6월 발표한 공개SW 연구개발 수행 가이드라인을 보면 공개SW를 활용한 연구 개발 수행은 5가지의 장점들을 가지게 된다. 첫째, 효율적인 개발이 가능하다. 둘째, 최신 기술 도입에 유리하다. 셋째, 공개SW를 기반으로 기업 주변 확보에 유리하다. 넷째, 벤더 종속성을 탈피할 수 있다. 다섯째, 효율적인 SW 교육을 통해 인재 양성에 유리하다. 따라서 안전 분야도 SW 경쟁력 강화를 위한 R&D 사업에서 공개SW 개발 방식의 적극적인 도입 필요하다.

안전 분야 연구 개발 사업을 위한 분야는 크게 2가지 분야로 정리할 수 있다. 첫째, 기존 안전 분야에서 가장 필요한 활동인안전성 분석 및 테스트 분야의 SW를 국가 R&D 형태로 개발하게 된다면 국내의 기업들에게 많은 도움이 될 것이고 안전 확보에도 기여할 수 있다. 둘째, 제4차 산업혁명의 신규 분야인 자율 주행 분야로 이 분야는 이미 해외에서 공개SW 개발 방식이 적극 활용되고 있는 분야이다.

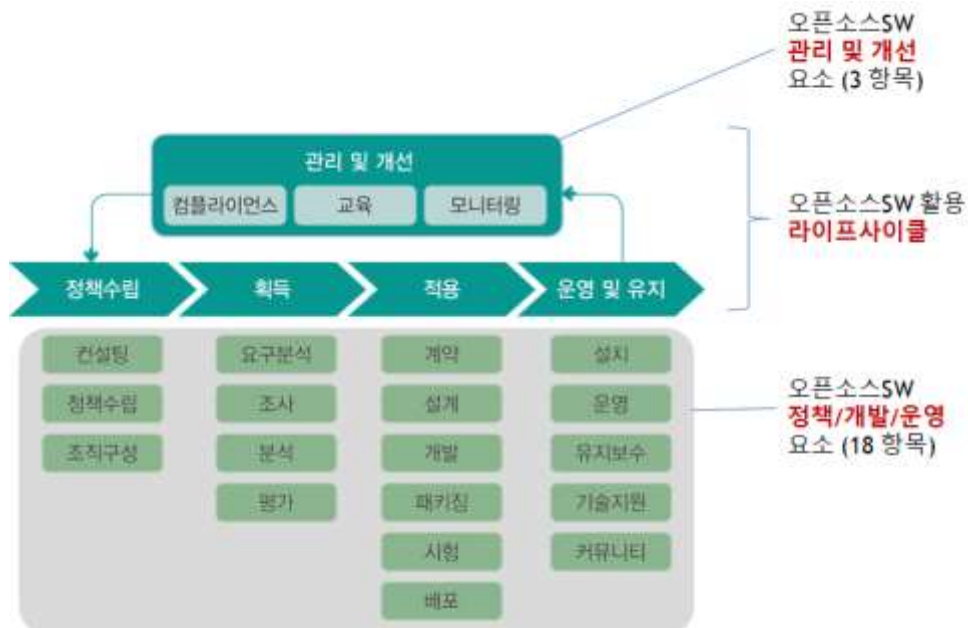
현재 국내 안전 산업의 경우 SW안전에 대한 인식 및 지식 부족으로 인하여 관련 국제 표준에 대한 인지도가 낮고 경험이 부족한 현실이다. 실제로 국가시설 안전 관련 시스템 담당자들 중에서 SW안전에 대한 인지도는 높지 않으며, SW 개발자 및 전문가들도 SW안전에 대한 개념이 명확히 정립되지 않고 있다. 그렇다 보니 추상적으로 표현되어 있는 국제 표준에 대한 이해도 부족으로 SW안전을 위한 위험원 분석 및 위험 평가에 대한 구체적인 방안을 자체적으로 수행하지 못하고 외부 기업들에게 의존하는 경향이 많다. 국내 산업 경쟁력 강화를 위해서 자체적인 R&D 수행을 통한 SW안전 도구를 개발할 필요가 있다. 앞의 4장의 국내 현황 조사에 의하면 SW안전 확보를 위해 가장 많이 필요한 활동으로 선정된 안전성 분석 및 테스트 분야의 SW(총 30개의 기업 중 각각 17개 기업들(약 57%)에서 활용)을 국가 R&D 형태로 개발하게 된다면 국내의 기업들에게 도움이 되고 안전 확보에도 이로울 것이다.

또한, 자율주행, 스마트시티 등의 제4차 산업혁명의 신규 기술 및 산업 분야는 인공지능, 빅데이터, 클라우드 기술의 활용이 필요하고 이들 SW기술들에 의한 제어가 일상화될 가능성이 매우 높다. 하지만 이들 분야에서 SW안전을 어떻게 제공할지에 대해서는 현재로써는 막연한 실정이다. 만약 지금 당장 자율주행 차량이 운행되거나 스마트시티의 자동화 서비스가 구현된다고 하면 이로 인한 인명 사고의 가능성이 매우 클 것으로 예상할 수 있다. 이와 같은 SW로 야기되는 안전사고를 예방하기 위해서는 기존 안전 산업에서 적용되던 SW안전 활동을 통한 위험원 분석, 안전 요구사항 도출, 기능안전 개발, 안전한 SW 검증 및 시험 등 새로운 SW안전 기술 개발이 필요한 시점이다. 실제적으로 3장에서 살펴본 해외 공개SW 동

향을 보면 자동차 분야의 자율주행 기술 개발에서 공개SW 개발 방식이 널리 활용되고 있다. 국내에서도 관련 분야 R&D 사업에서 공개SW 개발 방식을 적극적으로 도입할 필요가 있다.

이러한 SW안전 기술 개발을 진행할 때 공개SW 개발 방식을 적극적으로 도입해야 한다. 그 이유는 해당 연구 개발 결과물의 혜택을 다수의 기업 및 개발자들이 누리게 해야 하기 때문이다. 공개SW 개발 방식은 일반적인 SW개발과 달리 개발 단계에서 다수의 개발자들이 참여를 허용하면서 개발하게 된다. 불특정 다수의 개발자들을 참여시키기 위해서 그림 [5-2]처럼 SW관리 및 개선을 고려해야 함으로써 다수의 개발자들이 참여하기 위한 프로젝트 홍보가 매우 중요하다고 할 수 있다. 또한 다수의 개발자들이 프로젝트에 참여했을 경우 이들이 제기하는 이슈에 대하여 친절하게 대응할 수 있는 체계를 갖추어야 한다.

[그림 5-2] 공개SW 개발 방법



출처: 오픈소스 & 거버넌스 - 인베숨 김형재

안전 분야에서 공개SW를 활용한 연구 개발 수행시에는 과학기술정보통신부의 공개SW 연구개발 수행 가이드라인을 참고하여 공개SW 라이선스 선정, 소스코드의 공개 범위/시점/방법, 커뮤니티를 활용한 개발 방법, 개발자들의 커뮤니티 참여를 위한 홍보 방법, 공개SW 결과물의 사업화 방안 등에 대한 철저한 고민이 필요하다.

이와 같은 안전 분야의 SW경쟁력 강화를 위해 연구개발 과정에 공개SW 개발 방식을 도

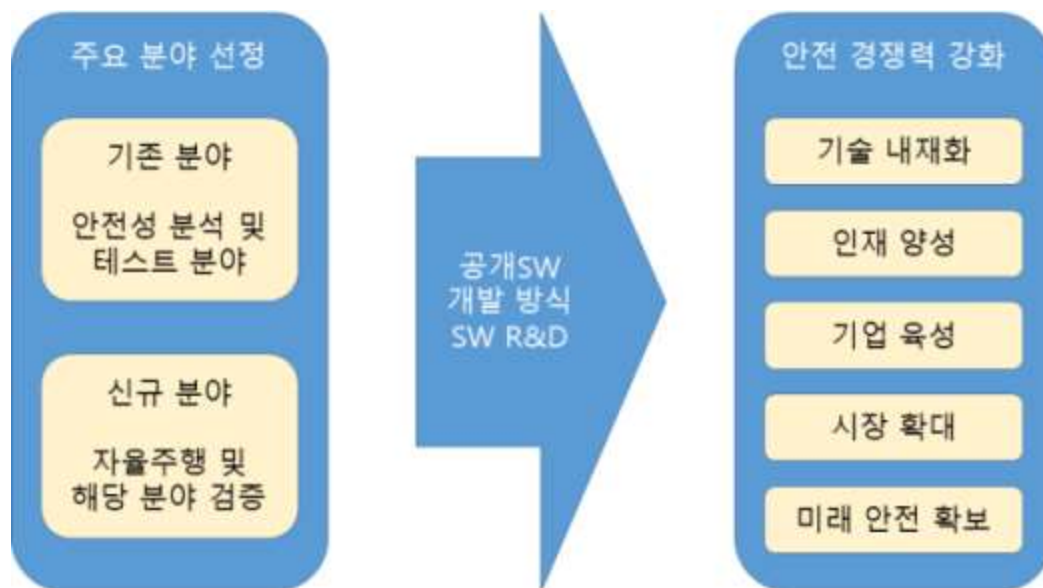
입할 경우 다음의 기대효과들이 발생할 것으로 예상된다.

첫째, 안전 기술 내재화이다. 공개SW 개발 방식을 활용함으로써 연구 개발에 참여한 연구 기관과 기업만 아니라 개발된 공개SW를 이용할 수 있는 타 기업들도 기술 확보가 용이해진다.

둘째, 안전 관련 인력 양성이다. 공개SW 개발 방식을 적용한 연구 개발에 참여한 인력 양성과 더불어 관련 분야의 인력 양성을 위한 교육 기관에서 공개SW를 활용함으로써 현장에서 바로 활용가능한 우수 인력 양성이 가능하다.

셋째, 안전 분야 전문 기업 육성이 가능하다. 공개SW 개발 방식을 활용함으로써 다수의 개발자들이 참여하여 산업 현장에 바로 적용할 수 있는 좋은 품질의 SW를 개발할 수 있고 해당 노하우를 기반으로 경쟁력 있는 기술 기업을 육성할 수 있다.

[그림 5-3] 안전 분야 공개SW 활용 방안 요약



넷째, 안전 분야 시장 확대가 가능하다. 신규 분야의 경우 기술 부족으로 인하여 시장 창출이 어렵고 기존 분야의 경우에도 저변 확대가 어려웠지만 개발된 공개SW는 다수의 기업 및 개발자들이 참여함으로써 기술의 보편성을 확장할 수 있기 때문에 이를 통해 시장 확대가 가능하다.

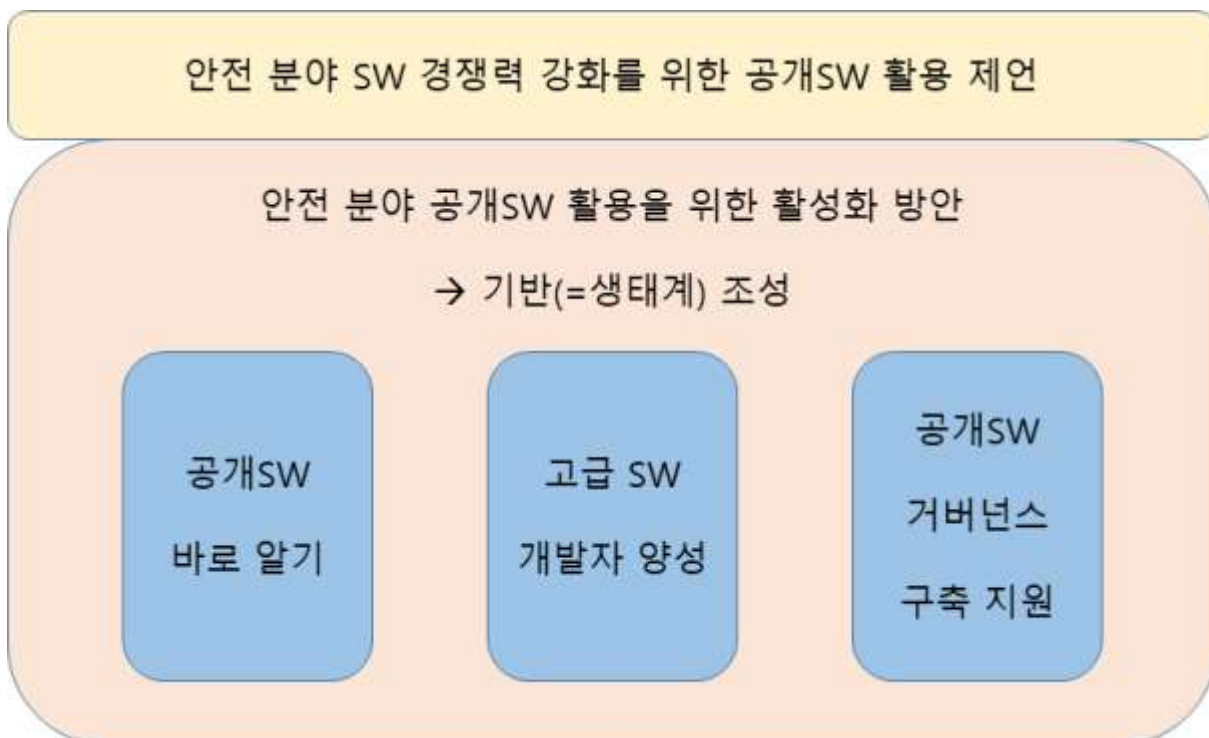
마지막으로 공개SW 개발 방식을 활용한 연구 개발의 결과로 기술 내재화, 인재 양성, 시

장 확대를 통해 미래 사회의 안전을 더욱 강하게 확보할 수 있을 것으로 예상된다.

제2절 공개소프트웨어 활성화 방안

현재 공개SW 개발 방식은 SW산업의 주도권 확보를 위해 필수불가결한 요소가 되고 있다. 공개SW 개발 방식은 다수의 기업들과 다수의 개발자들이 참여함으로써 좋은 품질의 SW를 개발하고 해당 결과물을 다수가 이용할 수 있도록 하여 SW 생태계 규모를 키우는 비즈니스적 관점에서 매우 효율적인 SW개발 방식이다. 이 방식은 구글 안드로이드를 통해 증명이 되었고, 이후 아마존 AWS, 페이스북, 우버, 에어비엔비 등 많은 IT 혁신 기업들이 활용하고 있다. 따라서 공개SW 개발 방식을 안전 분야 경쟁력 강화를 위해 활용해야 함을 앞에서 제안하였다.

[그림 5-4] 안전 분야 공개SW 활용을 위한 활성화 방안



하지만, 공개SW 개발 방식을 도입하기 위해서는 관련 산업 분야에서 이 방식을 잘 활용할 수 있는 기반을 조성해야 한다. 공개SW 기반을 조성한다는 것은 국내 안전 분야에서 공개SW 활용을 위한 생태계를 마련한다는 의미이다. 우선적으로 관련 연구기관 및 기업들이 공개SW 바로 알기가 필요하다. 다음으로 공개SW 개발 방식을 활용할 수 있는 SW개발 인력들이 필요하다. 세 번째로 안전 분야 SW 연구 개발을 위한 공개SW 거버넌스 구축이 필요하다.

안전 분야 공개SW 활성화를 위해 공개SW 바로 알기를 첫 번째로 꼽은 이유는 국내 안전 분야 공개SW 현황 조사에 의한 결과이다. 공개SW가 SW산업에서 중요해지고 있고 해외 동향 조사 결과 안전 분야인 자동차 산업에서도 중요해지고 있다. 실제로 GENIVI, AGL, 바이두 아폴로 프로젝트들에 이미 많은 자동차 산업의 주요 회사들이 IVI 기능 개발 및 자율 주행 기능 개발에 참여하고 있다.

하지만, 국내 현황 조사 결과에서 공개SW를 활용하고 있는 기업이 총 30개중에서 11개(약 37%)이며 개발 시스템 보다는 주로 개발 도구로써 1개 기업을 제외한 10개 기업들이 내부적으로만 활용하고 있다. 공개SW 개념에 대한 이해도에서도 30명의 응답자 중에서 24명(약 80%)이 개념 정도만 이해하고 있고 1명은 모른다고 밝혔으며 20개(약 67%)의 기업들은 공개SW 개념 정도를 이해하고 있고 5개(약 17%)의 기업들은 공개SW에 대해 모르고 있다고 답변하였다. 또한 안전 분야 공개SW 인식 설문에서도 30명의 응답자 중에서 기업에서 장려한다는 응답은 5개(약 17%)에 불과하고 제약한다는 응답이 10개, 관련없다는 응답이 5개와 모른다는 응답이 10개로 공개SW 활용의 필요성을 인식하지 못하는 비율이 약 83%에 이를 정도로 매우 높은 현실이다.

따라서 국내 안전 분야의 공개SW 활용을 활성화하기 위해서는 우선적으로 공개SW 바로 알기를 통한 공개SW의 가치에 대한 인식 제고가 필요하다. 실제로 현재 SW산업의 기술 혁신은 공개SW를 기반으로 이루어지고 있으며 인공지능, 빅데이터, 클라우드의 핵심 기술들이 공개SW 기반으로 이루어지고 있다. 그리고 세계적인 IT 기업들은 공개SW를 활용하여 기술 혁신과 비즈니스 혁신을 주도하면서 기업 가치를 끌어올리고 있다. 이러한 공개SW를 활용에 대한 장점을 홍보하여 국내 안전 분야 기업들의 공개SW에 대한 인식을 제고할 필요가 있다. 이 보고서에 포함된 국내 현황 조사 결과에서도 30명의 응답자 중에서 24명(80%)의 응답자들이 공개SW의 홍보 필요성을 인정하였으며 공개SW 홍보의 방안으로써 공개SW 정보 수집을 12명(40%)이 선택하였고, 공개SW 인식 변화를 10명(약 33%)이 선택할 정도로 설문 응답자들은 공개SW 인식 제고 및 정보 수집을 위한 홍보 필요성을 인지하고 있

었다.

안전 분야 기업들의 공개SW 관련 인식 제고 및 정보 수집을 위한 활동으로는 국내 세미나, 컨퍼런스, 관련 포럼 등을 활성화하여 유익한 공개SW의 정보 제공 및 교류를 위해 노력할 필요가 있다. 공개SW 관련 인식 제고에 도움을 줄 수 있는 정보는 다음과 같다.

- 공개SW 상업적 성공 사례
- 공개SW 비즈니스 모델
- 공개SW 사업화 방안
- 공개SW 활용한 해외 진출 방안
- 활용 가능한 유망 공개SW
- 공개SW 라이선스 및 법률 이슈
- 공개SW의 보안 취약점
- 공개SW 거버넌스

이와 같은 정보를 토대로 개별 기업들은 공개SW에 대한 인식을 제고함으로써 안전 분야에서 공개SW를 활용하기 위한 사전 작업으로써 공개SW 활용 방안 도출 및 사업화 전략 수립을 통해 공개SW 활성화에 기여할 수 있다.

안전 분야 공개SW 활용을 위한 활성화 방안으로써 두 번째는 고급 SW 개발자 양성이다. 이는 공개SW를 실질적으로 활용하기 위해서는 사전에 공개된 소스코드를 분석하고 이해하여 해당 코드를 필요한 분야에 수정 적용할 수 있는 고급 수준의 전문가가 필요하기 때문이다.

이번 국내 현황 조사에서도 공개SW를 활용하기 어려운 이유에 대한 복수 응답 설문에서 공개SW를 활용하고 있지 않은 응답자 19명 중 13명(약 68%)이 기술 지원에 대한 제약을 선택했고 10명(약 53%)이 상용 SW에 대한 비교적 성능 안전성 부족을 선택하였다. 이에 반하여 공개SW를 사용하는 이유에 대한 복수 응답 설문에서 공개SW를 활용하고 있다고 응답한 응답자 11명 중 6명(약 55%)은 개발 리소스 절감을 선택했고 3명은 높은 자유도와 편의성을 선택했다. 이 설문 내용에서도 공개SW를 활용하고 있는 응답자들은 공개SW를 활용하여 개발 리소스 절감 및 높은 자유도와 편의성을 누리는데 비하여 활용하고 있지 못하는 개발자들은 공개SW의 높은 난이도 때문에 활용을 못하고 있다고 밝히고 있다.

따라서 국내 안전 분야에서 공개SW를 적극적으로 활용하지 못하고 있는 가장 큰 이유는 실력 있는 SW 개발자의 보유 유무에 따른 결과로 해석할 수 있으며 안전 분야 공개SW 활성화를 위해서 개선해야 할 두 번째 이슈로 선택하였다. 이러한 이슈를 해결하기 위해서는 당연히 고급 SW 개발자 양성이 필요하며 정부에서 이를 해결하기 위한 노력으로 국내 안전 분야의 SW 경쟁력 강화에 이바지해야 할 것이다. 이 이슈는 단순히 안전 분야 산업만이 아니라 제4차 산업혁명 대응을 위한 SW융합 분야 모두에 해당될 수 있다. 특히 자율주행과 연관된 인공지능, 빅데이터, 클라우드 관련 고급 SW 개발자들은 전세계적으로 부족하기 때문에 공개SW 기반 SW 신기술을 활용할 수 있는 고급 SW 개발자들의 양성은 국내 SW 경쟁력 측면에서도 매우 필요하다.

SW안전 기술 및 SW 신기술을 활용할 수 있는 고급 SW 개발자들을 양성하기 위해서는 단기간에는 불가능하며 체계적인 양성 과정이 필요하다. 중장기적으로는 기존 대학 교육 과정에서 공개SW 기반 SW 신기술 교육 및 SW안전 관련 교육을 강화할 필요가 있고 고급 SW 개발자 양성을 위한 대학원 과정을 심화시킬 필요가 있다. 특히 기존 논문 위주의 대학원 과정을 공개SW를 활용한 연구 개발과 연계하여 석박사 과정에서 공개SW 개발 방식으로 SW안전 및 SW신기술 개발 과제를 경험하게 하여 산업 현장에서 바로 투입될 수 있도록 해야 한다. 그리고 단기적으로 기업 현장에서 필요한 인력을 제공하기 위해서는 SW안전 및 SW신기술 관련 기존 종사자들에게 공개SW 개발 방식에 대한 교육을 통해 기업에서 우선적으로 필요한 인력을 재교육함으로써 기존 인력을 활용할 수 있도록 해야 한다.

[그림 5-5] 안전 분야 공개SW 활성화를 위한 인력 양성 방향



안전 분야에서 공개SW 활용을 활성화하기 위한 마지막 제언으로 공개SW 거버넌스 구축 지원이 필요하다. 공개SW 거버넌스는 공개SW를 안전하게 사용/적용/배포하기 위해 필요한 사항을 다양한 관점에서 활용할 수 있도록 소프트웨어 라이프 사이클 단계별로 제시한 틀

로 정의⁹⁹⁾되고 있으며 효율적인 공개SW 활용 및 위험 관리를 위해서는 체계적인 관리를 위한 공개SW 거버넌스 구축이 필요하다. 하지만 안전 분야 국내 기업들은 공개SW 활용에 대한 관리가 부족한 현실이다.

이번 국내 현황 조사에서도 설문 대상이었던 30개 기업 중에서 8개(약 27%) 기업만이 공개SW 관련 사내 지침을 가지고 있으며 5개(약 17%) 기업만이 공개SW 관련 전담 조직을 가지고 있다고 응답하였으며 1개 기업이 사내 지침과 전담 조직을 모두 가지고 있다고 응답하였기 총 12개(40%) 기업만이 공개SW 관리하고 있다고 하였다. 이들 12개의 기업을 대상으로 공개SW 관리가 적절한지에 대한 설문에서 3명(25%)만이 적절하다고 응답하였고 나머지 9명(75%)의 응답자들은 적절하지 않거나(4명, 약 33%) 모르겠다(5명, 약 42%)고 응답하였다. 따라서 기존에 사내 지침을 가지고 있거나 전담 조직이 있는 기업의 응답자들 중 다수는 공개SW 관리가 부족하다고 느끼고 있거나 실질적인 도움을 받고 있지 못하다고 할 수 있다.

하지만 이번 설문 조사에 의하면 30명의 응답자 중에서 안전 분야 기업들의 공개SW에 대한 낮은 이해도(공개SW 개념 수준의 이해 및 모른다는 응답이 25개로 약 83%임)를 고려하면 해당 기업들은 공개SW 관리를 위한 노력이 더욱 필요할 수 있다. 실제로 이번 설문 조사 결과 30명의 응답자들 중 공개SW 관리가 필요하다는 응답자가 19명(약 63%)이었으며 필요없다고 응답한 응답자는 단 3명(10%)에 불과했고 모르겠다는 응답은 8명(27%)이었다. 이 결과는 안전 분야 기업에서 일하는 종사자들 중 일부는 기업 내에서 공개SW 관리가 필요하다고 보고 있다는 사실이다.

이번 설문조사 이외에도 시놉시스¹⁰⁰⁾는 2018년 1100개의 상용 응용프로그램을 분석한 결과로 96%의 프로그램에서 공개SW를 사용하고 있다고 밝혔다¹⁰⁰⁾. 기존 정보통신산업진흥원의 조사에 의하면 공개SW 검증 서비스 결과 2016년 약 95%의 프로젝트들이 공개SW를 활용하고 있고 이중에서 약 39%의 프로젝트들에서 공개SW 라이선스 위반이 발생하였다고 한다¹⁰¹⁾. 이와 같이 공개SW 활용에서의 라이선스 위반은 많이 발생하고 있기 때문에 공개SW 거버넌스 구축 지원은 꼭 필요하다.

공개SW 거버넌스는 라이선스 의무사항 준수, 제 3자의 특허권 및 저작권 침해 같은 법률적 이슈 같은 위험 요인을 회피하여 안전하게 공개SW를 사용하게 하고 공개SW의 기여 활

99) 정보통신산업진흥원, 2017 공개소프트웨어 백서,

100) 공개SW포털, 오픈소스 보안 현황과 시사점, 2018.10.

101) 정보통신산업진흥원, 공개SW 소비국을 넘어 기여국으로 성장을 위한 제언, 2018.01.

동 및 비즈니스 활용 전략 같은 효율적인 공개SW 활용 방안을 도출하기 위해 필요하다. 이를 위해 삼성전자, LG전자 등 일부 대기업들과 ETRI는 내부적인 공개SW 전담 조직을 이미 갖추고 있다. 하지만 이번 설문 조사 대상이었던 안전 분야 기업들은 전담 조직 및 사내 지침이 부족하였는데 그 이유는 비 SW전문 기업 또는 중소기업이라는 한계 때문으로 예측된다.

타 산업에서 SW 활용이 높아지는 제4차 산업혁명을 대응하기 위해서는 기존 SW 기업들만이 아니라 비SW전문 기업 및 중소기업들의 공개SW 활성화가 중요하므로 이들 기업들의 공개SW 거버넌스 구축을 위한 지원이 필요하다. 이를 위해서 우선적으로 기존 공개SW 라이선스 위반 검증 및 보안전문가가 없는 기업들을 위해 공개SW 보안 취약점 점검 프로그램의 저변을 확대하여 SW전문가가 부족하거나 공개SW에 대한 정보가 부족한 기업, 연구기관, 대학들에게 제공해 줄 필요가 있다. 연구기관 및 기업들이 공개SW 거버넌스 구축을 위해 필요한 전담 조직 구성을 위한 컨설팅을 제공하여 공개SW 정책 수립, 자체 라이선스 교육 및 컨설팅이 가능하도록 지원해줘야 한다. 그리고 마지막은 공개SW 라이선스 및 특허 분쟁 같은 법적 이슈가 발생할 경우에도 이에 대한 지원 서비스 제공이 필요하다. 대부분의 중소기업들은 법률 전문가를 자체적으로 고용하기 어렵기 때문에 이런 서비스를 통해 기업들의 공개SW 활용의 문턱을 낮춰 줄 필요가 있다.

[그림 5-6] 안전 분야 공개SW 활성화를 위한 거버넌스 구축 지원



제6장 결론

제1절 연구 요약

제4차 산업혁명과 맞물려 SW 비중의 증대로 인하여 안전 분야에서 SW 중요성이 커져감에 따라 SW안전 개념이 새로이 탄생하고 SW 안전성을 보장하는 이슈는 미래사회 안전 보장에 핵심으로 부각될 것이다. 하지만 국내 안전 산업의 역사가 해외 선진국 대비 짧기 때문에 관련 기술 축적 시간도 부족하기 때문에 아직까지는 기술력이 부족하다.

이러한 현실에서 SW산업 트렌드를 주도하는 공개SW 개발 방식은 선진 기술을 따라잡는데 있어 중요한 역할을 수행할 있으리라 생각된다. 이미 공개SW 개발 방식은 SW기술 혁신과 비즈니스 혁신을 주도하고 있으며 안전 분야에서도 자율주행 관련 프로젝트들이 등장하고 있다. 국내도 SW안전 분야의 경쟁력 강화를 위해 공개SW 활용 방안에 대한 고민이 필요한 시점이 되었다. 하지만 국내 안전 분야에서 공개SW가 어떻게 활용되고 있는지에 대한 국내 연구 자료는 매우 부족하고 특히 국내 안전 분야에서 공개SW 활용 현황 조사 자료는 전무하다.

따라서 본 연구에서는 SW안전 분야에서 공개SW 개발 방식을 활용을 위한 정책 수립을 위한 자료를 제공하기 위해 국내외 안전 분야 관련 공개SW 현황 조사를 수행하고 이를 기반으로 시사점을 도출하였다. 해외 공개SW 현황 조사에서는 제4차 산업혁명의 SW 신기술인 인공지능, 빅데이터, 클라우드 분야의 공개SW 현황과 안전 분야인 자동차, 철도, 항공 분야에서 공개SW 현황을 문헌 위주로 조사하였다.

SW 신기술 분야인 인공지능, 빅데이터, 클라우드 기술들은 이미 글로벌 IT 기업들이 공개SW 개발 방식을 활용하여 기술 혁신과 비즈니스 주도권을 강화하고 있었다. 안전 분야의 경우 자동차 분야에서 IVI 기능 개발 및 자율주행 기능 개발을 위해 공개SW 개발 방식이 GENIVI, AGL, 바이두 아폴로 프로젝트에서 이미 활용되고 있으며 일부 기업들은 공개SW 방식으로 개발된 결과물을 활용하여 상업적으로 활용하고 있었다.

국내 안전 분야에서 공개SW 현황을 파악하기 위해 자동차, 철도, 항공(국방)

분야별 10개의 기업을 선정하여 총 30개 기업들의 SW안전 종사자들을 대상으로 설문 조사를 수행하였다. 설문 결과 국내 안전 분야 기업들은 안전 확보를 위해서 국제 표준을 가장 많이 준수하고 있었으며 안전성 분석 및 테스트가 필요하다는 의견이 가장 많았다.

공개SW에 대한 인식이 활성화되지 않은 상태이어서 약 1/3 정도의 기업만이 공개SW를 이용하여 제품을 개발하고 있었다. 더욱이 공개SW에 대한 이해도가 매우 낮은 상황이었고 기업 내부에서 공개SW 관련 관리가 제대로 이루어지지 않고 있었다. 이런 상황과 달리 설문 응답자들은 기업 내부에서 공개SW 관리 필요성에 대해서는 긍정적으로 인식하고 있었고 라이선스 교육, 공개SW 활용 방안에 대한 정보의 필요성도 인식하고 있었다. 그리고 공개SW 활성화를 위한 정부 및 공공기관에 대한 요청사항으로 교육 및 세미나에 대한 응답이 가장 많았다.

이러한 현황일지라도 SW안전 분야에서 SW안전 분야에서 공개SW 개발 방식을 활용한다면 기술 경쟁력과 산업 경쟁력 강화에 많은 도움이 될 수 있을 것이라 생각된다. 하나의 예로 중국기업인 바이두는 이미 공개SW 개발 방식을 활용하여 자율주행 기능을 개발하기 위한 아폴로 프로젝트는 2017년 공개하였고 이미 100여개의 자동차 기업들이 해당 프로젝트에 참여하고 있다. 그리고 공개SW 개발 방식의 장점들인 개발 효율성, 최신 기술 도입, 기업 저변 확보, 기술 내재화, 인재 양성 등은 이미 SW 산업에서 검증받았다. 구글, 아마존, 페이스북, 마이크로소프트 등 많은 글로벌 선진 기업들이 기술 혁신과 비즈니스 혁신을 위해서 공개SW 개발 방식을 활용하고 있다.

따라서 이번 연구 보고서에서는 안전 분야에서 공개SW 활용 방안과 활성화 방안을 제언함으로써 안전 산업의 경쟁력 강화를 위한 방안을 제안한다. 우선적으로 공개SW 개발 방식을 활용하여 기술 확보의 필요성을 제언한다. 기존 분야에서 안전 확보에 가장 많이 필요하다고 응답된 안전성 분석 및 테스트 분야와 신규 분야에서 제4차 산업혁명의 주요 분야이면서 안전 분야에서 공개SW 개발 방식이 가장 활발한 자율주행 관련 분야를 대상으로 연구 개발 사업을 통한 기술 확보가 필요하다. 해당 연구 개발 사업으로 SW안전 기술 내재화, 인재 양성, 기업 육성, 시장 확대 효과를 얻을 수 있을 것으로 예상된다.

또한, 이러한 연구 개발 사업의 성공과 효과를 극대화하기 위해 국내 안전 분야의 공개SW 기반 조성을 위한 공개SW 활성화 방안을 제언한다. 우선적으로

공개SW 바로 알기를 위해 공개SW 사업화 방안, 해외 진출 방안, 라이선스 관련 정보 같은 정보 제공으로 공개SW에 인식을 제고할 필요가 있다. 다음으로 공개SW를 제대로 활용하고 공개SW 개발 방식으로 SW를 개발할 수 있는 고급 SW 개발자 양성이 필요하다. 마지막으로 공개SW를 안전하고 효율적으로 사용할 수 있는 환경을 제공하기 위한 공개SW 거버넌스 구축 지원이 필요하다.

이러한 제언들은 안전 분야 경쟁력 강화를 위한 SW안전 기술 확보 정책과 SW안전 기업 육성 정책 수립을 위한 기초 자료로 활용될 수 있을 것으로 예상된다. SW안전 기술 확보 방안으로 공개SW 개발 방식을 활용을 제언하였으며 SW안전 기업 육성 정책의 일환으로 안전 분야의 공개SW 활성화로 기업 경쟁력 강화할 수 있는 방안을 제언하였다. 이러한 정책적 방안으로 안전 분야에서 수준 높은 연구 개발물을 획득하여 기술 내재화, 인재 육성 등을 기대할 수 있으며 검증된 최신 SW 개발 방식으로 기업 역량을 강화하고 고급 인재를 공급함으로써 기업 경쟁력을 강화를 기대할 수 있다. 최종적으로는 SW안전 기술 확보와 SW안전 기업 육성을 통해 SW 중심의 자동화가 진행되는 제4차 산업혁명의 미래사회의 안전 확보에 기여할 수 있을 것으로 예상된다.

제2절 한계점 및 향후 연구 방향

이번 연구를 수행하면서 가장 아쉬웠던 점은 시간과 예산의 제약으로 공개SW 현황 조사의 범위를 한정할 수 밖에 없었던 것이다. 안전 분야와 연관된 SW 신기술 분야를 인공지능, 빅데이터, 클라우드로 한정하였고 안전 산업 분야도 자동차, 철도, 항공(국방)으로 한정하였다. 이들 분야 이외에도 SW 신기술 분야에서 블록체인을 포함한 많은 공개SW 활용 사례들이 있었고 안전 산업 분야도 선정된 분야 이외에 의료, 에너지, 기계 등 다양한 분야에서 공개SW 활용 사례들이 있음이 확인되었지만 체계적으로 분석할 수 없었다. 또한 국내 안전 분야 현황 조사에서도 30개 기업들로 한정할 수밖에 없었기에 시범 조사 성격으로 진행되었기 때문에 국내 현황 조사 결과 국내 모든 기업들을 대표한다고 할 수 없다.

따라서 후속 연구를 진행하는 연구자가 있다면 이런 한계점을 극복하여 보다 넓은 분야의 공개SW 현황 파악이 필요하다고 보이며 정확한 국내 현황 파악을 위하여 보다 많은 산업 분야와 많은 기업들을 대상으로 전수 조사 성격의 심층 면담 조사가 필요하다고 생각된다. 또한 조사 대상자들도 SW 관련 직군을 포함

하여 기획, 관리, 마케팅 등 기업 전반의 업무를 수행하는 사람들을 포괄해야 한다. 이러한 정보들은 향후 국내 국가 산업 정책 수립 및 기업 사업 전략 수립을 위한 밑거름이 될 수 있는 중요한 자료라고 생각된다.

또한 이번 보고서는 안전 분야에서 공개SW 활용 방안 및 활성화 방안을 제언하고 있지만 국가 산업 전반에 대한 공개SW 활용 및 활성화 방안을 제언하고 있지는 않다. 산업 분야별로 현황이 다르기 때문에 이번 제언에 대한 확대 해석을 경계해야 할 것이다. 정부에서는 약 15년 가량 공개SW 활성화 정책을 추진하고 이를 기반으로 국내 공개SW 저변이 확장되고 있으며 많은 실력있는 전문가들이 양성되었다. 하지만 이번 보고서는 기존의 공개SW 산업이라고 할 수 있는 SW 산업에 대한 조사가 아니기 때문에 이들 산업의 현황과는 차이가 생길 수 있을 것으로 생각된다.

그리고 이번 보고서는 실제 산업 현장에서 관련 정부 정책들이 어떠한 효과가 있는지에 대한 분석을 하고 있지는 않다. 그 이유는 산업 분야가 다르기 때문에 기업 종사자의 필요 역량 및 경험이 다르기 때문에 이에 대한 확대 해석을 할 수 없다고 본다. 따라서 이번 보고서의 제언이 정책화되어 기존 SW산업에서 공개SW 정책이 아닌 안전 분야 또는 제조 분야의 공개SW 정책이 수행된다면 실제 현장에서 어떠한 효과를 가져오는지에 대한 분석이 추가로 필요하다고 생각된다. 이를 통해 이번 보고서의 제언과 연관된 정책들에 대한 향후 분석을 통해 실효성 검증 및 개선 방향 도출을 위한 연구도 필요하다고 생각된다.

참 고 문 헌

국내 문헌

정보통신산업진흥원(2017), “공통 분야 소프트웨어신뢰·안전성 확보를 위한 가이드”,
저작권위원회, 오픈소스 라이선스 종합정보시스템(<https://olis.or.kr>)
정보통신산업진흥원, 공개SW포털(<https://www.oss.kr>)
권영환(2018), 공개 소프트웨어 산업의 이해 - 해외 기업들의 공개소프트웨어 활용, 소프트웨어정책
연구소
텐서플로우 블로그, <https://tensorflow.blog/category/tensorflow/>
Bloter, MS, 오픈소스 딥러닝 툴킷 깃허브에 공개
SK하이닉스 블로그, 도커 기술 소개
네이버 포스트, 쿠버네티스
쿠버네티스, 쿠버네티스 사용자 사례 연구
AUTOMOTIVE Electronics (2018), 바이두 자율주행차 프로젝트 핵심은 ‘HAD맵과 AI’
네이버 블로그, IT 융합 산업의 HW 및 SW 안전 표준 IEC 610508 개요 참조
SW공학포털, ISO 26262 자동차 기능 안전을 위한 MISRA 코딩 규칙 적용 방법
슈어소프트, IEC 62279 기반 소프트웨어 Safety 프로세스 구축 사례 재구성
네이버 블로그, DO-178C 소개와 개요 재구성
방위사업청, 무기체계 SW 개발 및 관리 매뉴얼
과학기술정보통신부, 공개소프트웨어 연구개발 수행 가이드라인
김형체, 오픈소스 & 거버넌스, 인베슝

해 외 문헌

ISO/IEC Guide 51 (2014), “Safety aspects - Guidelines for their inclusion in standards”, ISO/IEC
Smith, David J. and Simpson, Kenneth G. L. (2004), “Functional Safety(A Straightforward Guide to Applying IEC 61508 and Related Standards)”, Hutterwirth-Heinemann
GNU Project, Philosophy of the GNU Project (<https://www.gnu.org/philosophy/philosophy.html>)
Free Software Foundation, Free Software Foundation (<https://www.fsf.org/about/>)

GNU Project, What is GNU? (<https://www.gnu.org/>)

Open Source Initiative, Open Source (<https://opensource.org>)

Autonomous Vehicle Technology(2017), Global automotive infotainment market worth \$40.17 billion by 2024,

EverSQL (2018), Most popular databases in 2018 according to StackOverflow survey

Tensorflow, <https://www.tensorflow.org>.

Revolutions Blog (2017), CRAN now has 10,000 R packages. Here how to find the ones you need,

Jono Bacon (2017), The decline of GPL?, opensource.com(<https://opensource.com>).

statista(2018), Deep learning market revenues in the United States, from 2014 to 2025, by segment (in million U.S. dollars)

Accord.Net Machine Learning Framework, (<http://accord-framework.net/>)

BigData Academy, Hadoop Training in Mumbai

Caffe2, <https://caffe2.ai/>

Amazon AWS, MXNet Architecture

TDWI Research(2011), “Big Data Analytics Report,”.

Wikibon, Big Data in the Public Cloud Forecast, 2016-2026

The R Project for Statistical Computing, <https://www.r-project.org/>

David H Deans(2018), Why the cloud IT infrastructure market is set to reach \$52.3bn in 2018, CloudTech.

Mike Wheatley(2018), Cloud infrastructure giants Amazon, Microsoft, Google, and Alibaba all gain market share, siliconANGLE.

OpenStack, OpenStack Components.

TechTarget, <https://searchcloudcomputing.techtarget.com/definition/Cloud-Foundry>

CloudFoundry, Gateway to the Cloud Foundry Ecosystem

GitHub(2017), The State of Octoverse 2017

Blackduck(2012), BearingPoint, FOSS Management Study.

CHINA INTERNET WATCH (2017), China search engine market share in Apr 2017

Baidu Apollo, <http://apollo.auto/>

embedded (2017), Will China and Baidu’s Apollo accelerate autonomous transportation?

technode (2017), Baidu launches their open platform for autonomous cars-and we got to test it

Pandaily (2018), AI Feast at Baidu Create 2018: Level 4 Autonomous Bus, Apollo 3.0,

DuerOS 3.0

Medium Corp (2018), Apollo3.0:Entering the New Era of Autonomous Driving

Medium Corp, Baidu Apollo Releases Massive Self-driving Dataset: Teams Up with Berkeley DeepDrive

CB INSIGHTS (2018), Android of the Auto Industry? How Baidu May Race Ahead Of Google, Tesla, And Others In Autonomous Vehicles

VOYAGE. <https://oas.voyage.auto/>

LINUX FOUNDATION(2018), Introduction to Automotive Grade Linux

Automotive Grade Linux. <https://www.automotivelinux.org/>

PC World (2016), Automotive Grade Linux: An open-source platform for the entire car industry

Audiokinetic (2018), Integrated Automotive Audio Management

GENIVI(2014). BMW Case Study.

FOSS(2016). GENIVI Development Platform(GDP)

openETCS(2015). Open Proofs Methodology for the European Train Control Onboard System, openETCS co-summit

Military Embedded Systems(2015), Open source software platform released by Lockheed Martin

GitHub, streamflow

Penetration Testing, Laika BOSS: Object Scanning System

NASA Code Open <https://code.nasa.gov/>

주 의

1. 이 보고서는 소프트웨어정책연구소에서 수행한 연구보고서입니다.
2. 이 보고서의 내용을 발표할 때에는 반드시 소프트웨어정책연구소에서 수행한 연구결과임을 밝혀야 합니다.



[소프트웨어정책연구소]에 의해 작성된 [SPRI 보고서]는 공공저작물 자유이용허락 표시기준 제 4유형(출처표시-상업적이용금지-변경금지)에 따라 이용할 수 있습니다.
(출처를 밝히면 자유로운 이용이 가능하지만, 영리목적으로 이용할 수 없고, 변경 없이 그대로 이용해야 합니다.)