

# 월간 SW중심사회

MONTHLY SOFTWARE ORIENTED SOCIETY

04

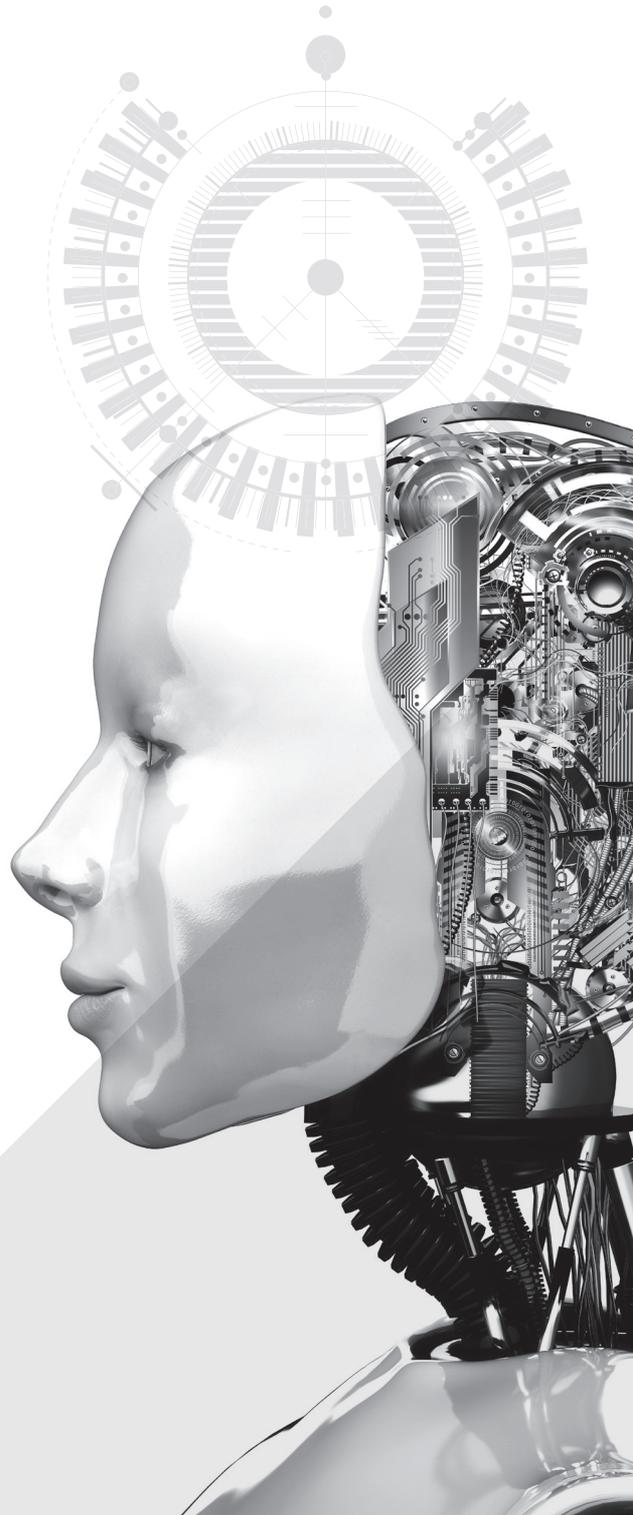
## COLUMN

# 알파고의 충격과 인류의 승리

교육혁신은 거꾸로교실이 해답이다  
 소프트웨어 컨트롤타워 새로 정비해야  
 지능경제시대가 온다  
 인공지능은 어떻게 현실화되는가

## TREND

법률분야에서의 자동화와 일자리 전망  
 구글 검색 알고리즘  
 국내외 다중 채널 네트워크(MCN) 시장 현황  
 스마트 농업 현황  
 인공지능 헬스케어 시장의 미래



# 04 월간 SW 중심사회

MONTHLY SOFTWARE ORIENTED SOCIETY

## 알파고의 충격과 인류의 승리



### C O N T E N T S

# 04

#### 칼럼

- 인공지능 알파고가 남긴 숙제 ①
- 교육혁신은 거꾸로교실이 해답이다
- 소프트웨어 컨트롤타워 새로 정비해야
- 지능경제시대가 온다
- 인공지능은 어떻게 현실화되는가

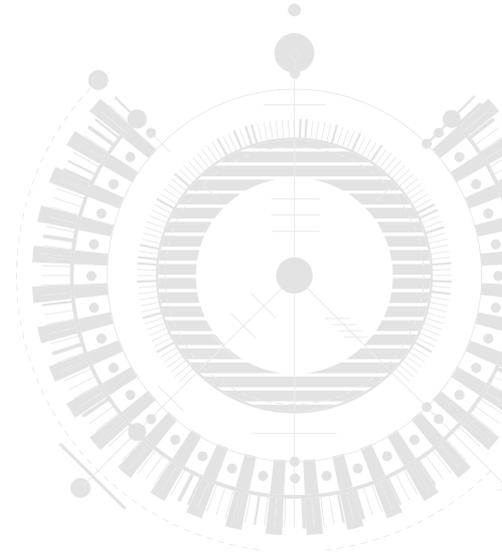


# 24

#### 소프트웨어 산업 및 융합 동향

- 법률분야에서의 자동화와 일자리 전망
- 구글 검색 알고리즘
- 국내외 다중 채널 네트워크(MCN) 시장 현황
- 스마트 농업 현황
- 인공지능 헬스케어 시장의 미래





46

## 소프트웨어 산업 통계

월별 SW산업 생산 및 수출

48

## 이슈

2016년 SW기업 수출, 고용 및 R&D 투자 전망 조사

SW 원격개발 환경 활용방안

92

## 동정

이효섭 연구소장(인코어테크놀로지스) 초청 강연

이희우 대표(Cognitive Investment) 초청 강연

제21회 SPRi Forum

2016 SPRi Spring Conference - AI is here. 성큼 다가온 인공지능



# AlphaGo

## 알파고의 충격과 인류의 승리

### 알파고 충격은 역설적 행운

알파고와 이세돌 기사의 대국은 우리 국민들에게 인공지능의 능력과 한계를 명확하게 보여 주었다. 대국 전에는 이세돌이 전승할 것이란 예측이 우세했었으나 첫 번째 대국에서는 “어 알파고 생각보다 잘하네”하더니, 두 번째 대국에서는 많은 분들이 “와 알파고 놀랍네. 사람이 못 보는 수도 두네”라고 했다. 세 번째 대국도 알파고 승리로 이어지니까 “인공지능이 인간을 지배하나?”라고 하며 충격에 빠졌다. 네 번째 대국에서는 이세돌이 이기니까 “인간 승리”라고 안도하면서도 인공지능이 실수도 한다고 어리둥절하더니, 다섯 번째 대국에서는 “역시 인공지능이 대단하구나”라고들 반응했다.

소프트웨어의 중요성을 역설하고 다니던 인공지능 학자로서 알파고가 인공지능의 능력과 중요성을 전 국민에게 깊이 있게 각인시켰으니 이 얼마나 좋은 일인가? 더구나 알파고가 전승했으면 인공지능에 대한 맹신이 있었을 텐데 한번은 저춤(?)으로서 그 한계도 알려주어서 금상첨화였다.

알파고 덕분에 청와대 간담회에도 참석하는 영광을 누렸다. 이 자리에서 박 대통령은 우리 국민이 받은 알파고 충격을 역설적 행운이라고 평가하며 이를 계기로 혁신을 가속화하자는 취지의 말씀이 있었다. 알파고는 우리 사회에 과연 어떤 숙제를 남겼는지, 우리는 그 숙제를 어떻게 풀어야 하는지 알아보고자 한다.

김진형

소프트웨어정책연구소 소장

[jkim@spri.kr](mailto:jkim@spri.kr)



## 알파고는 인류의 승리

알파고의 성과는 인류의 승리다. 인공지능은 인간 지능의 산물이기 때문이다. 경기의 승패와 관계없이, 이번 대국 그 자체가 인류 역사에서 커다란 기념비적인 사건이다. 인류가 생각을 담을 수 있는 컴퓨터를 발명한 것이 어언 70년. 이후 꾸준히 발전한 컴퓨터과학은 인간 최고수를 능가하는 바둑 프로그램을 만들기 위해 이르렀다. 많은 가능성을 순식간에 검토하여 최적의 한 수를 찾아내는 능력은 계산 속도에 의한다고 하더라도 기보를 통해서 수를 배우고, 기계끼리 두는 바둑에서 좋은 수를 더하는 학습능력이 놀랍지 아니한가? 컴퓨터 프로그래밍을 예술이라고 하지만 아, 알파고는 최고의 예술 작품이다.

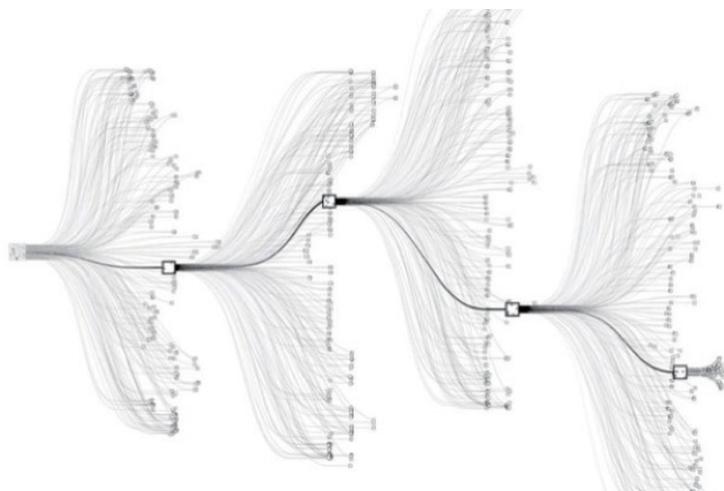
## 컴퓨터와 기계학습은 인류 최고의 발명

컴퓨터는 인류 역사상 최고의 발명품이다. 사람의 생각을 옮겨서 자동화하는 기계이기 때문이다. 사측계산과 비교하는 능력뿐인 이 기계에게 일을 시키는 것은 사람이 작성한 프로그램, 즉 소프트웨어다. 소프트웨어의 명령에 따라 컴퓨터는 빠른 속도로 지시된 작업을 수행한다. 반도체 기술에 의하여 컴퓨터 하드웨어는 손톱만해 졌지만 엄청난 속도와 계산 능력을 갖게 되었다. 컴퓨터가 여기 저기에 탑재되고, 그 위에 소프트웨어로 해결책을 만들어 나누어 쓰는 것이 일상이 되었다.

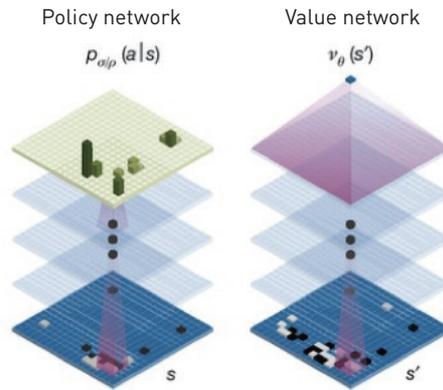
그런데 역사적 사건이 일어났다. 컴퓨터가 학습할 수 있도록 프로그램을 만드는 방법을 인간이 터득한 것이다. 지난 50여년간 인공지능 학자들의 연구를 통해서 점진적으로 발전시켜 얻어진 것이지만 그 성과는 충격적이다. 컴퓨터가 데이터로부터 배운다! 물고기를 주지 말고 물고기를 잡는 방법을 가르쳐주라는 격언이 실현된 것이다.

컴퓨터는 이제 사람이 가르쳐 주지 않아도 데이터로부터 학습하여 프로그램을 만들어 내는 능력이 생겼다. 18년 전 서양장기에서 인간 최고수를 물리친 딥블루는 사람이 만든 프로그램이 판단은 했었지만 이번 알파고는 기보로부터 배운 능력과, 또 자기들끼리의 대국에서 스스로 깨우친 능력으로 대국에 임했다. 기계학습의 가공할 능력을 전 세계인이 같이 본 것이다. 알파고를 만든 개발자들도 그 능력에 놀랐다고하지 않는가!

〈그림 1〉 바둑의 경우의 수는 우주 상의 원자 수보다 많다



〈그림 2〉 알파고의 정책망과 가치망



### 알파고에서의 학습

바둑은 매 순간 놓을 수 있는 수가 평균적으로 200개 정도 된다. 따라서 바둑 판이 만드는 모든 경우의 수는 10의 170승, 즉 10 다음에 0이 170개 있는 큰 수가 된다. 전통적인 탐색기법으로는 아무리 빠른 컴퓨터라고 할지라도 이를 주어진 시간 내에 다 탐색하여 좋은 수를 고를 수 없다. 아찔 수 없이 판세에 따른 어려움이나 확률적 판단을 할 수 밖에 없다. 알파고는 판단을 하는데 세 가지의 학습 결과를 사용하였다. 첫 번째 학습 목표는 주어진 바둑 판의 상황에서 '다음 수를 어디에 놓아야 바람직한가'이다. 3,000만개의 데이터로부터 학습하여 다음 수를 놓아야 할 곳의 확률을 구했다고 한다. 즉 '잘 두는 사람들이 이렇게 두더라'는 것을 배운 것이다. 이를 위해 알파고 팀에서는 딥러닝이란 이름의 신경망 기법을 사용하였고 이렇게 학습한 결과를 정책망이라고 부른다.

정해진 시간 내에 다음 수를 결정하여야 하기 때문에 알파고는 마지막 수까지 탐색하지 못하고 중간에서 중단하여야 할 경우가 대부분이다. 두 번째 학습 목표는 현 탐색 위치의 판세가 나에게 얼마나 유리한가를 학습하는 것이다. 알파고는 이 학습도 16만 개의 기보로부터 딥러닝 신경망 기법을 사용하였다. 이렇게 학습한 결과를 가치망이라고 부른다.

세 번째 학습은 알파고 프로그램끼리의 대국에 의하여 발생한 기보를 이용하여 판세를 학습한 것이다. 이는 사람이 두지 않았던, 그래서 기보로 존재하지 않았던 경우도 학습할 수 있는 강력한 기능이다. 컴퓨터끼리 두니까 쉬지 않고 기보를 모을 수 있다. 매일 3만회의 알파고 프로그램끼리의 대국을 진행시켰다. 작년 가을 프로2단인 영국 프로기사 판후이와의 대국에서 보인 능력보다 이번 이세돌과의 대국에서 알파고가 더 나은 기량을 보였던 것이 학습 때문이라고 판단된다. 물론 판후이와의 대국을 보고 알파고의 실력을 가름하는 것은 무리이다. 왜냐하면 알파고는 많이 이기는 것이 목표가 아니고 확실히 이기는 것이 목표이기 때문에 안전하게 두는 경향이 있기 때문이다.

### 인공지능 70년의 역사

인공지능은 지능이 필요로 하는 일을 기계에게 시키는 방법을 연구하는 컴퓨터과학의 한 분야다. 컴퓨터를 사람의 계산과 생각을 자동화하기 위한 기계라고 본다면 컴퓨터과학이 바로 인공지능 그 자체라고 할 수 있다.

지난 70년간 지속적으로 컴퓨터의 계산 능력을 키우고, 많은 데이터를 모으고 관리하는 기법, 사람이 쉽게 프로그램을 만드는 방법을 연구해 왔던 것이 오늘날 인공지능이 이룬 성과의 바탕이 되었다.

인공지능은 물체를 보고 이해하며, 복잡한 언어를 이용하여 소통하는 사람의 능력을 컴퓨터로 흉내 내고자 한다. 또 복잡한 상황에서 많은 지식과 정보를 취합하여 의사결정하는 방법에 대하여도 연구한다. 불확실한 정보로도 가장 그럴듯한 결정을 내리는 방법도 주요 연구 과제 중 하나였다. 물론 데이터로부터 학습하는 능력, 즉 기계학습은 오랜 연구 주제였다. 체스나 바둑 같은 보드게임은 인공지능 연구의 초기 시절부터 방법론의 테스트베드로 사용되었다. 알파고 이전 최고의 바둑 프로그램은 아마 5단 정도의 수준이었다.

인공지능은 오래 전부터 우리 일상생활에 자연스럽게 스며들었다. 우편물에서 주소를 읽어서 이를 분류하거나, 손으로 쓴 글씨를 인식하고, 세탁 시간을 스스로 선택하는 인공지능 세탁기 등이 대표적이었다. 그러나 요즘에는 더욱 똑똑해 졌다. 사람과 자연스럽게 대화하며 도움을 주는 가상 비서가 이미 여러분의 스마트폰에 들어와 있다. 이를 가정용 로봇과 연결하여 요리 방법을 도움 주기도 하고, 독거 노인의 친구가 되기도 한다. 사진 속 친구의 얼굴을 사람보다 정확하게 인식하고, 로보어드바이저는 어느 펀드메네저보다 실적이 좋다. 무인 자동차는 이미 복잡한 시내를 질주하고, 골프 로봇은 몇 번의 시도로 홀인원을 기록했다. 인공지능 컴퓨터는 퀴즈대회에 나가서 사람들을 모두 물리치고 최고의 상금을 획득하는 능력을 보였다. 방대한 자료를 모아서 이해하고 학습하는 이 능력을 투자 상담이나 병원에서 암 진단하고 처방하는데 사용한다.

## 인공지능이 사람을 지배한다?

이쯤 되면 사람들은 로봇이 인간을 공격하고 지배하거나, 또는 구매한 인공지능 인형이 인간에게 정을 느끼는 공상과학영화를 연상한다. 불안해 하거나 불쾌감을 느끼는 분들이 많다.

프로그램이 프로그램을 '스스로' 만드는 상황, 그래서 인공지능이 스스로 그 능력을 신장하는 상황을 특이 점(Singularity) 상황이라고 한다. 그런 상황에서는 인공지능이 생산하는 지식이 폭발적으로 증가해서 인간이 이해 못하는, 그래서 통제불능의 세상이 될 것으로 예측되기도 한다. 한 연구원은 2045년경이면 그런 상황이 올 것으로 예측했다. 그 때는 "인공지능이 가라사대"라고 하면 누구도 거부할 수 없는 진리가 되는 세상이 될 것이라.



그러나 걱정할 필요는 없다. 현재의 인공지능 수준은 '지시하면' 학습하고 사람의 지능을 흉내 내지만 '스스로' 의지를 갖고 목표를 추구하지는 못한다. 충직한 하인이 더욱 똑똑해 진다고 생각하면 된다. 영혼이 있거나 자아 의식이 있는 인공지능, 사랑이나 증오 등의 감정을 느끼는 인공지능, 그래서 스스로 의지를 불태우는 인공지능은 아직도 과학의 영역은 아니다. 위협할 수 있으니 인공지능의 발전 방향에 대한 관심을 갖자는 데에는 동의하지만 그렇다고 인공지능의 연구를 중지하거나, 발전을 늦추려는 시도는 동의할 수 없다.



인공지능은 도구다. 복잡한 데이터를 분석하여 의사결정을 잘하기 위하여 필요한 도구일 뿐이다. 어떤 목적으로 사용하느냐는 것은 인간의 의지이다. 선한 목적으로 사용하면 득이 되고, 악한 목적으로 사용하면 해가 될 것이다.

본 칼럼은 국가미래연구원 블로그 3월 22일자에 게재된 글입니다.

[http://www.ifs.or.kr/bbs/board.php?bo\\_table=NewsInsight&wr\\_id=879](http://www.ifs.or.kr/bbs/board.php?bo_table=NewsInsight&wr_id=879)



## 교육혁신은 거꾸로교실이 해답이다

지금 당신의 고등학생 자녀가 교실에서 자고 있을 확률이 3분의 2가 넘는다. 어느 교실이던지 무작위로 선택하여 확인해 보라. 앞자리에 모인 몇 명 학생을 제외하고는 모두 엎어져 잠을 자거나 딴짓을 한다. 교사가 자는 학생을 깨우면 '왜 방해하냐'며 오히려 화를 낸다. 이미 교사는 교실의 통제를 포기했다. 권위와 함께 교사의 사기도 바닥에 떨어졌다. 수업이란 것이 엎어져 있는 학생 머리 위로 교과서를 읽어주는 것이 고작이다. 작년에 방영된 TV방송을 보고 놀라서 서울 근교의 학교를 방문하여 직접 확인한 것이다. 학교 교실이 완전히 붕괴됐다. 이런 학생들이 대학에 진학하니 대학교육은 제대로 되겠는가? 대한민국 장래가 암담하다.

학생들은 왜 수업에 관심이 없을까? 상당 수의 학생들은 '포기족'이다. 학업에 관심 없고 미래에 대한 꿈도 없다. 아직 포기 안 한 학생에게도 수업 내용이 학원에서 배워서 다 아는 것이다. 설사 모른다 해도 곧 학원에서 배울 것이라고 생각한다. 학원이 더 효율적으로 시험에 나올 핵심만을 꼭 짚어서 알려 줄 것이라고 믿는다.

우열반 편성이 금지되고, 선행학습은 안되고, 수행평가가 무시된다. 기말 시험 한방이면 높은 등급을 받을 수 있다. 실수 안 하기 경쟁인 수능능력시험 체제에서 잘 배우고, 더 배울 필요가 없다. 이게 입시 위주의 우리 교육 현장의 모습이다.

교육부에서는 내신을 강화하는 것이 교육 혁신의 길이라고 생각하는 모양이다. 내신을 기반으로 수시 선발을 하도록 대학에 권하고 있다. 실제로 2017학년도에 대학정원의 69.9%, 24만 8천 명을 수시로 선발했다.

대학입시가 일회성 수능 성적이나 시험 중심의 내신에만 의존하는 것은 바람직하지 않다는 비판이 일자 입학사정관제라고 알려진 학생부종합전형 제도를 도입했다. 학생부와 자기 소개서가 입학을 결정한다. 2017년 총 정원의 20.4%가 학생부종합전형을 통하여 대학에

김진형

소프트웨어정책연구소 소장

[jkim@spri.kr](mailto:jkim@spri.kr)

입학했다. 맞는 방향이다. 다양한 교과목에서의 역량과 특기, 생활 태도 등을 다면적으로 평가하는 것이 바람직하다. 선진 대학에서는 이미 오래 전부터, KAIST도 30년전부터, 이런 방법으로 학생을 선발한다.

그러나 대학에서는 고교의 학생부가 부실하여 신뢰할 수 없다고 반발이 크다. 자기소개서에는 외부 수상 경력을 일절 기록할 수 없게 되어 있다. 더구나 자기소개서를 써주는 학원도 있다고 한다. 이러니 '근거 없는 주관'으로 학생을 선발하도록 강요 받는 것 아니냐는 대학의 불평에 수궁이 간다. 공교육도 살려야 하고 동시에 학생 선발권을 대학에 되돌려 주어야 하는데 이도 저도 못하는 판국이다.

다행히도 일부 뜻 있는 교사들이 이런 상황을 타개하기 위해 나섰다. 수행평가를 규정이 허용하는 한도 내에서 제대로 한다는 것이다. 수행평가란 학생이 직접 행하는 것을 측정하는 평가방법이고 현행 규정은 내신에서 수행평가를 70%까지 반영할 수 있다. 교사가 수행평가를 하는 것은 당연한 것이지만 우리 교육계에서는 해야 하는 것을 했다는 것이 이야기 거리가 된다. 수행평가가 교육정상화에 미치는 효과는 매우 크다.

중간고사와 학기말 고사만으로 평가하라는 학부모들의 요구를 뿌리치기는 쉽지 않다. 숙제를 내주면 학원 갈 시간 없다고 학부모가 항의한다. 그러나 뜻있는 교사들은 숙제를 주고 학생들이 팀으로 탐구 프로젝트를 수행하게 한다. 학생들 스스로 문제를 제기하고, 자기 나름대로의 방식으로 해결책을 탐구하게 한다. 그 과정에서 스스로 포트폴리오 쌓는 것을 도와준다.

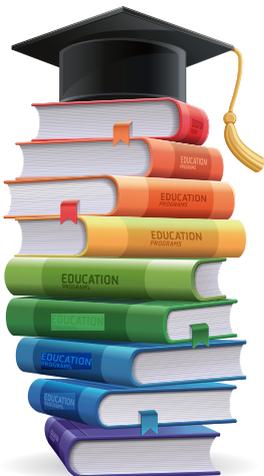
이런 교사들은 밤을 세워 가며 학생 개개인의 성과를 평가한다. 이 과정 중에 학생의 талан트를 발견하고 이를 구체적으로 수행평가서에 기록한다. 학생 개개인에 대하여 할 이야기가 넘친다. 데이터를 기반으로 자신 있게 주관적으로 학생을 평가하여 자존심과 명예를 걸고 학생을 추천한다.

이런 교사들의 제자들은 입시에서 좋은 결과를 얻는다. 탐구 프로젝트에 몰입했었기 때문에 학생들은 입학사정관의 어떠한 질문에도 깊이 있는 대답을 할 수가 있다. 이런 운동에 참여한 지방의 한 생물교사는 KAIST 등에 20명의 제자를 진학시켰다고 자랑한다. 수행평가는 대학입시를 '근거 있는 주관'이 작동하게 하는 뇌관이다. 교사의 성의 있는 평가가 교육 혁신 선순환의 시작이다.

그러나 평가보다 더 중요한 것은 잠자는 학생들을 깨우고 교실에 활력을 불어 넣는 것이다. 지긋지긋한 암기위주의 학교 교육, 이제는 바뀌어야 한다. 인터넷상에서 검색하면 되는데 왜 시시콜콜한 것을 외워야 하는가? 600년 전의 방식으로 100년 전이나 필요한 사람을 키우고 있다는 우리의 교육을 어떻게 바꿀 것인가?

교육기본법에 의하면 우리의 교육이념은 홍익인간(弘益人間)이다. 교육을 통하여 모든 국민이 인격을 도야(陶冶)하고 자주적 생활능력과 민주시민으로서 필요한 자질을 갖추게 한다. 인간다운 삶을 영위하면서 민주국가의 발전과 인류공영(人類共榮)의 이상을 실현하는 데에 이바지하게 함을 교육의 목적으로 명시하고 있다.

그러나 시대의 변화에 맞추어 교육의 목표는 더 구체화하고, 교육의 내용과 방법에서 혁신이 있었으면 한다. 비판적 사고(Critical Thinking), 소통(Communication), 협동(Collaboration) 능력을 갖추고 창의력(Creativity) 있는 인재를 키우는 것으로 4C교육을 목표를 제안하고 싶다.



교육 내용은 새 시대에 필요한 것으로 지속적으로 바뀌어야 한다. 과학기술의 비중을 높이는 것이 필요하다. 새로운 발견과 새로운 기술이 넘쳐나니 옛 것은 축약하고 새로운 것을 배울 시간을 할당해야 한다. 교과목과 그 내용을 교육 소비자의 입장에서 지속적으로 개선 하여야 할 것이다.

교육의 방법은 무엇보다도 학생들이 재미를 느끼고 스스로 참여하도록 혁신해야 한다. 이를 위하여 다양한 스마트 미디어를 활용하는 것이 바람직하다. 다행스럽게도 교육 현장에서 혁신운동이 일어나고 있다. '거꾸로 교실' 운동이 바로 그것이다. 최근 공영방송에서의 '거꾸로교실' 현장 보도를 보면 감동적이다. 교사들이 자발적 노력으로 교육의 방법을 바꿈으로써 교실의 변화를 가져오고 있다.

〈그림 1〉 거꾸로교실 수업에서 팀별 탐구하는 모습



지금까지의 학습 방법은 교실에서 처음 배우고 집에서 복습하는 것이 일상적이었다. 거꾸로교실이란 이 것을 뒤집은 것이다. 교사가 제공한 동영상으로 미리 과업 내용을 학습한 후에, 학교 수업에서는 질문과 토론, 탐구 프로젝트, 또는 미션을 수행한다. 일반적으로 지식 전달만 받던 학생들이 이제는 스스로 참여하고 협동하여문제를 해결하고, 지식을 발견하며, 그 과정에서 창의력을 키운다. '가르침의 종말'이라는 표현이 나올 정도이다.

교실은 그룹 별로 대화와 토론으로 항상 시끌벅적하다. 공동 프로젝트를 수행하는 과정 속에서 공동체의 의식을 갖게 된다. 책임감, 경청, 배려 등 협동하는 방법을 터득한다. '교육은 경쟁이 아니다'라며 먼저 터득한 학생이 팀원들을 지도하는 사례도 종종 볼 수 있다. 한 교실 안에서 동시다발적으로 다양한 수준의 활동이 벌어진다.

학생들이 너무나 즐거워하며 적극적으로 참여하고 부모들도 좋아한다. 전통적인 평가에서도 거꾸로교실의 효과가 좋다는 연구 결과가 많다. 시험 성적도 오른다는 것이다. 거꾸로교실을 운영해 본 교사들은 교실의 변화에 흥분한다. 주입식강의에서 뒤쳐졌던 학생들의 숨겨졌던 능력을 발견하곤 감격하여 눈물을 흘린다. 먼저 경험한 교사들은 스스로 동료들에게 거꾸로교실을 권하는 전도사가 된다.

올해부터 중학교에서 전면 시행하고 있는 자유학기제도 기대해 볼만하다. 토론, 실습 등 학생 참여형으로 수업 운영을 개선하니 학생들이 재미있어 하며 꿈과 끼를 맘껏 발휘한다고 한다. 학생들의 다양한 잠재력을 발굴하는 것이 중등교육은 가장 큰 목적이 아닐까? 다양한 진로의 탐색 활동이 가능하도록 배려하는 것이 중요하다. 자유학기제가 학교 현장에 잘 정착되기 위하여는 기업, 전문가, 학교, 정부의 협조가 절대적으로 필요하다.

학생들은 문제를 스스로 해결하고자 노력하는 과정 중에서 인터넷을 지식의 창고로 적극 이용하게 된다. 또 과제물을 제작하기 위하여 동영상 제작 등 멀티미디어 활용 능력을 스스로 깨쳐 배운다. 교사와 학생들

이 교실 내외에서 SNS(Social Network Service)를 적극 활용한다. 교실 환경에 맞춘 SNS는 교재의 배포, 학생간의 소통은 물론이고, 교사와 학생간의 소통도 원활하게 한다.

〈그림 2〉 자유학기 직업 탐구 현장



수업에 사용할 디딤돌 동영상은 수업을 담당한 교사가 공급해야 한다. 물론 인터넷에 많은 자료가 있지만 교사가 가르칠 내용을 동영상으로 미리 제작하는 작업은 만만치 않다. 그러나 젊은 교사들이 열정으로 도전하고 있다. 교육 당국에서는 이 작업을 도와야 한다. 장비와 소프트웨어를 제공하고, 자유로이 교육 자료가 공유되는 인터넷 환경을 제공해 주어야 한다.

거꾸로교실의 전제 조건은 수업 중에 자유로이 사용할 수 있는 무선인터넷이다. 모든 교실에서 인터넷 검색은 물론 디딤돌 동영상을 학생 전체가 원활히 시청할 수 있어야 한다. 우리 학교들에서의 무선인터넷 구축 상황을 보면 WIFI가 되어 있다고는 하나 특별 교실에서만, 그것도 소수의 동시 사용자만이 허용된다. 하루 속히 모든 교실에서 무선인터넷을 마음대로 사용할 수 있는 환경이 구축되기를 촉구한다.

다행히 무선 외장하드디스크를 사용하면 통신사 망을 거치지 않고도 교사가 교실단위 WIFI 환경을 제공할 수 있다. 인터넷 검색은 안되지만 교사가 만들어 온 디딤돌 동영상을 학생들이 시청할 수 있다. 가격도 저렴하고 사용법도 간단하다. 궁극적인 해결책은 아니지만 거꾸로교실 운영을 위하여 임시적으로 사용할만하다.

그런데 당국에서는 학교 전체가 보안 지역이라는 이유로 무선 외장하드디스크 반입을 제한하고 있다. 당국 이 교사들의 열정에 찬물을 끼얹는 것이다. 교무실의 학사행정 자료는 당연히 강하게 보안되어야 하지만 수업을 진행하는 교실이 보안지역일 필요가 있을까? 보안을 잘하기 위하여는 보안해야 할 것과 보안하지 않아도 되는 것을 확실히 구분하는 것이 우선이다. 교육 당국의 섬세한 정책을 촉구한다.

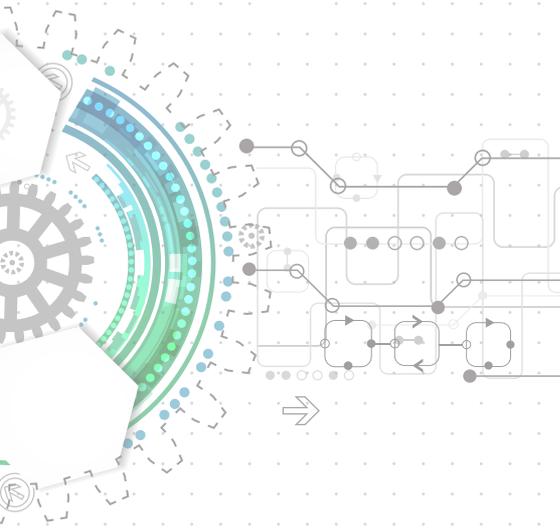
교육혁신은 학생 참여형 수업이 해답이다. 재미있는 거꾸로교실, 여기에 우리 교육의 미래를 건다.



〈그림 3〉 시판 중인 무선 외장하드디스크의 모습

본 칼럼은 국가미래연구원 블로그 2월 23일자에 게재된 글입니다.

[http://www.ifs.or.kr/bbs/board.php?bo\\_table=NewsInsight&wr\\_id=857](http://www.ifs.or.kr/bbs/board.php?bo_table=NewsInsight&wr_id=857)



## 소프트웨어 컨트롤타워 새로 정비해야

소프트웨어 중요성에 대해 재론할 여지가 없는 세상에 우리는 살고 있다. 유비쿼터스 컴퓨팅이라는 용어를 창안한 마크와이저의 표현을 빌리자면 소프트웨어가 사회 곳곳에 스며들어 편재해 있는 상태다. 정보화 사회에서 초연결 사회로, 또 인공지능 사회가 될 것이라고 이야기한다. 이러한 변화를 견인하는 핵심요소가 소프트웨어이고 달리 말하면 소프트웨어에 대한 의존성이 더 높아진다고 할 수 있다.

최근 알파고라는 인공지능 소프트웨어가 보여 준 위력을 고려한다면 개인의 능력에 의존하던 많은 분야에서 곧 인공지능이 활용될 것이다. 다양한 분야에서 자원을 관리하고 활용이 최적화되고 제조분야는 사람의 개입이 최소화된다. 관리자나 의사결정권자는 인공지능의 도움으로 더욱 스마트해질 것이다. 자율주행자동차는 교통체계 및 이동문화의 근본적인 변화를 가져올 것이고 무인화를 이끄는 선도적 역할을 할 것이다. 국방 분야에서도 무기체계의 첨단화에 따라 지능화되고 무인화 되는 비율이 더욱 커질 것으로 예상된다. 미국 전 국방장관 럼스펠드가 말한 것처럼 피를 흘리지 않는 '클린워'가 실현되는 날이 멀지 않은 것 같다.

이러한 예상은 소프트웨어가 정상적으로 작동한다는 가정 하에 가능한 시나리오다. 만약 우리가 의존하는 운영 시스템과 기반체계에서 소프트웨어가 부정확한 결과를 내 우리의 판단을 오도하거나 시스템의 오작동을 유발한다면 치명적인 상황을 초래할 수 있다. 부품이나 원재료가 부족함에도 공장을 가동해 큰 손해가 날 수도, 잘못된 시장 분석 보고서를 기초로 의사결정을 해 큰 돈을 날릴 수도 있다.

유천수

소프트웨어정책연구소 방문연구원

linuxyoo@naver.com



차가 좌표에도 없는 목적지를 향해 돌진하는 사태가 발생할 수도 있다. 최악의 경우 적의 위협이 없음에도 공격을 개시하는 사고를 맞게 될지도 모른다. 이 모든 문제의 근본에는 소프트웨어가 자리 잡고 있다. 최근 자동차, 원전, 철도 등의 산업을 중심으로 시스템 및 소프트웨어 안전성에 관심을 기울이고 있다. 인명의 안전과 밀접한 산업계에서 우선 드라이브하고 있는 형국이다. 특히, 요구하는 국제표준을 충족하고 이를 기준으로 한 안전성 인증 활동에 산업 전체가 나서거나 개별기업이 중심이 되어 유관 기업 간 협업을 진행하고 있다.

자동차의 경우에는 2009년 도요타 렉서스 ES350차량의 급발진 사고에 대해 도요타가 차량 리콜과 과징금을 통해 책임을 지게 된 사건이 계기가 됐다. 비슷한 시기에 기능 안전성 국제 표준으로 ISO 26262가 2011년 제정되었으며 자동차 산업계가 적용에 발 벗고 나서고 있다. 차량이 최대한 안전하게 제조됐느냐는 입증에 따라 배상책임을 면하거나 경감해 주는 제조물책임법을 고려할 때 ISO 26262의 준수는 그 해답이 될 수 있을 것이라 더욱 주목된다.

이 같은 점은 원전의 IEC 61513 이나 철도의 IEC 62279 처럼 안전성이 중시되는 산업에서 동일하게 적용될 수 있는 표준으로 기능하고 있다. 소프트웨어가 일으키는 시스템의 고장이라는 문제 또한 신뢰성에서 생명의 안전을 담보하는 안전성으로 패러다임이 바뀌어 가는 중이다.

우리 사회 전체적으로도 세월호, 메르스 사태를 연속적으로 겪으며 안전에 관한 관심이 고조되어 있다. 부처를 개편하고 대응 매뉴얼을 보완하고 컨트롤타워를 새로이 정비하는 등의 작업이 이뤄졌거나 진행 중이다. 2015년을 기점으로 미래부가 ICT 기반 시스템의 안전 점검을 중요 시스템에 대해 실시하게 된 것도 같은 맥락이라 하겠다. 거의 같은 시기에 소프트웨어 안전에 관한 종합적인 계획 수립이나 소프트웨어 안전 산업을 육성한다는 방향도 함께 고려한 것으로 안다.

하지만 소프트웨어가 모든 산업과 기반 시스템에 쓰이게 되면서 안전에 관한 문제를 소프트웨어에 한정해 바라보기 어렵게 됐다. 즉, 해당 산업의 주요 제품과 기술에 내장된 소프트웨어의 경우 주요 하드웨어의 구성 품이나 완성품과 분리하여 기능과 안전을 다루기가 불가능해진 것이다. 하드웨어에 내장된 임베디드 소프트웨어를 미래부가 아닌 산업부가 담당하게 된 이유이기도 하다. 미래부의 ICT 및 소프트웨어에 관한 정책과 각종 법제도에서 임베디드 소프트웨어가 제외되어 있다는 사실이 이를 보여준다. 따라서 이번 기반 시스템 안전성에 관한 점검에서도 그 역할의 한계가 명확하게 드러났을 것으로 본다.

소프트웨어 산업 전체에서 보면 얼마 되지 않는 테스트와 컨설팅 기업 중심으로 대부분 시스템 차원의 안전성에서 접근해 점검하고 진단하는 일에 완전성을 기대하는 것은 처음부터 무리한 요구다. 더구나 미래부가 주관하는 영역 외의 산업까지 포괄해 시행해야 하는 제도나 사업의 실효성도 제한적일 수밖에 없다.

미래사회에서 무엇보다 중요한 소프트웨어 안전을 확보하려면 소프트웨어를 가장 전문적으로 다루어 온 미래부가 컨트롤타워로서 기능할 필요가 있다. 임베디드 소프트웨어에 대한 정책 또한 전체 소프트웨어 산업의 일부로서 수립하고 발전시켜야 한다. 이를 통해 최소한 소프트웨어 산업 전체의 수준과 관련 통계는 제대로 밝혀질 수 있을 것이다.

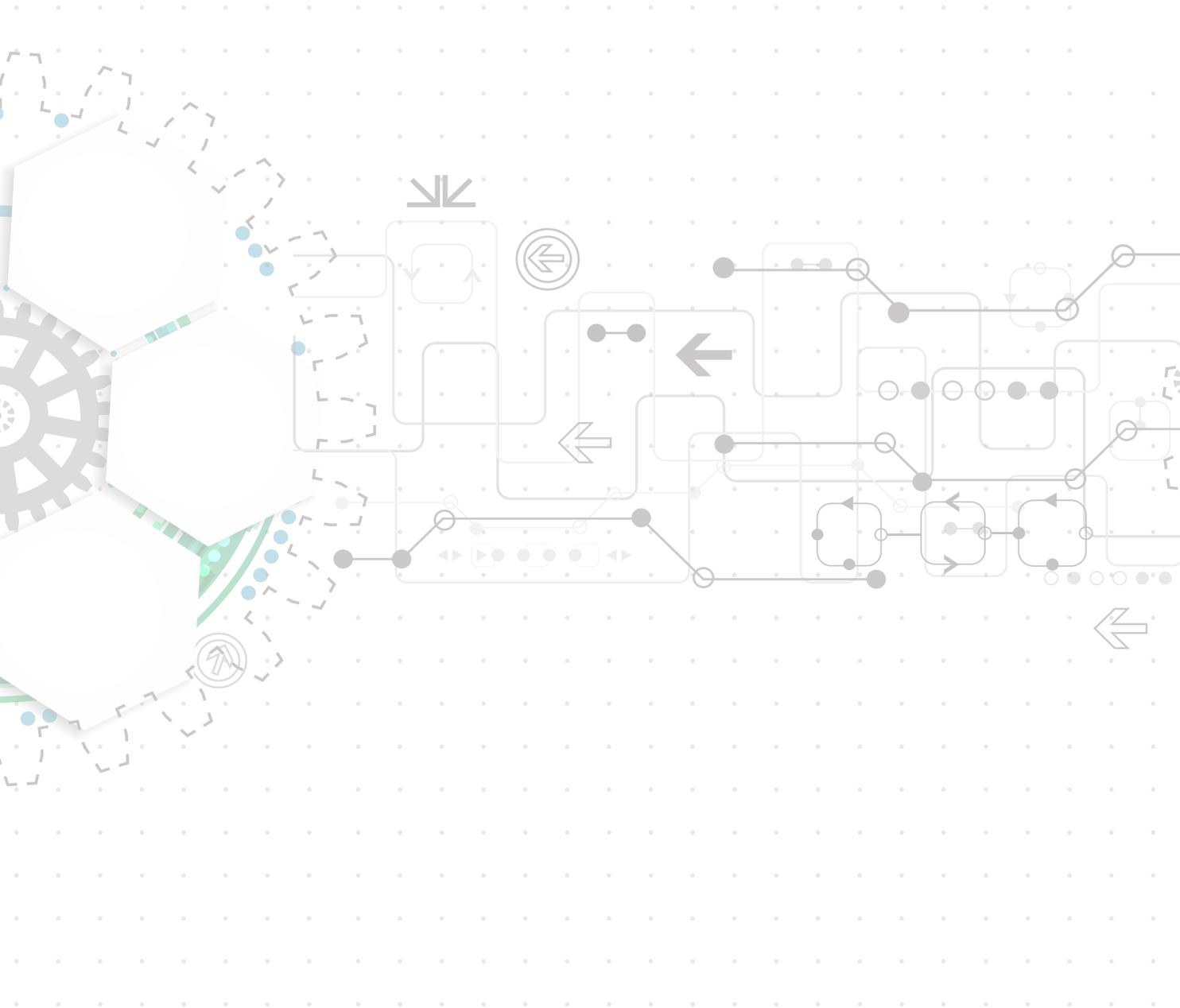
또 미래부는 소프트웨어 산업 정책의 주무부처로서 소프트웨어가 중요하게 쓰이는 모든 산업에 필요한 공통 기술과 지침을 마련해 제공하는 지원자가 돼야 한다. 이를 통해 각 산업별로 별도의 지침과 매뉴얼을 제작하거나 공통적인 기술을 중복 개발하는 낭비요인을 줄일 수 있다. 또 소프트웨어 안전 요구를 분석해 구현

하고 하드웨어와 통합적으로 안전 시험을 수행할 수 있는 임베디드 소프트웨어 고급인력 양성이 필요하다. 이 또한 주무부처인 미래부가 나서서 해결해야 할 과제로 각 산업에 필요한 소프트웨어 인력 공급자의 역할을 담당해야 할 것이다.

이외에도 소프트웨어 안전이라는 새로운 요구사항을 공공 부문 소프트웨어 발주관리 프로세스에 포함해 관리하고 그에 맞는 비용 산정체계 및 예산 항목을 마련하는 일도 중요한 과제다. 새로운 과업의 실행에 필요한 예산이 뒷받침될 때 정부가 이야기하는 안전 산업이 조성되고 일자리도 창출될 수 있다. 소프트웨어 중심 사회를 건설하는 것 못지않게 소프트웨어 사고가 국가사회에 재앙이 될 수도 있는 안전 문제 또한 적극적인 정책과 노력이 필요하다.

본 칼럼은 디지털타임즈 2016년 4월 14일에 게재된 글입니다

[http://www.dt.co.kr/contents.html?article\\_no=2016041402102251607001](http://www.dt.co.kr/contents.html?article_no=2016041402102251607001)





# 지능경제시대가 온다! 모든 것에 지능을

## 개와 늑대의 시간

붉은 노을을 등에 업고 나에게 천천히 걸어오는 실루엣이 개인지 늑대인지 분간하기 어려운 시간이 있다. 서서 지켜보기도, 그렇다고 무턱대고 도망치기도 애매한 순간이다. 조금 더 기다려 얼굴을 확인하고야 반감게 손을 내밀던, 몸을 돌려 전속력으로 도망치던 선택할 수 있다.

구글의 알파고(AlphaGo)가 이세돌을 물리치는 모습을 보면서 인공지능에 대한 태도는, 무관심에서 기대로 기대에서 긴장으로 긴장에서 공포로 변한 듯하다. 한쪽에서는 4차 산업혁명을 가속화시킬 획기적인 생산성의 도구로서 환영한다. 산업혁명과 기계혁명이 인간을 노동으로부터 해방했듯이, 인공지능이 새로운 생산성의 시대로 인도하면서 인간이 좀 더 인간다운 일들에 집중할 수 있는 기회를 제공할 수 있다고 얘기한다. 반면에 반감을 갖는 쪽에서는 인간의 일자리를 빼앗고, 인간을 잉여적 존재, 모든 사회 문제의 근원으로 인식하여 기계에 의해 통제 받는 세상, 즉 디스토피아의 지배자로 군림하게 될 것을 우려한다.

이런 상반된 태도는 기술이 주도하는 산업 혁신의 과도기에는 늘 존재했다.

19세기 초 산업 혁명기에 영국에서 발생한 러다이트 운동이 대표적이다. 러다이트 운동은 기계의 의해 일터에서 밀려난 노동자들의 단순한 반기계 운동이 아니었다. 그 이면에 기계를

유재흥  
소프트웨어정책연구소 선임연구원

[jayoo@spri.kr](mailto:jayoo@spri.kr)



소유하는 자본계급과 소유의 불평등에서 발생한 구조적 계급차별에 대한 저항운동이었다. 이후, 기계는 인류 삶의 필수 생산재로 자리 매김하게 되었고, 인류는 기계와의 공존 방법을 만들어 내기 시작했다. 농업에서 공업으로 그리고 서비스업으로 산업의 축이 옮겨지기 시작했고, 여성의 사회 진출, 교육과 복지의 확대, 소득의 증대로 인한 기본적인 삶의 영위에 있어 이전 보다 나아졌다.

하지만, 지금의 인공지능혁명은 지난 과거의 산업 혁명과는 그 충격이 전혀 다르다는 주장이 있다. 제리 카플란 스탠포드대 교수는 '인간은 필요없다 (Humans Need Not Apply)'라는 저서에서, 현재 상당부분의 일자리(운송업, 단순사무직, 서비스업 등)\*를 차지하고 있는 분야가 인공지능으로 대체될 것은 자명하다고 말한다. 근본적 문제는 직장을 잃은 이들이 새로운 직장으로 유입될 가능성이 별로 없다는 점을 지적한다.

인공지능은 학습을 통해 더 정교하고 정확한 솔루션을 찾을 것이며, 인간의 능력으로 기계의 학습력과 계산을 따라갈 수 없다. 더욱이, 인간의 능력으로 컴퓨터의 학습 방법을 이해할 수 없는 특이점(Singularity)을 지나면 인간보다는 인공지능이 더 신뢰를 받는 기계중심의신뢰 사회가 될 것이라는 주장도 일리가 있다. 지능정보사회의 딜레마가 여기에 있다.

그렇다면, 이러한 변화에 어떻게 대응해야 하는가?

현재까지 어떠한 전문가도 명쾌한 해결책을 내놓지 못한다. 다만, 카플란 역시 지적했듯이 이러한 기술적 흐름이 선악을 판별해야할 대상이 아닌 피할 수 없는(avoidable) 현상이라는 적시할 필요가 있다. 저마다 유토피아의 모습을 말할 순 있겠으나 결국 미래의 사회는 사회 구성원들의 합의에 의한 대응과 그 시행 착오의 결과물이기 때문이다.

분명 무엇인가 성큼성큼 다가오는 것은 분명하고, 얼굴이 드러날 때 쯤 도망칠 곳은 마땅이 없다. 도망쳐보아 따라잡히는 게 시간문제라는 것도 안다면, 현실적 방법은 미리 뛰는 것이다.

## 모든 것에 지능을

인류의 삶을 획기적으로 바꾼 증기기관과 전기 에너지처럼, SW가 앞으로 모든 산업의 범용 생산 요소로 자리매김 하는 것은 시간문제다. 구글, IBM 등의 첨단 SW 기업은 이미 인공 지능 기술을 범용 기술로 확장시키겠다는 의지를 밝혔다. 첫 번째 타겟으로 모든 인류의 꿈인 생명 연장을 위한 의료 분야를 지목했다. 머지않아 Dr.Watson, Dr.AlphaGo에게 진찰을 받는 시기가 올 것이다. 이 인공지능 기술들은 지금도 수 십만 건의 논문과 의학 자료, 그리고 영상 정보를 읽고 학습함으로써 인간 의사를 넘어서는 정확한 판단을 하게끔 훈련받고 있다. 2014년 한 보고에 따르면 IBM의 왓슨은 이미 대장암의 98%, 직장암 96%, 자궁경부암 100%로 인간 의사와 같은 진단을 내놓고 있다(사이언스타임즈, 2016. 3).

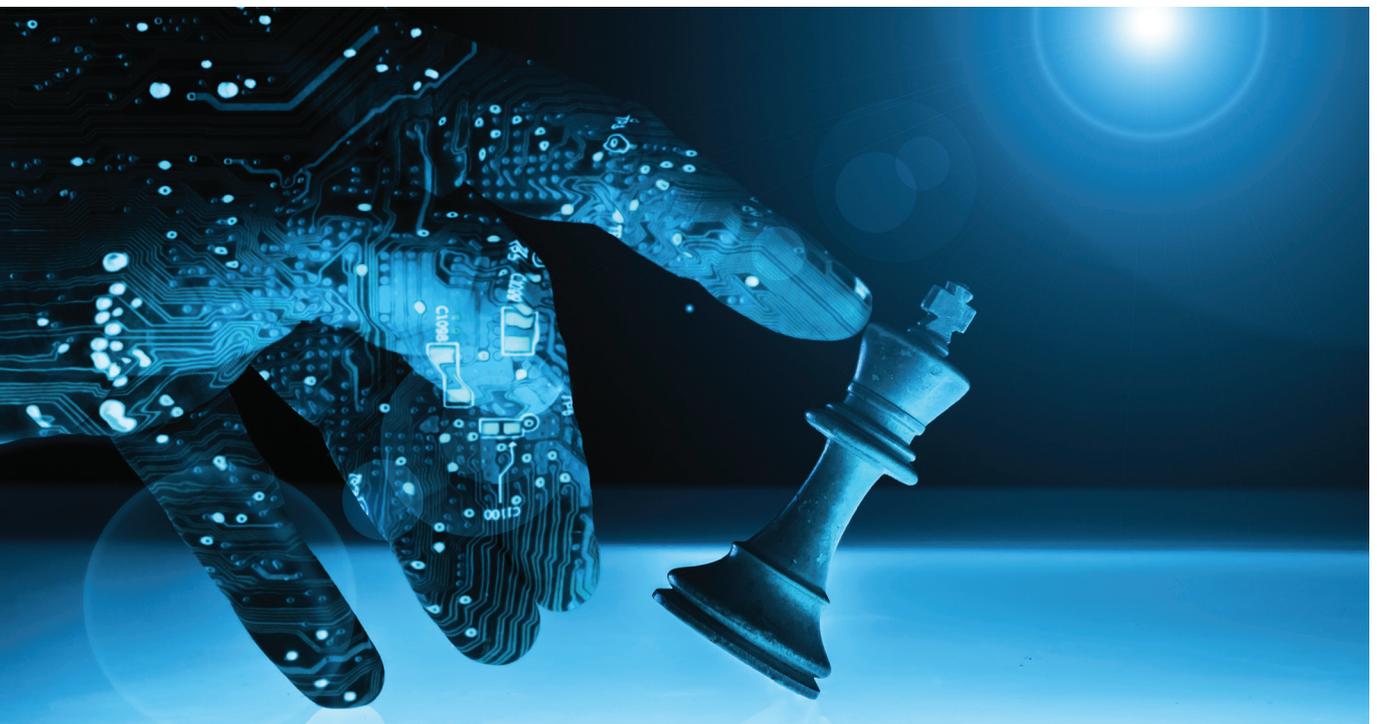
한발 나아가 지난 3월 Microsoft의 CEO 사타니 나델라는 '앱의 시대는 가고 인공지능의 시대'가 왔으며 '모든 것에 지능을 불어 넣겠다'고 천명했다 (매일경제, '16.3). IT왕국의 권좌를 Google과 Facebook에 내어준 Microsoft가 새로운 전략적 목표로 지능기술을 선택했다는 것은 시사하는 무기가 다르다.

모든 서비스가 지능화 되는 현상은 앞으로 미디어, 금융, 제조, 법률, 교육, 심지어 가장 창의적이라 여겨지는 예술 분야를 포함해 전 산업을 망라해 빠르게 진행될 것이다. 이제 단순 오프라인 작업을 디지털로 옮겨놓는 소극적 의미의 디지털 전환으로는 생존을 보장할 수 없다. 데이터에 지능을 불어 넣고 그 안에서 부가 가치 있는 지식을 발견하고 공유함으로써 새로운 서비스 사업으로 확장이 필요하다.

244년 전통의 브리태니커가 위키피디아에 자리를 내준 것은 지식의 깊이가 아닌 방대한 양의 지식을 빠르게 업데이트할 수 있는 SW 플랫폼을 구축했기 때문이다. 미국의 Newsweek에 이어, 영국의 일간지 Independent지도 지난 달 종이 신문 발행 중단을 선언했다. 오늘날의 뉴욕타임즈는 전문 블로거들이 중심이 되어 기사를 만드는 허핑턴포스터보다 덜 읽힌다. 미국인 1/3은 페이스북으로 대부분의 기사를 읽고 있는 요즘, 전통 미디어 매체들은 '편집권'이 소셜 미디어 업체에 넘어가는 것을 지켜볼 뿐이다.

금융분야에서도 인공지능 확산이 이미 진행중이다. 뉴욕타임즈는 2월 27일자 신문에 '로봇이 월가를 침공했다'라는 기사를 내보냈고, '켄쇼라는 프로그램 매매 SW가 50만달러 연봉을 받는 전문 애널리스트가 40시간에 걸쳐하는 일을 수분 내에 처리하고 있다고 소개했다. 실제 골드만삭스에 자동 주식거래 프로그램이 도입된 후 주식거래인 10명의 몫을 프로그래머 1명이 해내고 있다고 추정했다(한국일보).

제조분야에서는 독일, 미국, 일본을 중심으로 자동화가 확산되고 있다. 스마트팩토리 기술은 부품의 주문부터, 조립, 생산, 유통까지 완전 자동화를 목표로 추진된다\*(조선일보, '15.3). 미래의 제조기업이라 불리는 전기차 제조업체 테슬라의 자동화율은 50% 수준이다. 최근 2017년 말 출시 예정인 보급형 전기차 Model3를 36시간 만에 23만 대 이상 선주문을 받았다. 현재 160개 이상의 자동화 로봇이 생산에 투입되고 있으며 자동화율을 더 높여 생산량을 연간 50만 대까지 늘려 전기차 시장을 주도하겠다는 계획이다.



법을 시장에서는 각종 법률을 검색하는 것부터, 질의를 분석해 관련성 높은 법률 근거들을 정리하여 제시하며 결과를 예측까지 한다. IBM의 왓슨은 ROSS라는 이름으로 법률서비스까지 진출하고 있다(세계일보). 교육에 있어서는 이미 개방형 온라인 공개강좌(MOOC)가 혁신을 일으키고 있다. Coursera, edX 등 MOOC 플랫폼을 통해 최신의 정보를 최고의 스승으로부터 원하는 때에 들을 수 있다. 입학 시험도 필요 없다. 심지어 등록금도 필요 없다. 내가 들은 강의를 추적하고 이해한 정도를 파악하여 수준에 맞는 관련 과목들을 추천해 준다. 흥미를 잃지 않고 지속적으로 배울 수 있도록 심리학적 이론 위에 SW알고리즘이 구현된 것이다. 2011년 이후 현재 MOOC에는 전세계 500여 개 이상의 대학, 4200개의 강좌, 35백만 명 이상의 수강하고 있는 교육 플랫폼이 되었다(ClassCentral, 2015).

인간의 가장 고유 능력으로 간주된 예술 창작의 영역도 예외가 아니다. 작곡 알고리즘을 내장한 인공지능 SW들은 성악곡, 기악곡, 협주곡 전체를 15분 정도에 완성한다. 인공지능 작곡의 장점은 인간은 스테레오 타입을 가질 수 있으나 기계는 전혀 다른 방식의 음계와 악기의 조합을 만들어 낼 수 있다는 점이다. 예일대의 컴퓨터 공학과 교수가 개발한 Kulitta라는 프로그램이 만든 100곡을 사람들에게 들려주니 다수가 사람이 만든 곡으로 생각했다고 한다. 구글은 딥마인드에 적용된 뉴럴네트워크의 기술을 이용해 이미지를 만들어 전시회를 통해 1억 원 이상의 판매실적을 보이기도 했다. 영국의 화가 헤럴드 코헨은 캔버스 위에 붓과 물감으로 색체를 완성해 나가는 인공 지능 로봇을 개발하기도 하였다.

현재의 디지털 전환은 생산, 유통, 소비를 전혀 다른 차원으로 옮겨 놓고 있다.

특정 소비자에게 유용하고(useful), 재미있고(fun), 새롭고(new), 믿을 만한 정보(trustable)를 얼마나 많이(sufficient) 빠른(fast)속도로 시각적으로(visual) 제공해 줄 것인가가 기업의 혁신과제이면서 동시에 산업의 경쟁력으로 본격적으로 부상하고 있다. 최고의 지식의 독점을 위한 기업들이 경쟁은 계속될 것이다. 하지만, 지식만으로 부족하다. 구슬도 꿰어야 보배가 되듯, 지식을 엮어 사업으로 산업으로 만들어 낼 수 있는 지혜가 필요하다.

## 지능 위에 지혜를

지혜란 무엇인가? 때를 분별하여 적절한 지식을 사용하는 능력이다. 맥킨지의 보고에 따르면 산업 일선에서 생존과 성장을 위해 고민하는 최고 경영자의 관심은 단순한 개선보다 재창조라고 한다. 문제의 가시화, 최적화, 자동화, 유연화 시킬 수 있는 SW능력 바로 이해하고 이를 전략적으로 활용하여 새로운 지식서비스의 창출(creating)로 연결하고 전통산업의 재탄생(reinventing)의 기회를 제공할 수 있다.

지능경제사회에서 게임의 룰이 바뀌고 있다.

이 게임의 승패에 따라, 승자독식은 더욱 뚜렷해 질 것이며 이에 양극화가 더욱 심해질 것이라는 것이 (제2의 기계시대)에서 에릭 브린올프슨이 일관적으로 제시하는 주장이다. 생산, 유통, 소비에 이르는 비즈니스 가치 사슬의 지능화가 가속화되면서, 기업의 생명 나이가 산업의 재편이 이뤄지고, 국가적 주도권도 달라지게 될 것이다.

산업혁명과 함께 떠오른 영국의 태양이, 기계 전기 시대에 미국에 머물다 새로운 지능정보사회에서 어디로 움직일지 모른다. 해가 조금씩 떠오르면서 저만치 걸어오는 짐승의 어두운 실루엣도 견힐 것이다. 성큼



다가오는 저 무엇이 인간의 반려견이 될지, 혼돈의 파괴자가 될지 단정하기 어렵다. 하지만, 신기술의 본성과 상관 없이 한 가지 믿음은 필요하다. 우마를 길들여 농업혁명을 일궈냈고, 기계와 전기를 조합해 기계혁명을 만들어 내었던 인간의 지혜에 대한 믿음이다.

사회학자 Burt가 주장했듯 혁명기에는 구조적 공백(structural hole)이 있기 마련이며, 그 공백이 새로운 혁신을 만들어낸다(Burt, 1992). SW혁명으로 촉발되는 새로운 경제적 공백기와 전환기에서 지능경제의 강국으로 도약하는 계기로 삼아야 한다. 피할 수 없는 문제라면 실사구시의 정신으로 이득은 취하되 피해는 최소화할 수 있도록 시민과 정부가 지혜를 모아야 할 때다.

## 참고자료

- \* 이동현 (2016.1), SW중심사회에서의 미래 일자리 연구, SPRI 이슈리포트  
- 미국의 47%, 한국은 63% 컴퓨터 대체 가능성이 큰 직군으로 분석되고 있다.
- \* 에릭 브란올프슨, 앤드류 맥아피 (2014), 제2의 기계시대, 청림출판  
- 레이 커즈와일은 2045년을, 에릭 브란올프슨은 그 보다 10년 앞선 35년을 이 시점으로 보고 있다
- \* 한국일보, 50명 직원이 만든 금융분석 로봇, 월가를 위협하다, 2016. 2. 28.  
<http://www.hankookilbo.com/v/3973863d07354b138630e841fe93ab12>
- \* 사이언스타임즈, 인공지능이 바꾸는 미래 의료, 2016. 3. 16. (닥터왓슨)
- \* 세계일보, 법률분야 AI 진출 적합, 2016. 3. 18.  
<http://www.segye.com/content/html/2016/03/18/20160318002906.html> (IBM ROSS)
- \* ClassCentral, <https://www.class-central.com/report/moocs-2015-stats/> (MOOC 2015)
- \* 조선일보, 사물인터넷 선진국' 美·獨엔 100% 자동화 공장까지, 2015. 3. 25.  
[http://review.chosun.com/site/data/html\\_dir/2015/03/25/2015032500462.html](http://review.chosun.com/site/data/html_dir/2015/03/25/2015032500462.html)
- \* 매일경제, 나델라 MS CEO “모든 것에 지능을 불어넣겠다” 2016. 3. 31.  
[http://vip.mk.co.kr/newSt/news/news\\_view.php?t\\_uid=21&c\\_uid=2529410&sCode=21&termDatef=&p\\_page=&search=&topGubun=](http://vip.mk.co.kr/newSt/news/news_view.php?t_uid=21&c_uid=2529410&sCode=21&termDatef=&p_page=&search=&topGubun=)
- \* Burt, Ronald S. (1992). Structural holes: the social structure of competition. Harvard University Press. ISBN 978-0-674-84371-4.



# 인공지능은 어떻게 현실화되는가?

자율주행차의  
성공을 위한 기술과  
법제도의 역할

김윤명  
소프트웨어정책연구소 선임연구원

[infolaw@spri.kr](mailto:infolaw@spri.kr)

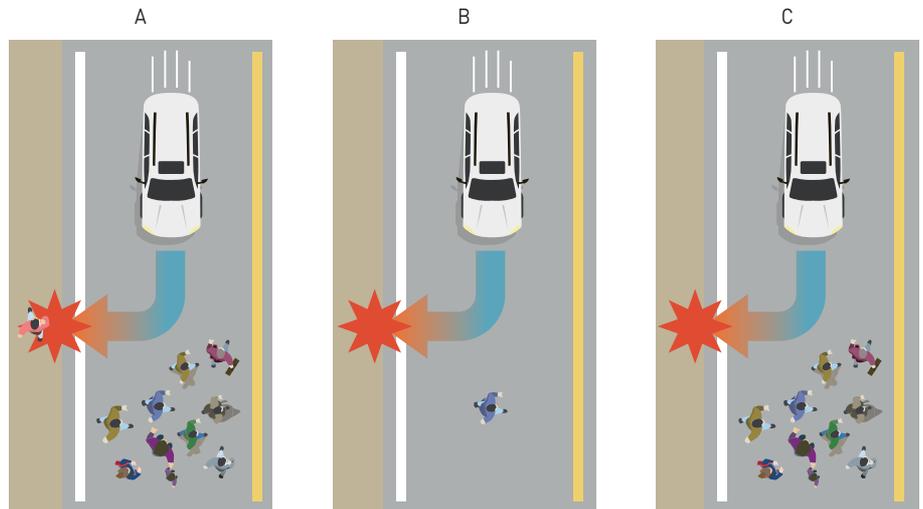
인공지능에 대한 고민의 결과는 단순한 법제의 정비가 아닌 어떠한 법철학이 로봇과 인공지능에 적용되어야 할 지에 대한 것이어야 한다. 그 자체가 SW이며 다양한 네트워크의 연결에 의해 구조화될 인공지능에 대한 법제도적 고민은 SW에 대한 이해와 인간에 대한 깊은 성찰로부터 시작되어야 할 것이다. 현실에서 인공지능은 다양하게 시도되고 있다. 자율주행차가 대표적인 사례가 될 수 있다. 무엇보다도, 바람직한 선례로서 자율주행차의 성공은 인류를 위한 인공지능의 성공으로 귀결될 수 있기 때문이다.

## 윤리는 프로그래밍될 수 있는가?

자율주행차는 인공지능이 사람을 대신하여 운전을 하는 차를 말한다. 2016년 개정된 자동차관리법에서는 자율주행자동차를 “운전자 또는 승객의 조작 없이 자동차 스스로 운행이 가능한 자동차”로 정의하고 있다. 자동차관련 법률에서는 사고로부터의 안전을 위해 사람이 반드시 탑승토록 하고 있다. 아직은 기술 수준이나 다른 환경이 마련되지 않아 사람이 탑승해야 하지만, 특이점(singularity)을 넘는 순간부터는 사람이 대체될 것이다. 이때는 사람의 판단이 아닌 인공지능이 판단을 내리게 될 것이다. 물론, 인공지능의 판단은 기술적이거나 기능적인 수준을 넘어서게 될 것이다. 또한 넘어서야 한다. 자율주행이나 인공지능은 사람의 통제 영역에 벗어나 있기 때문에 고도의 윤리가 프로그래밍될 필요성이 커지고 있다.

그렇다면, 윤리는 프로그래밍화할 수 있는가? 윤리는 상대적인 개념이기 때문에 이를 객관화할 수 있는 것은 아니다. 인공지능이 기계학습을 통해 윤리의식을 높일 수 있는 지는 의문이다. 따라서 사람이 모든 경우의 수를 시뮬레이션하여 윤리적 판단에 대한 설계를 해야하는 것이라면, 이는 불가능한 영역으로 판단 된다. 규제당국은 이러한 전제하에 자율주행차만의 운영을 허용할 수 있을지 의문이다. 다만, 자율주행차량 간의 커뮤니케이션에 따라 주행 중의 사고는 줄일 수 있을 것으로 보이나, 사람과 자율주행차량간의 사고에 대해서는 윤리적 판단이 요구된다.

<그림 1> 트롤리 딜레마



\* 출처 : 구글이미지검색(2016.3.16.)

정의론에서 사례로 드는 트롤리 딜레마(Trolley Dilemma)는 여전히 가치판단의 영역이라는 점에서 인공지능에게 수행토록 하는 것이 타당한지에 대한 규범적 판단을 넘어서 사회적 합의가 필요하다.

### 책임은 누가 지는가?

물론 다양한 경우의 수를 시뮬레이션하여 윤리적인 판단에 대해 자동차의 제조자 또는 인공지능 개발자가 프로그래밍할 수 있을지 의문이다. 다만, 당장 인공지능이 도덕적이거나 윤리적인 판단을 하지는 아닐 것이다. 2016년 2월 발생한 구글 자율주행차의 사고처럼, 일상적인 접촉사고의 수준은 자율주행차라고 하더라도 양해될 수 있다. 그렇지만 궁극적으로 책임문제는 엄격하게 정리될 필요가 있다. 자율주행차에 의해 발생할 수 있는 책임문제는 운전자 과실과 제조자 과실을 들 수 있다. 제3자의 차량에 의한 사고도 예상되지만, 운전자 과실 내지 제조자 과실에 대한 책임 논의와 같이 동일하게 수렴될 것이기 때문이다.

운전자 과실에서 보면, 기본적으로 운전자는 도로교통법상 주의의무를 지게 된다. 운전자는 자연인인지, 아니면 차량 자체인지 확인되지 않는다. 궁극적인 자율주행차의 모습은 인간이 운전에 관여하지 않는 것이 될 것이다. 면허발급의 주체가 지금까지 사람이었다면 이제는 차량이 발급 주체가 되고 있는 것을 통해 알 수 있다. 그렇지만, 발급주체라는 것은 책임의 주체와 같은 개념으로 볼 수 있기 때문에 자동차 사고로 인하여 발생하는 책임을 어떻게 처리할 것인지 논란이 예상된다. 한 가지 방법은 자동차배상법상 형사책임을 배제하고, 손해배상 책임으로 정하고 있는 것처럼 향후에도 유사하게 적용하는 것이다.

또한, 운행 중 외부의 원인이 아닌 차량에 발생하는 사고는 어떻게 할 것인가? 제조자 과실에 대해서는 자동차의 하자에 따른 것을 원인으로 제조물책임을 통해 해결하는 방안을 들 수 있다. 자동차 제조자는 안전한 제품을 만들어야 하며, 경우에 따라서는 제조물책임법에 따른 무과실책임을 질 수 있다. SW와 관련된 제조물 책임에는 임베디드 SW인 경우는 가능하지만, SW자체는 제조물성이 부인된다는 것이 다수적 견해이다. 그렇지만 자율주행차와 같이, SW적으로 운행되는 경우라면 SW는 자동차에 체화된 것이기 때문에 당연히 SW를 포함한 제조물성이 인정될 것이다. 따라서 SW에 대한 제조물책임에 대한 대응책이 마련될 필요가 있다. 자율주행차로 발생한 손해에 대해 자동차 제조자와 SW개발자 등이 연대책임을 지며, 쌍방은 이에 대한 책임소재를 다투게 될 것이다.

〈그림 2〉 벤츠 자율주행차 (컨셉카)



\* 출처 : 구글이미지검색(2016.3.16.)

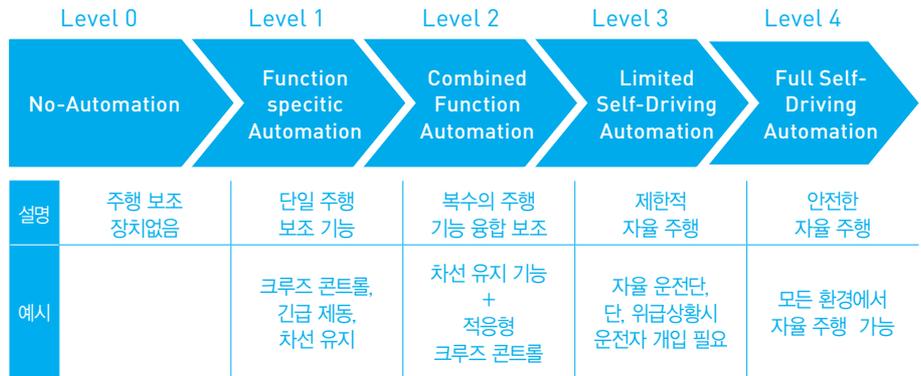
결국, 사람이 개입될 여지가 있는 부분에 대해서는 사람의 판단에 따른 책임소재를 가릴 수밖에 없을 것이다. 사람이 감독해야한다는 것이며 감독자가 책임을 져야한다는 의미이다. 이는 자율주행차에 탑승한 탑승자에게 책임을 묻는 것이다. 트롤리 딜레마처럼 자율주행차의 운행시 나타날 수 있는 사고는 인공지능이 판단할 수 있는 윤리적인 영역에서 가능할 수 있을지 의문이기 때문이다. 다만, 인공지능과 사람의 통제권을 달리하거나 개별적인 통제권을 행사하도록 규범화되지 않은 이상 사람의 통제권을 우선으로 할 수밖에 없을 것이다. 인공지능의 판단에 따른 결과에서 그 책임은 소유자에게 물을 수밖에 없기 때문이다.



### 자율주행차를 대하는 자세 - 법과 기술의 바람직한 관계

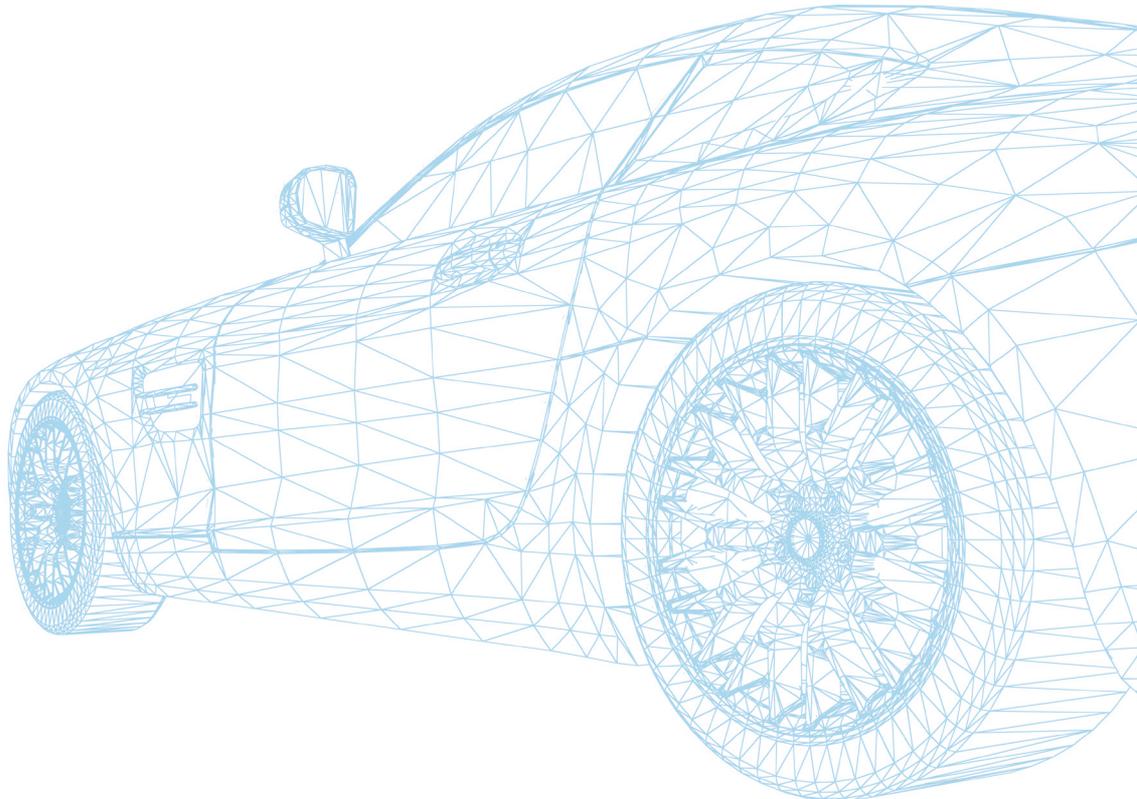
법이 기술이나 사회현상을 따르지 못한다는 비판은 어느 정도 타당성을 갖는다. 2020년을 기점으로 상용화가 진행될 자율주행차는 그 동안 많은 법제도의 정비가 이루어질 것으로 보인다. 대표적으로 자율주행차의 전제는 자율주행차간 커뮤니케이션을 위한 정보의 공유와 교환이다. 어떤 형태로든 차량간 정보교환이 되지 않을 경우에는 자율주행이 불가능할 수 있기 때문이다. 현행 개인정보보호법이나 위치정보법은 이를 불법으로 볼 수 있기 때문이다. 하나의 예이지만, 이러한 법적 한계를 극복할 수 있어야 할 것이다. 다만, 법이 선도적으로 기술을 시뮬레이션하여 대응하는 것은 법적 안정성이라는 측면에서 문제가 될 수 있다. 확정되지 않은 현상과 기술에 대해 법적 재단을 할 경우, 기술이나 현상에 대한 유연성이 떨어질 수 있기 때문이다. 이러한 접근법은 기술의 발전에 저해되며 자칫 규제로 작용할 수 있다는 한계를 지닌다.

〈그림 3〉 자율주행성의 단계



\*출처 : 미국 도로교통안전국(NHTSA)

따라서 기술현상에 대해서는 정책적 접근을 통해 유연하게 대응하는 것이 타당하다. 자율주행차의 경우도 다르지 않다. 그 동안 기술수준을 높이고, 개발자의 윤리의를 높이는 것이 필요하다. 배기가스의 조작과 같은 사례는 소비자의 신뢰를 떨어트릴 수 있기 때문이다. 아울러, 자율주행차를 포함한 인공지능로봇에 대한 소비자에게 안심을 시키는 것이 필요하다. 유연한 정책은 바로 시장과 소비자의 법적 안정성을 담보해야 하기 때문이다. 물론 점진적인 법제도의 정비를 위해 가칭 「지능사회 대응을 위한 특례법」과 같은 특별법의 제정을 통해 국가적 대응체계를 마련하는 것도 고려할 필요가 있다.





# 법률분야에서의 자동화와 일자리 전망

이현승 선임연구원

hslee94@spri.kr

- 최근, 로봇과 인공지능 기술이 발전하면서 미래에 일자리가 줄어들거나 직무에 변화가 생길 것이라는 연구결과가 속속 나오고 있음
  - 옥스퍼드 대학의 프레이교수와 오스본 교수는 2013년 논문에서 미국의 702개 직업에 대해 컴퓨터화 가능성을 계산하여 전체 직업종사자 중 47%가 고위험군에 속한다는 연구결과를 발표하였고, 가트너는 2025년에 전체 일자리의 1/3이, 2030년에는 90%가 대체된다는 시나리오계획을 내놓았음<sup>1)</sup>
  - 한편 MIT의 데이비드 오토교수는 기계가 인간의 노동력을 대체하는 정도가 과장되어 있다는 입장이고 로봇공학 전문가인 로드리 브룩스는 지루한 업무를 대체하는 정도라는 긍정적 견해를 제시하며, 딜로이트는 전체 프로세스나 일자리를 모두 자동화하는 사례는 극히 드물고 가까운 시일 내에 일반화될 가능성도 낮다고 보고 있음
- 변호사, 판사, 검사로 대표되는 법률 분야에 대해서도 상반된 시각이 존재
  - 미국 등에서는 인공지능 기술을 활용해 소장작성, 대화형 법률서비스, 법률안 제정확률 예측 등의 서비스가 제공되고 있고, 국내에서 수사기관들은 다양한 프로그램을 활용하면서 빅데이터를 활용한 범죄예측 시스템도 추진 중
  - 딜로이트는 자동화기술로 인해 영국 법률산업의 일자리가 114,000개 사라질 것이라는 전망을 내놓았지만, 월스트리트저널은 변호사 직종이 오히려 서서히 증가할 것으로 전망하고 있고, 옥스퍼드대의 연구결과로는 미국 변호사와 판사의 컴퓨터화 확률이 다르거나 영국의 경우에도 달라서 서로 견해가 상반됨
- 가까운 미래에도 자동화 기술은 법률 분야에서 인간의 보조적 역할에 그치나 생산성 향상에 따른 전체적인 수요변화는 불가피할 것으로 전망됨
  - 변호사, 판사, 검사의 업무는 다양하기 때문에 업무의 일부분의 자동화로 보다 중요한 업무에 집중하게 될 것임
  - 발전된 자동화 기술의 활용은 기존 종사자들의 이해득실 보다는 실질적 사법정의구현을 위한 사회적 합의에 기초하여 결정되어야 함



## ■ 자동화로 인한 미래 일자리 변화에 대한 연구들

2013년에 옥스퍼드 대학의 프라이(Carl Benedikt Frey) 교수와 오스본(Michael A. Osborne) 교수는 미국 내 702개 직업에 대한 컴퓨터화 가능성을 계산하여, 미국 전체 직업종사자의 47%가 컴퓨터화 가능성 0.7 이상의 고위험군에 속해 있다는 연구결과를 발표함<sup>(2)</sup>

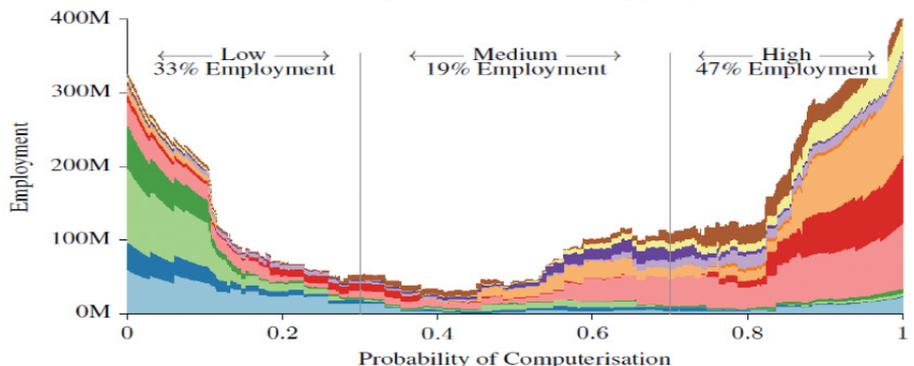
- 이 연구는 미국 노동·고용·훈련관리국에 제공하는 특정 직업에 대한 기술, 지식수준, 실무경험, 자격 요건 등의 직업정보를 토대로 각 직업의 컴퓨터화 가능성을 계산하였는데, 구체적인 방법은 다음과 같음
- 먼저 기계학습의 전문가들이 가까운 미래에 컴퓨터화 유무를 판단할 수 있는 70개 직업을 선정하여 컴퓨터화 유무(전체 자동화에 1, 그렇지 않으면 0)를 결정함
- 다음으로 인지 및 조작, 창조적 능력, 사회적 능력 분야의 9가지 특성을 컴퓨터화의 장애요소로 규정하고, 70개 직업의 9가지 특성에 부여된 저·중·고 수치를 활용해, 확률모형으로 나머지 632개 직업의 컴퓨터화 확률을 계산함

〈표 1〉 컴퓨터화의 9가지 장애요소

컴퓨터화 장애요소	직업정보 상 고려요소
인지 및 조작	정교한 손가락 움직임
	손재주
	좁은 작업 공간 및 불편한 자세
창조적 능력	독창성
	순수예술(작곡, 공연, 제작 등의 이론 및 기법)
사회적 능력	사회적 지각
	협상
	설득
	타인에 대한 배려 및 보살핌

- 마지막으로, 2010년 기준 전체 직업종사자 분포에 따라서 컴퓨터화 확률을 감안한 고·중·저 위험군을 분류함 (그림 1) 참조

〈그림 1〉 미국 직업종사자에 대한 컴퓨터화의 영향 추정



(1) <http://www.gartner.com/newsroom/id/2605015>

(2) Frey, C. B., & Osborne, M. A. (2013). The Future of Employment : How Susceptible are Jobs to Computerisation? Oxford Martin School

- 이러한 연구방법을 한국에 적용한 경우에는 SW개발자, 의사, 초등학교 교사, 변호사, 성직자 등이 저위험군에 속하며, 회계사, 세무사, 관세사 등의 전문직을 포함한 약 63%가 고위험군에 속하는 결과가 나옴<sup>(3)</sup>

한편, 가트너는 2013년 9월 발간한 보고서에서 2020년 경에 스마트기계가 부상하며, 2030년 경에는 선망받는 전문직이 몰락하고, 2030년 이후는 현재 존재하는 직업의 90%가 사라질 것이라고 전망하고 있음

- 이에 따르면 2023년 경에는 의사, 변호사, 중개인, 교수들이 수행하는 고급기술업무의 1/3이 스마트 기계나 인지컴퓨팅기술의 도움을 받는 저숙련 노동자로 대체되며, 2030년 경에는 현존 일자리의 90%가 스마트기계로 대체된다고 전망함

한편, 포레스트 리서치의 2015년 연구결과에 따르면 2025년까지 미국에서 2,270만 개의 일자리가 사라지며, 새로 생기는 일자리를 감안해도 순 일자리 손실은 910만 개에 이를 것으로 전망됨<sup>(4)</sup>

이러한 우울한 전망에 대해, 인지기술의 발달이나 자동화가 일자리의 특정한 업무 측면에만 영향을 끼치기 때문에 일자리 감소위험이 과장되어 있다는 반론도 존재함<sup>(5)</sup>

- MIT의 경제학자 Autor 교수는 생산성 향상에 따른 단기적 고용감소가 결국엔 연이은 고용증가로 상쇄된 역사를 거론하면서, 대학교육이 필요없는 중급 정도의 직업도 다양한 난이도의 업무로 구성되어 있어 일부 업무가 자동화되더라도 인간은 나머지 업무를 계속 맡을 것으로 예상함<sup>(6)</sup>
- 옥스퍼드대의 프레이 교수도 자신의 연구결과에 대해, 많은 일자리가 자동화되었지만 신기술이 노동시장에 미치는 영향을 평가하기는 어렵다는 입장을 표명한 바 있음
- 산업용 로봇 Baxter를 만드는 인공지능 전문가 로드니 브룩스(Rodney Brooks)도 로봇이 인간을 매우 지루한 일자리로부터 해방시켜준다는 긍정적 시각을 제시하고 있음<sup>(7)</sup>

딜로이트도 좁은 영역에서의 인지기술의 적용이 상당한 진보를 이루는 점에 주목하지만, 전체 프로세스나 일자리를 모두 자동화하는 사례는 극히 드물고 가까운 시일 내에 일반화될 가능성이 낮다고 전망함<sup>(8)</sup>

- 다만, 3~5년 내에 일자리의 일부가 인지기술로 자동화될 가능성은 높다고 보고 있음

(3) 이동현, "SW중심사회에서의 미래 일자리 연구:컴퓨터화의 위험과 대응전략", 2015-06호, 2016, 2-4면을 요약정리한 것임

(4) 조용수, "인공지능이 여는 미래세상", 2015. 9. 15. 미래전략위원회 발표자료 참조  
[http://www.ifs.or.kr/bbs/board.php?bo\\_table=research&wr\\_id=281&page=6](http://www.ifs.or.kr/bbs/board.php?bo_table=research&wr_id=281&page=6)

(5) "내 일자리, 과연 로봇이 빼앗아 갈까?", 월스트리트저널 한국판, 2015. 8. 18. 자 기사 참조  
<http://kr.wsj.com/posts/2015/08/18/%EB%82%B4-%EC%9D%BC%EC%9E%90%EB%A6%AC-%EA%B3%BC%EC%97%B0-%EB%A1%9C%EB%B4%87%EC%9D%B4-%EB%B9%BC%EC%95%97%EC%95%84-%EA%B0%88%EA%B9%8C/>

(6) 또한 Autor 교수는 적응력, 일반 상식, 창조성을 요구하는 업무 분야에서 기계로 인간을 대체하려는 시도는 매우 어렵다(immense)고 보고 있는 점에 관해서 Autor, "Polanyi's Paradox and the Shape of Employment Growth", 2014. 9. 3. 참조(<http://economics.mit.edu/files/9835>)

(7) Joseph Guinto, "Machine Man: Rodney Brooks", Boston Magazine, 2014. 11. 참조  
<http://www.bostonmagazine.com/news/blog/2014/10/28/rodney-brooks-robotics/print/>

(8) David Schatsky, & Jeff Schwartz, "인지기술시대의 업무 재설계", 딜로이트 리뷰 17호 한글번역본, 2015. 참조 (<https://dir.deloitte.co.kr/download/nip/ebook/DR/201502/index.html>)

## ■ 법률분야에서의 인공지능기술 또는 컴퓨터 프로그램의 활용 사례

국내에서는 이세돌과 알파고의 대국 이후 인공지능에 대한 관심이 높아지고 있지만, 국외에서는 이미 인공지능기술을 활용한 법률서비스들이 선보이고 있는 상황이고 국내에서도 유사한 서비스를 개발하는 중임<sup>(9)</sup>

- 미국에서는 소송의 당사자들이 상대방이나 제3자로부터 소송과 관련된 사실관계나 증거자료를 수집하기 위하여 이용하는 증거개시제도 분야 등에서 로펌 주니어 변호사들이 주로 수행하는 문서리뷰를 컴퓨터가 대신하는 소프트웨어 및 서비스, 그 외에 소장이나 신청서를 자동으로 작성해주는 온라인 서비스가 계속 성장하고 있음
- 지능형법률자문회사인 로스인텔리전스는 IBM의 인공지능 ‘왓슨(watson)’을 기반으로 이용자가 일상의 대화체로 질문을 하면 질문과 연관성이 높은 법률적 답변과 함께 판례 등 근거 자료를 제공하는 대화형 법률서비스를 제공하고 있으며, 블랙스톤 디스커버리의 인공지능과 빅데이터 기법을 이용한 지능형 검색 기술은 2012년 벌어진 삼성과 애플의 특허소송에서 사용되었음
- 법률분석기업 피스킵노트는 미국 의회와 정부 데이터에 인공지능 기술을 접목시켜 연방정부와 주정부의 모든 법안과 규제는 물론 상하원 의원들의 영향력 정보까지 기업의 정부정책 담당자에게 실시간으로 제공 하는데, 상정된 법안의 의회 통과 가능성 예측의 정확도가 94%에 달함
- 미시간 주립대 법학교수인 Katz 교수팀은 2014년에 기존에 구축된 미국 연방대법원과 대법관들에 대한 정보, 사건에 관해 알려진 정보들을 토대로 기계학습 기법으로 연방대법원의 판결을 예측하는 프로그램을 만들어서, 7,700건의 판결에 대해 69.7%, 68,000건의 개별 대법관들의 투표결과에 대해 70.9%의 적중률을 기록하기도 했음<sup>(10)</sup>
- 국내에서는 2011년 현직 변호사와 인공지능 전문가들로 구성된 인텔리온 메타연구소(대표 임영익 변호사)가 만든 지능형 법률정보시스템인 '아이리스(i-LIS)'가 내년에 베타서비스를 시작할 예정임

한편, 국내의 검찰과 경찰은 업무과정에서 다양한 컴퓨터 프로그램들을 활용하고 있고, 경찰청과 미래부는 과학치안 구현에 ICT기술을 적극 활용하기로 하는 업무협약을 체결하면서 빅데이터를 활용한 범죄예측시스템 개발에 나서기로 함

- 수사기관은 지오프로스(GeoPros), 마디모(MADYMO), PGS 등을 활용 중임<sup>(11)</sup>
- 또한 경찰청과 미래부는 빅데이터(big data) 분석을 통한 실시간 범죄 예측·예방 시스템 개발을 위해, 미래부는 범죄 프로파일링 정보와 실시간 정보(시시티브이)(CCTV), 음성, 인터넷(internet) 등 융합을 통해 실시간으로 범죄 예측이 가능한 알고리즘(algorism)과 지도서비스 개발을 추진하고, 경찰청은 방대한 내부데이터(data)와 공공·민간 공개 데이터를 종합·분석하여 범죄 발생지역, 용의자 등을 예측하기 위한기술을 개발할 예정임<sup>(12)</sup>

(9) 이하의 내용은 “인공지능(AI), 법률서비스에 이미 등장”, 법률신문 2016. 3. 17.자 기사 참조  
링크 : <https://m.lawtimes.co.kr/Legal-News/Legal-News-View?serial=99241>

(10) 미국 연방대법원의 판결 및 대법관의 투표결과와 예측에 사용된 정보들과 기계학습 방법론에 대해서는 DM Katz, “Predicting the Behavior of the Supreme Court of the United States: A General Approach”, 2014.를 참조  
링크 : [http://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2463244](http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2463244)

(11) 지오프로스는 용의자의 거점을 중심으로 범죄 발생을 분석하며 마디모(MADYMO, MAtheMatical DYnamic MOdels)는 교통사고를 재연하여 사고 발생원인 등을 분석하는 프로그램이며, PGS(사건처리정보시스템, Prosecutorial Guideline System)은 수백만건의 법원 판결을 분석해 피고인 전과, 피해역수 등 양형인자를 입력하고 평균치를 산출해 적절한 구형량을 결정하는데 활용됨 “청년변호사들이 본 알파고...‘보험·의료 분야 사건에 도용될 것’”, the L 리포트, 2016. 3. 22.자 기사 참조  
링크 : <http://thel.mt.co.kr/newsView.html?no=2016032210328212958&pDepth1=0402>

(12) 경찰청, “경찰청-미래부, 국민안전과 글로벌(Global) 과학치안 구현에 협력”, 2015. 7. 9.자 보도자료 참조  
링크 : <http://police.go.kr/portal/bbs/view.do?bbsId=B0000011&nttlId=17478&menuNo=200067>

## ■ 자동화가 법률분야에 영향을 끼칠 것이라는 입장

위에서 언급된 2013년 옥스퍼드 대학의 연구결과에 따르면 미국의 법률 분야 직업의 컴퓨터화 확률은 3.5% ~ 94.5%로 직업에 따라 큰 차이를 보임

〈표 2〉 미국 법률분야 직업들의 컴퓨터화 확률

컴퓨터화 확률	대체불가능 순위	전문가 판단	직업 명칭	원 명칭
98.0%	672/702	없음	법무비서	Legal Secretaries
94.5%	609/702	가능	법률 보조원 및 유사직역	Paralegals and Legal Assitants
50.2%	301/702	없음	법원 속기사	Court Reporters
40.9%	276/702	가능	재판 연구원	Judicial Law Clerks
40.1%	271/702	불가능	판사, 하급법관	Judges, Magistrate Judges and Magistrates
3.5%	115/702	불가능	변호사	Lawyers

위 연구방법론을 영국 노동청의 일자리 통계와 딜로이트 자료를 활용해 영국에 적용했더니 향후 20년 내에 약 35%의 일자리가 컴퓨터화 고위험군에 속하는 것으로 나오며, 법률 분야 직업의 자동화 확률은 〈표 3〉과 같음<sup>(13)</sup>

〈표 3〉 영국 법률분야 직업들의 자동화 확률

자동화 확률	대체가능 순위	고용현황 <sup>(14)</sup>	직업 명칭	원 명칭
98%	3/366	44,000	법률비서	Legal secretaries
66%	137/366	70,000	법률사무 보조원	Legal associate professionals
3%	320/366	23,000	법정변호사, 판사	Barrister or judge
3%	320/366	45,000	기타 법률전문가	Legal professionals(other)
3%	320/366	107,000	사무변호사	Solicitors

위 연구방법론을 한국고용정보원이 국내 주요 직업 400여 개에 대해 적용한 결과는 〈표 4〉와 같음<sup>(15)</sup>

〈표 4〉 한국 법률분야 직업들의 자동화 확률

자동화 확률	대체가능 순위	직업 명칭
62.3%	142/406	법무사 및 집행관
42.7%	194/406	법률 관련 사무원
34.5%	210/406	법률·경찰·소방 및 교도 관리자
22.0%	251/406	변리사
13.4%	279/406	변호사
5.7%	306/406	판사 및 검사

(13) "Will a robot take your job?", BBC News, 2015. 9. 11.자 기사 참조  
링크 : <http://www.bbc.com/news/technology-34066941>

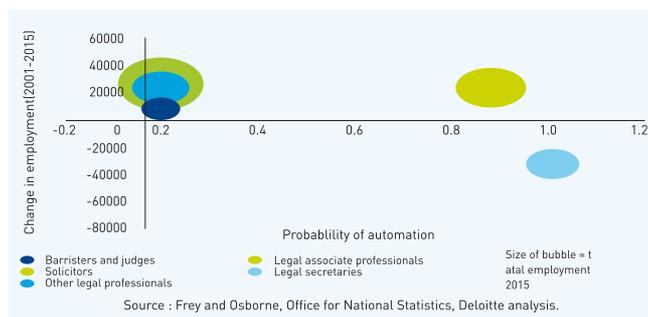
(14) 2015. 9. 기준 영국 총 일자리수는 3,132만명임

(15) "AI. 로봇-사람. 협업의 시대가 왔다!", 한국고용정보원, 2016. 3. 25.자 보도자료 참조  
링크 : <http://www.keis.or.kr/user/bbs/main/203/720/bbsDataView/32701.do>

한편 영국 언론들은 딜로이트 보고서를 인용하면서 향후 10년 내에 법률분야의 39%에 달하는 114,000개의 일자리가 자동화로 인해 대체될 가능성이 높다는 기사를 게재함

- 기사에 따르면, 자동화 기술의 발전으로 법률산업 분야에서 법률비서 같은 저숙련 업무에서 31,000개의 일자리가 사라지고, 보다 다양한 능력을 가지고 좋은 대우를 받는 변호사들의 일자리가 80,000개 가량 증가해 왔음<sup>(16)</sup>
- 하지만, 기사가 인용한 딜로이트 보고서에는 향후 데이터와 기술의 활용도가 높아지면 스마트하고 자기 학습이 가능한 알고리즘이 고급 노동자의 역할 중 반복적 요소가 포함된 일을 대체할 잠재적 위험이 있다고 서술되어 있을 뿐 향후 114,000개의 일자리가 사라질 위험이 있다는 직접적인 언급은 없음<sup>(17)</sup>

〈그림 2〉 딜로이트의 영국 법률 분야 고용상황 및 자동화 전망



## ■ 법률분야는 자동화의 영향에 제한적이거나 바람직하지 않다는 입장

윌스트리트저널은 몇 년 내에 새로운 기술들이 등장하겠지만 고객을 안심시키고 사건수입활동을 벌이는 등 변호사가 하는 많은 업무를 컴퓨터가 맡기는 어렵다고 보면서, 앞으로도 오랫동안 인간이 많은 일을 하게 될 것이며 변호사 직종 자체는 서서히 증가할 것으로 전망함

〈표 5〉 판사, 검사, 변호사 업무개요

업무성격	세부내용
공통	판례, 법령, 논문, 양형기준 검색
판사	사실관계 확정, 증거 취사선택, 주장과 항변에 대한 판단, 법령 적용, 손해배상비용 결정, 조정안 제시, 형량 결정, 위헌위법여부 판단
검사	사건 수사, 범죄 여부 판단, 법령 적용, 증거 수집, 공소제기, 공소사실 입증, 구형
변호사	의뢰인 상담, 법률적 요건사실로의 변환, 법률행위 관련 자문, 소송여부 및 승소가능성 판단, 증거방법 및 증거수집, 기타 소송전략 수립

(16) 딜로이트 보고서 내에는 구체적인 비율과 수치가 없으나 모든 기사들이 동일한 내용을 실고 있음  
<http://futurism.com/new-report-predicts-over-100000-legal-jobs-would-be-lost-to-technological-automation/>  
<http://www.legaltechnology.com/latest-news/deloitte-insight-100000-legal-roles-to-be-automated/>  
<http://www.ft.com/intl/cms/s/0/c8ef3f62-ea9c-11e5-888e-2eadd5fbc4a4.html#axzz44MqmY900>

(17) Peter Saunders, "Developing legal talent : Stepping into the future law firm", Deloitte, 2016. 2.  
 딜로이트 보고서 원문은 다음 링크를 참조  
 링크 : <http://www2.deloitte.com/uk/en/pages/audit/articles/developing-legal-talent.html>

판례, 법령, 논문 등의 검색에 자동화기술이 널리 활용될 것에는 이견이 없는 것으로 보이나, 판사, 검사, 변호사의 업무성격에 따라서 영향이 다르거나 가능하지 않을 것이라는 견해도 존재함

- 판례검색 등의 자동화기술이 일차적으로 사무직원 업계에 큰 영향을 줄 것이고 일반인들도 이 서비스를 활용하면 현재 '나홀로소송' 등 스스로 법률업무를 처리할 때 드는 수고와 어려움을 많이 절감할 수 있음
- 옥스퍼드대의 연구방법론에 따르면이라도 영국, 미국, 한국에서의 판사와 변호사의 컴퓨터화 확률이 다른 것은 각 나라 별 직업의 구체적인 업무에 따라 영향이 다를 수 있는 한 예임
- 또한 거짓말 탐지기 조차 인권침해 시비가 있는 마당에 기계가 사람을 대신해 수사를 하는 것은 어불성설 이라거나, 사람이(판사 역할의) 기계의 판단을 신뢰할 수 있는가는 기술 발전과는 별개로 사회적 합의가 필요한 문제라는 입장도 존재함<sup>(18)</sup>

## 정리와 전망

미래 일자리에 대한 논의는 객관적 근거에 의한 추정, 혹은 전문가들의 예상으로 일정 정도 한계를 지니고 있음

- 옥스퍼드대의 연구결과는 소수 전문가들이 70개의 직업의 컴퓨터화 여부를 이분법적으로 분류해서 나머지 632개 직업의 자동화 확률을 추정했기 때문에 전문가들의 판단오류가 최종 연구결과에 영향을 끼칠 수 있음<sup>(19)</sup>
- 또한 2030년에 일자리의 90%가 없어진다는 가트너의 시나리오에서도 스마트기계는 오직 인간만이 할 수 있었거나 기계는 할 수 없다고 생각되었던 일을 할 수 있다고 보고 있어 현재의 인공지능 기술 발전 추세보다는 보다 가속화된 발전을 상정하고 있음<sup>(20)</sup>

법률분야에서도 기존에 축적된 법령·판례 데이터와 인공지능 기술을 결합해 다양한 서비스들이 나오고 있으며, 이러한 자동화기술의 발전에 따라서 법률사무 직원 등 저숙련 노동자들은 벌써 영향을 받고 있음

- 판례는 숙련된 변호사, 검사, 판사들의 노력에 의해 사건의 사실관계와 법적 판단을 잘 정리해서 담고 있는 수백년에 걸쳐 축적된 데이터이므로, 이들에 대화형 서비스 같은 인공지능기술을 잘 결합하면 의뢰인의 궁금증을 보다 효율적으로 해소해 줄 수 있음
- 또한 법률서면은 서식이 정형화되어 있으므로 법률문서 작성도 많은 부분 자동화할 수 있을 것으로 판단됨
- 특히 국내의 제1심 민사 본안사건의 경우, 2003~2013년 간 변호사 선임비율은 12.5~36.2%에 불과하며 특히 소액사건은 최근 20% 미만 대에 머물러 있어 인공지능의 발달이 나홀로소송을 수행하는데 큰 도움이 될 수 있음<sup>(21)</sup>

(18) "판례 등 자료 탐색 순식간에 ... 법률사무직원 업계는 타격", 법률신문, 2016. 3. 17. 자 기사 참조  
링크 : <https://m.lawtimes.co.kr/Legal-News/Legal-News-View?serial=99234>

(19) 예를 들면, 컴퓨터화가 어렵다고 본 웨이터의 컴퓨터화 확률은 94%, 가정부 및 청소부의 컴퓨터화 확률은 69%에 달함  
링크 : <https://m.lawtimes.co.kr/Legal-News/Legal-News-View?serial=99234>

(20) 보다 구체적으로는 딥러닝 능력, 적응능력, 새로운 규칙을 생성하거나 결과를 학습하는 능력, 가설을 검증하고 새로운 것을 탐지하기 위해 데이터를 수집하는 능력들 중 전부 혹은 일부를 가진다고 서술하고 있음

(21) 이에 관해서는 "변호사 업무의 현황", 적정 변호사 수에 관한 심포지엄 자료집, 2014. 12. 15., 서울지방변호사회, 7면 참조  
링크 : <https://m.lawtimes.co.kr/Legal-News/Legal-News-View?serial=99234>

한편, 법률 분야에서 인공지능을 적용하기 어려운 점은 판사, 검사, 변호사가 다 다를 수 있음

- 검사는 피의자와 참고인들을 직접 대면하면서 여러 수사기법을 활용해 범죄사실을 구성해 내고 입증 방법을 취사선택해서 유죄를 입증하며, 판사는 당사자들 간의 진술을 종합하고 증거들을 취사선택하여 사실관계와 법률관계를 확정한 후 판결을 내리며, 변호사는 의뢰인을 위하여 사실관계를 주장하고 이를 입증하고자 다양한 방안을 강구함
- 따라서 판사의 업무 중 처분문서와 같은 명백한 증거 외에 증인의 증언과 같은 다른 증거방법에 의할 경우에 증거들 간의 신빙성에 따른 취사선택과 증명력의 우월 정도에 대한 비교는 자동화가 어려울 수 있음
- 그리고 변호사나 검사의 업무 중 다양한 증거방법을 생각해 내는 등의 소송전략을 수립하거나, 판사의 경우 손자를 기른 조부모에 법률 상 존재하지 않는 면접교섭권을 인정하는 판결을 내리는<sup>(22)</sup> 등 법조문에 의존하지 않고 인간적 도리나 사회적 정의 실현을 위해 전례없는 판결을 내려야 하는 경우를 어떻게 자동화할 것인지도 관건임

그러므로 당분간은 법률분야에서 인공지능은 기존 법조인들의 여러 업무 중 일부를 자동화하는 등 보조적 역할에 그칠 것으로 전망됨

- 아직까지 거짓말탐지기의 증거능력도 받아들여지지 않는 현실<sup>(23)</sup>에서 인공지능에 의해 법적 판단을 자동화한다는 것은 당분간 요원한 일이지만, 부동산, 교통사고로 인한 손해배상 등의 전형적인 소송사건과 판례·법령 검색 등 보조업무에서는 자동화 기술이 기여할 수 있는 부분이 많음
- 이에 따라 나홀로소송을 하는 국민의 부담도 줄어들 수 있고 판사, 검사, 변호사의 업무도 지엽적이고 반복적인 부분이 감소하여 소송전략이나 위법여부의 판단 등의 보다 중요하고 창조적인 일에 집중할 수 있을 것이나, 생산성의 향상에 따른 전체 법조 인력의 수요 변화는 불가피할 것임
- 또한 법률 관련 직업의 상당부분을 대체할 만큼 자동화기술이 발전되더라도 그 기술의 활용 여부는 사회적 합의에 기초한 사법정의의 실질적 구현이라는 관점에서 결정되어야 하며, 최근 미국에서 마취전문의들의 집단적 반발에 의해 마취유도 로봇이 시장에서 퇴출된 것<sup>(24)</sup>과 같이 이해관계자들의 반발에 좌우되는 것 바람직하지 않음

(22) 현행 민법 제837조의2는 면접교섭권의 주체로 '자(子)'를 직접 양육하지 아니하는 부모의 일방만을 규정하고 있어 법문만으로 보면 손주를 기웠다고 해도 조부모는 원칙적으로 면접교섭권이 없는데도, 딸이 세상을 뜬 이후 손자를 직접 길러온 원고에게 서울가정법원이 면접교섭권을 인정하는 1심 판결을 내렸고 이 판결에 대해 전문가들의 견해는 엇갈리고 있음. 다음의 두 기사를 참고  
 "[판결] '할머니의 기른 정(情)...법원, 조부모 면접교섭권 첫 인정', 법률신문 2016. 2. 23.자 기사 참조(<https://www.lawtimes.co.kr/Legal-News/Print-News?serial=98809>) "조부모 면접교섭권' 허용 놓고 찬반논쟁 '시끌'", 법률신문 2016. 3. 7.자 기사 참조(<https://www.lawtimes.co.kr/Legal-News/Print-News?serial=98859>)

(23) "[이창현 교수의 형사교실] 거짓말탐지기 검사결과의 증거능력', 법률저널 2013. 11. 15.자 기사 참조  
 링크 : <http://www.lec.co.kr/news/articleView.html?idxno=30554>

(24) 세계최대 바이오·제약기업인 존슨앤존슨(J&J)가 미국 식품의약품안전청의 허가 아래 지난해 출시했던 수면유도 마취 로봇인 '세디시스(Sedasys)'가 마취전문의들의 반발 속에서 결국 판매를 중단한 점에 관해서는 "인간의 역할...로봇의 일자리를 뺏다", 매일경제 2016. 3. 31.자 기사 참조 링크 : <http://news.mk.co.kr/newsRead.php?no=235774&year=2016>



## 구글 검색 알고리즘

최근 구글 검색엔진 알고리즘 업데이트의 방향은 인공지능으로, 스팸 웹사이트를 지능적으로 판별하고 사용자의 검색어 의도를 파악하여 양질의 검색결과를 제공

### ■ 구글의 검색엔진 알고리즘

구글 검색엔진은 1998년 PageRank 알고리즘으로부터 지속적인 연구개발로 2016년까지 136건의 업데이트를 통해 세계최고의 웹 검색 성능을 보유

- 검색 알고리즘의 변경의 주요 목적은 크게 스팸 웹사이트를 걸러내는 것과 사용자의 검색의도를 더 지능적으로 이해하는 부분에 있음

- 예를 들면, '재규어'를 검색했을 때 자동차인지 동물인지 사용자의 의도를 파악하여 더 정확한 결과를 보여주는 것
- 'New York', 'New York Times', 'New York Times Square'는 단어 상으로 큰 차이가 없지만 의미에 대한 검색결과는 극명히 다르기 때문에, 세 가지 검색에 대한 적절한 결과가 요구됨

- 구글의 검색엔진 업데이트 Timeline (17년간 136건의 업데이트)

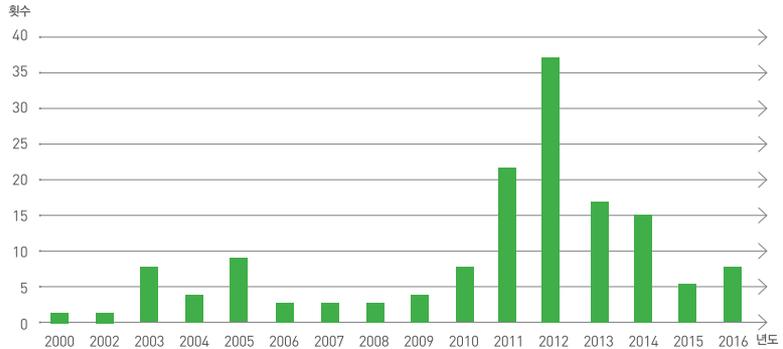
- 실제로 내부 알고리즘이 변경된 횟수는 2009년 350~400회, 2010년 516회 정도로 지속적인 검색성능 개선이 이루어짐
- 검색 알고리즘의 변경이 실제 검색결과에 많은 영향을 미침
  - ※ 2011년 11월 Freshness update는 검색결과의 35%가 변경됨



추형석 선임연구원

[hschu@spri.kr](mailto:hschu@spri.kr)

〈그림 1〉 구글의 연도별 검색엔진 업데이트 횟수



- 〈그림 1〉에서 2013년 이후 업데이트가 감소하는 이유는 구글이 검색엔진의 성능향상을 위해 인공지능 연구에 더 투자를 한 것으로 추정됨<sup>(1)</sup>

성공적인 검색엔진을 개발하기 위해서는 다양한 요소가 필수적임<sup>(2)</sup>

- 웹 수집 : 접근가능한 모든 웹 사이트에 대한 정보수집
- 랭킹 알고리즘 : 웹 검색의 결과를 어떠한 기준으로 어떠한 정보를 먼저 보여줄 것인가에 대한 기준. 구글은 약 200여 가지의 우선순위 부여 알고리즘을 활용
  - ※ PageRank는 랭킹 알고리즘의 한 종류
- 사용자 데이터 : 사용자가 검색한 이력을 바탕으로 사용자의 검색의도를 파악하고 검색엔진의 보완점을 도출해 냄
  - ※ 예를 들면, 'Audrey Fino'는 인물의 이름인데 Fino는 이탈리아어로 Fine이기 때문에 이탈리아 웹페이지가 계속 검색되는 상황이 발생함. 이러한 점을 사용자의 데이터에서 추출하여 검색엔진 업데이트 시 반영함
- 컴퓨팅 파워 : 웹 페이지의 저장과 페이지들 간의 연산, 사용자 데이터 저장 및 분석 등 막대한 규모의 컴퓨팅 파워가 요구됨

■ 구글 검색엔진 알고리즘 변천사 – 사용자의 의도를 파악하는 지능형 검색엔진으로 진화

PageRank 소개

- 1996년 구글 창업자인 Larry Page와 Sergey Brin이 개발한 PageRank는 웹사이트의 중요도를 결정하는 알고리즘으로 1998년 구글의 검색엔진에 도입됨
- 구글의 검색 메커니즘에는 PageRank가 존재하나 구글 검색에서 사용하는 약 200여 가지의 검색 알고리즘 중 하나임
  - PageRank는 구글이 최초로 적용한 검색 알고리즘이라는 상징적인 의미로 중요성을 가짐
- PageRank는 '중요한 웹 사이트는 다른 웹 사이트로부터 많은 링크를 받는다.'라는 가정아래 특정 웹 페이지의 rank를 부여하는 알고리즘
  - rank는 확률값으로 0과 1사이의 값인데, 예를 들어 'A웹 페이지가 0.5의 rank 값을 갖는다.'가 의미 하는 바는 어떤 사람이 랜덤하게 링크를 클릭하여 50% 확률로 A웹 페이지로 이동하는 것을 말함

(1) Google May No Longer Announce Major Algorithm Updates, Search Engine Land, [2015. 03]  
 url : <http://searchengineland.com/fooled-us-google-no-longer-announces-major-algo-updates-217494>

(2) How Google's Algorithm Rules the Web, WIRED, [2010. 02]  
 url : [http://www.wired.com/2010/02/ff\\_google\\_algorithm/all/1](http://www.wired.com/2010/02/ff_google_algorithm/all/1)

- 댐핑 팩터(damping factor)를 도입하여 다른 웹페이지로의 링크가 없는 웹페이지에 도달할 경우, 그 다음은 모든 웹페이지를 접근할 수 있는 확률을 부여함
- PageRank는 고유벡터(eigenvector)값으로 구할 수 있음<sup>(3)</sup>
  - 웹페이지의 관계와 댐핑 팩터를 고려한 정방행렬의 최대 고유치(maximum eigen-value)에 대한 고유벡터로 PageRank값을 구할 수 있음

#### 년도별 주요 사건 및 업데이트

〈표 1〉 구글의 검색엔진 업데이트 내역과 주요 사건

년 도	업데이트 코드네임	주요내용
2001	-	SEO (Search Engine Optimization) 전문가 Amit Singhal 영입
2004	-	Google IPO
2010	Caffeine	새로운 색인(index) 인프라를 업데이트하여 검색결과의 50% 이상을 갱신함 검색결과의 속도를 향상시키기 위해 초당 수십만 웹페이지를 처리 (총 처리 용량은 1억 기가바이트)
2011	Panda	·랭킹 알고리즘의 전반적인 변화: (정보의 질이 낮은 웹사이트에서 검색된 결과나 광고가 많은 웹페이지는 검색결과의 우선순위를 낮추고, SNS나 뉴스 검색은 우선순위를 높임) ·Google Quality Raters로 웹사이트를 평가함 ·2015년 7월까지 지속적인 업데이트 수행 (v4.2)
2012	Penguin	·구글의 웹페이지 검색 정책에 반하는 웹사이트의 랭킹 순위를 낮추는 업데이트 ·의도적으로 검색엔진을 조작하려는 행위에 대한 대책 마련 ·Page Layout 알고리즘 업데이트 : 광고가 너무 많은 사이트나 정보가 너무 적은 사이트는 검색 우선순위에서 제외 ·2014년 12월 까지 지속적인 업데이트 수행 (v3.0)
2013	HummingBird	·새로운 검색 알고리즘 체계 : 약 200여 가지 알고리즘으로 구성되고 PageRank도 이 중 하나로 포함됨 ·'시맨틱 검색'의 구현으로 사람들이 검색하는 검색어나 검색문장에서 의도를 판단함 (자연어 처리 기반)
2014	Pigeon	지역검색 알고리즘과 일반적인 검색알고리즘과의 관계를 강화함 ※ Google 지도와 Google 검색을 관련시킴
2015	RankBrain	인공지능에 기반한 검색엔진 알고리즘 : 검색키워드 뿐만 아니라 키워드의 구성을 바탕으로 사용자의 의도 파악

자료: Google Algorithm Change History에서 재정리 <https://moz.com/google-algorithm-change> Timeline of Google Search, Wikipedia [https://en.wikipedia.org/wiki/Timeline\\_of\\_Google\\_Search](https://en.wikipedia.org/wiki/Timeline_of_Google_Search)

구글의 검색엔진 알고리즘은 양질의 검색결과를 제공하기 위해서 특히 인공지능을 활용하여 사용자의 검색의도를 파악하고, 웹 사이트의 속성과 질을 분류

## ■ 시사점

구글의 검색엔진은 단순히 PageRank로만 이루어진 것이 아니라 스팸 웹 사이트를 걸러내고, 검색 키워드에서 검색 의도를 분석하는 등 최적의 결과를 제공하기 위해 17년간 총 136회의 업데이트를 진행  
최근 구글 검색엔진 업데이트의 큰 방향은 인공지능이기 때문에, 검색엔진에 인공지능이 어떤 역할을 하는지 이해하고 주목해야 할 필요성이 있음

(3) PageRank, Wikipedia, <https://en.wikipedia.org/wiki/PageRank>



## Multi Channel Network



## 국내외 다중 채널 네트워크(MCN) 시장 현황

### ■ MCN (Multi Channel Network) : 다중 채널 네트워크

- 개념정의 : 1인 콘텐츠 창작자들을 발굴, 지원하고 이들을 종합적으로 관리하는 인터넷방송 서비스로 MCN은 1인 창작자들이 콘텐츠를 인터넷에 게재할 수 있도록 플랫폼 역할 뿐만 아니라 방송장비와 스튜디오, 콘텐츠 유통, 광고유치, 저작권 관리, 외부협업 등 다양한 혜택을 지원하기도 함
- 국내기업 : CJ E&M, 아프리카 TV, 콩두컴퍼니, 트레저헌터, 예피스튜디오 등

김준연 선임연구원

[catchup@spri.kr](mailto:catchup@spri.kr)

## ■ 국내 MCN 시장 동향

- 한국전파진흥협회에 따르면 국내 MCN 사업자는 100개가 넘으며, 이 중 유튜브가 인정한 곳은 아프리카TV 등 5개 내외다. 국내 MCN시장이 성숙되면서 동영상 플랫폼만 제공했던 아프리카TV도 2014년부터 MCN사업을 본격 추진했고, 이 외에도 판도라TV, KBS 예띠TV, 네이버 V앱 등은 뒤늦게 진입
  - 국내 MCN 시장 규모는 대략 2,000억~3,000억 원으로 추산되고, 국내 MCN 사업자 중 세계 100위 안에 드는 기업도 CJ E&M(31위)과 트레저헌터(96위) 두 곳뿐
- 2016년 3월 아프리카TV는 인터파크와 공동으로 'BJ 4인 4색 낭만여행'을 콘셉트로 시청자와 실시간으로 소통하는 여행방송을 진행한다. '로이조', '창현', '디바제시카', '버블디아' 등 아프리카TV에서 높은 인기를 끌고있는 유명 BJ들은 국내 숙박업 '체크인나우'를 활용해 여행 계획부터 실제 1박 2일까지의 일정을 생중계 추진
- 2016년 3월 9일 국내 시장에도 MCN 협회가 정식으로 출범해서 본격적인 활동을 시작했다. CJ E&M 미디어솔루션 이성학 부문장이 협회장으로 취임했다.
- KBS의 MCN 채널인 'Yettie Studio'는 중국의 1인 미디어 플랫폼사업자인 롱주(Longzhu)h와 MCN 사업 협력을 합의(2016.3.28.)하고 중국의 1인 미디어 플랫폼에 한류 콘텐츠를 진출시키는 발판을 마련

순위	회사명	한 달간 구독자 증가 숫자	MCN	주요 내용												
1	브로드밴드	3,000만	아프리카TV	· 2015년 9월 인도에 MCN 창작자 전용 YouTube Space 개설하고 인도의 영화 교육기관인 Whistling Woods International과 협력해서 오디오 및 동영상 편집 장비 등을 제공 <table border="1"> <thead> <tr> <th>순위</th> <th>2013년</th> <th>2014년</th> <th>2015년</th> </tr> </thead> <tbody> <tr> <td>매출액</td> <td>367</td> <td>505</td> <td>629</td> </tr> <tr> <td>영업이익</td> <td>43</td> <td>56</td> <td>76</td> </tr> </tbody> </table>	순위	2013년	2014년	2015년	매출액	367	505	629	영업이익	43	56	76
순위	2013년	2014년			2015년											
매출액	367	505			629											
영업이익	43	56			76											
2	플스크린	2,440만														
3	베보	2,180만														
4	머시니마	1,450만														
5	메이커젠	1,230만														
6	스타일홈	1,140만														
7	콜렉티브디지털	910만	CJ E&M	· 2013년 7월 CJ E&M이 처음 MCN 개념을 도입 · '다이아 TV'를 출시하면서 MCN 시장에 뛰어 현재 650명의 크리에이터를 보유하고 있으며, 구독자는 3,600만 명에 월평균 8억 뷰를 돌파												
31	CJ E&M	148만	트레저헌터	· CJ E&M MCN사업팀장 소속이었던 송재홍 대표가 국내 대표 크리에이터 양명, 악어, 김이브 등과 함께 2015년 1월 트레저헌터 설립 · 최근 트레저헌터에 소속된 크리에이터는 총 150명, 구독자는 약 1,200만 명 확보												
96	트레저헌터	47만														

자료 : 소셜블레이드(2016.1.14.), 최근 언론기사



## ■ 글로벌 MCN 시장 동향

- 해외 미디어그룹의 MCN 투자 활성화
  - 모바일 동영상이 기존 방송 미디어 수요를 빠르게 대체하면서 해외 미디어 기업들은 전통적인 방송, 영화사업 모델에서 탈피해 새로운 콘텐츠 제작 유통 플랫폼으로 MCN을 적극 활용하는 전략을 추진

〈표 1〉 X-X 글로벌 MCN의 사업 현황

MCN 기업명	주요 내용
Maker Studios	·2014년 5월 Disney 5억 불에 인수 ·2015년 1월 Vimeo와 독점 콘텐츠 제휴 체결 ·2015년 3월 Digital 오락 스튜디오 Rebl HQ 설립
Fullscreen	·2014년 9월 AT&T는 2~3억 달러 투자
Machinima	·2014년 2월 Warner Bros는 1,800만 달러 투자 ·2015년 2월 Vimeo와 일부 콘텐츠 독점 제휴 계약 체결 ·2015년 3월 신규 OTT기업인 Vessel에 콘텐츠 공급계약
Awesomeness TV	·2013년 5월 드림웍스가 33백만불에 인수 ·2013년 7월 미국 유료채널을 통해 방영 ·2014년 12월 Hearst가 지분 25%를 인수 ·2015년 3월 Film360과 제휴하여 자체 콘텐츠 제작
StyleHaul	·2014년 11월 RTL Group은 1억 7백만 달러 투자 ·2014년 6월 글로벌 메이크업 브랜드인 Maybelline과 공동 마케팅 ·2014년 8월 의류업체 PacSun과 공동 캠페인 진행 ·2015년 12월 Verizon의 OTT서비스인 'Go90'에 콘텐츠 제작/제공 계약 체결
Vevo	·2009년 2월 뮤직 비디오 전문 MCN으로 설립 ·2014년 9월 도요타의 지원으로 'Triple V'을 발표 ·2014년 11월 아티스트 개발사 Made Music과 제휴
3 Minute	·2015년 9월 패션, 뷰티, 피트니스, 요리 등 여성의 라이프스타일에 특화된 MCN

자료: KT Digieco(2015.12), 스마트TV이슈리포트(2015.12), 최근 언론보도 참조

- 기존 온라인 플랫폼과 통신사들도 MCN간의 협력 활성화

〈표 2〉 X-X 글로벌 MCN의 사업 현황

온라인 플랫폼	주요 내용
YouTube	·2015년 9월 인도 뭄바이에 세계에서 7번째 MCN 창작자 전용 YouTube Space를 개설하고 인도의 영화 교육기관인 Whistling Woods International과 협력해서 오디오 및 동영상 편집 장비 등을 제공
Netflix	·2015년 8월~9월 : Anthony Hecox와 Ian Padilla 등 유튜브 스타들을 중심으로 'Smosh : the Move', 'Bad Night' 등을 영상물을 제작해서 전세계에 제공
Amazon	·2015년 6월 MCN의 자체 앱개발과 제작을 지원하는 동영상 기술기업 Beachfront Media와 제휴 ·Amazon은 화장 등 Beauty 관련 유튜브 팬 780만 명을 보유한 Michelle Phan과 협력해서 MCN 앱인 'Icon'을 출시
Verizon	·2015년 10월 밀레니엄 세대를 겨냥한 모바일 OTT서비스 'Go90'을 런칭 ·2015년 10월 Maker Studio와 '오리지널 시리즈' 제작 계약 체결하고, Style Haul, Machinima, Fullscreen, Defy, Tastemade 등 인기 있는 MCN들과 파트너 계약 체결을 추진하며 'Go90'의 시장 지배력을 제고하는 중

자료: 최근 언론보도 참조



## ■ 최근 MCN은 교육, 유통과의 결합 등 다양한 수익모델을 구체화

- (자체 브랜드) 자체 브랜드를 내걸고 쇼핑몰을 런칭한 대표적인 사례로는 트레저헌터이며, 양평, 악어, 김이브, 잉여맨, 최고기 등 소속 크리에이터들의 개성을 담은 상품을 파는 쇼핑몰 크리마켓을 공개
- (교육, 컨퍼런스) 영상 창작자 교육 아카데미나 컨퍼런스 사업을 진행하는 경우다. 레페리는 유튜브와 함께 1인 뷰티 크리에이터에게 촬영과 편집, 기획, 디자인 강의와 일대일 멘토링, 영상제작 실습을 3주 코스로 단기 집중 제공하는 '뷰티 크리에이터 랩' 프로그램을 공동운영 중
- (홍소핑) 최근 아프리카TV는 모바일이나 PC 웹에서 간편하게 시청할 수 있는 영상 콘텐츠로 소비자 접점을 확대하기 위해 현대홍소핑과 협업하거나, 인터파크투어와 실시간 여행방송을 선보였고, FT아일랜드 멤버 이홍기가 진행하는 '홍소핑'을 진행하는 등 빠른 속도로 유통 및 소핑업계와의 협력을 확대하고 있음

- 다양한 수익모델이 등장하고, 커머스와 결합하며, 중국 등 글로벌 시장 진출을 추진하며 진화
- 다양한 분야의 창의적 크리에이터가 다양한 MCN 사용자층을 대상으로 차별화된 서비스를 제공하고, 새로운 플랫폼을 형성하는 것이 글로벌 경쟁력 확보에 핵심





## 스마트 농업 현황

### ■ 전통 농업에 SW, IoT 기술 등을 접목한 스마트농업 확산

- 스마트 농업 적용으로 생산성·품질 향상, 노동력 절감, 유통구조 혁신 구현
- 국내 상황을 반영한 맞춤형 적용 필요

### ■ 시장 및 동향

- (세계 시장 규모) 세계 농업관련 융복합 시장규모 꾸준히 증가
    - 식량 시장(약 5조 8천 억\$, '14), BT 융합 시장(144억\$, '12→248억\$, '17)
    - ※ 자동차(1조 6천 억\$), 전자·정보통신(2조 9천 억\$), 철강(1조 4백 억\$)
  - (국내 현황) 쌀 위주 생산, 곡물자급률은 낮고 소규모 자작농 위주
    - 국내 곡물자급률은 24%, OECD 주요국 34개국 중 32위
    - ※ 자급률(%): 쌀(95.7), 보리(24.8), 밀(0.7), 옥수수(0.8), 콩(11.3) ('14년)
    - 농가인구의 감소 및 고령화, 소규모 자작농 위주
    - ※ 농가 인구의 41.8%가 60세 이상 고연령층, 경지면적이 0.6ha 대부분 소규모 자작농
  - (농업 환경의 변화) ICT, BT와 접목해 혁신이 가속화 되고 있음
    - 미국, 네덜란드, 일본 등 ICT 융복합 농업으로 경쟁력 강화
    - ※ 미국(농경관리 SW), 네덜란드(시설원예), 일본(식물공장)
    - 작물 생산비용 절감, 환경오염 최소화, 최적의 생산 환경 제공, 농산물 유통 효율화를 추구
- 세계 스마트 농업 기술 투자 현황은 (23억 4000만\$, '14)에서 (22억 6000만\$, '15 상반기)로 2015년 전체적으로 40억\$ 이상의 투자가 이뤄진 것으로 추정 (AgTech Investment Report, 2015)

김태호 선임연구원

teokim@spri.kr

〈표 1〉 농업 투자 주요 분야

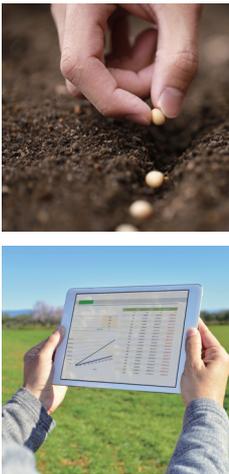
- 농업의 새로운 온·오프라인 유통
- 바이오 재료·화학·지속가능한 단백질
- 실내농업(식물공장 포함)
- 농업부산물 폐기처리
- 정밀농업을 통한 의사결정 지원시스템 및 스마트 농업
- 스마트 장비와 하드웨어
- 농업용 드론과 로보틱스
- 토양과 종자
- 바이오 에너지

자료 : 글로벌 자금, 다음 승부처는 스마트 농업, 푸드테크, 김판건 (2016.01)

- (국내 융복합 농업 확산 현황) 시설원예의 일부 품목 확산 가능한 수준, 축산 분야는 도입 단계, 유통·소비분야(이력추적) 등에서도 일부 성과

〈그림 1〉 ICT 융복합 모델 개발 현황

■ 추진중 ■ 결과물



분야/단계	R&D		실증단계			확산 단계		
생산	<div style="text-align: center;"> <span style="color: green;">→</span> 시설원예 복잡환경제어 <span style="color: green;">→</span> </div>							
	시설원예	최적 생육 모델	생육환경 모니터링	오이·깻잎	버섯	딸기·수박·참외	파프리카·토마토·장미·국화	
		축산	질병관리	낙농정보 통합관리		양돈		
			<div style="text-align: center;"> <span style="color: green;">→</span> 지능형 축사관리 <span style="color: green;">→</span> </div>					
유통	<div style="text-align: center;"> <span style="color: green;">→</span> 전자직거래 <span style="color: green;">→</span> </div>							
		가상스토어		Pos-Mall	ERP	로컬푸드		
소비	<div style="text-align: center;"> <span style="color: green;">→</span> 소비안전 정보 <span style="color: green;">→</span> </div>							
		품질모니터링		학교급식		이력추적		
농촌	<div style="text-align: center;"> <span style="color: green;">→</span> 농촌관광/체험 <span style="color: green;">→</span> </div>							
		농촌 관광정보						

※ R&D단계에서 현장수요 원천·응용기술 개발, 실증단계에서 개발기술을 검증, 성과모델 발굴 후 보급 확산

※ 자료 : 농식품산업의 미래성장산업화를 위한 농식품 ICT 융복합 확산대책(안), 농림축산식품부 정보화 담당실(2013.08)

● (그간의 정책) ICT 융복합 농업 정책 추진 현황

- 농촌지역에 PC 및 네트워크 기반 조성을 위한 1차 농업·농촌정보화 기본계획('02)을 시작으로, 현재 3차 정보화 기본계획 추진 중

〈표 2〉 국내 정보화 기본계획 추진성과

단계 (기간)	1단계 (2002 - 2006)	2단계 (2007 - 2011)	3단계 (2012 - 2016)
추진 방향	·농촌지역 정보 활용 기반조성	·농업·농촌의 u-Life 현실화로 농촌 삶 및 복지 향상	·농식품 ICT 활성화를 통한 모델 구축 및 확산
	·효율적 자원관리 및 농산업 활성화를 위한 디지털 기반 조성	·지능기반의 농촌 정보화 인프라 확충	·농어촌 지역 디지털 기반 조성
		·농식품분야에 IT 기술이 융합된 u-Farm 사업모델 도입·검증	·행정정보화를 통한 미래지향적 농정 실현
주요 성과	·농업기술정보 활용	·농어촌 지역 광대역통신망 구축정보	·스마트농업 추진 전략 마련
		·정보자원의 통합·연계	

출처 : 농림·식품 분야의 新ICT융합전략, 한국정보화진흥원, 재구성 (2014.06)

## ■ 시사점

- (ICT 융복합 농업 토대 마련) 다양한 ICT 기술이 농업 분야에 접목될 수 있도록 산업 생태계 조성 필요
  - (데이터 활용 기반 마련) 생산·유통·소비에서 발생하는 데이터의 축적 및 활용을 위한 통합정보 활용 체계 구축
- (농업기술벤처 육성) 선진 외국 기술의 단순 적용을 지양하고, 소농, 고령화 등 국내 실정에 맞는 스마트 농업 기술 개발 필요
  - 스마트 그린하우스, 지능형 축사, 스마트 팜 등 국내 실정에 맞는 농업기술벤처 육성
- (유통 구조 혁신) 생산자 - 소비자의 거리를 좁혀 중간유통의 비효율성 개선
  - 온라인과 모바일을 통한 농식품 유통 비즈니스 플랫폼 구축 등으로 농산물 거래·판로 다각화
- (현장 연계 강화) 농가 대상 교육 등 현장 밀착형 사업체계 마련
  - (현장 수요 반영) 추진 사업의 실효성 증진을 위해 사업 평가·검증·환류 절차를 보완하여 현장 연계성 강화



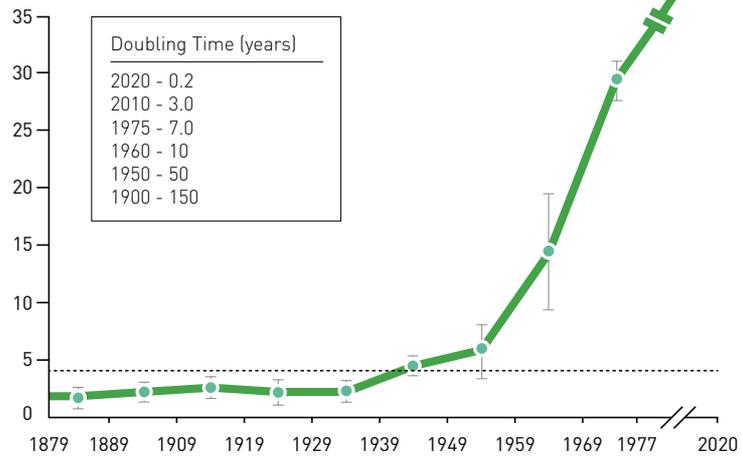
## 인공지능 헬스케어 시장의 미래

- 인공지능 헬스케어 시장은 2021년까지 연평균 40% 이상의 고속 성장 기대, 최근 인공지능 기반의 스타트업에 많은 투자가 이루어지고 있음
- 의료 빅데이터를 활용한 개인 맞춤형 진단 및 예방으로 의료의 질을 향상시키기 위한 다양한 비즈니스 모델에 대한 고민과 투자가 필요한 시점

### ■ 인공지능 헬스케어 시장의 고속 성장 예상

- 인공지능 헬스케어 시장, 연평균 40% 이상의 거대 산업으로 성장 예측
  - 헬스케어에서의 인공지능과 인지 컴퓨팅 시장은 2014년 10조 \$633.8M(약 7000억 원)에서 연평균 40% 이상 성장하여 2021년에는 약 \$6662.2B(약 7744조 원)규모의 거대한 시장을 형성할 것으로 전망(Frost & Sullivan, 2015.12)
- 의료분야의 미래 핵심 요소는 데이터
  - 의료 데이터는 2020년까지 73일마다 두 배씩 늘어날 것으로 기대됨<sup>(1)</sup>

〈그림 1〉 의료 Data 증가 속도



자료: University of IOWA Carver College of Medicine

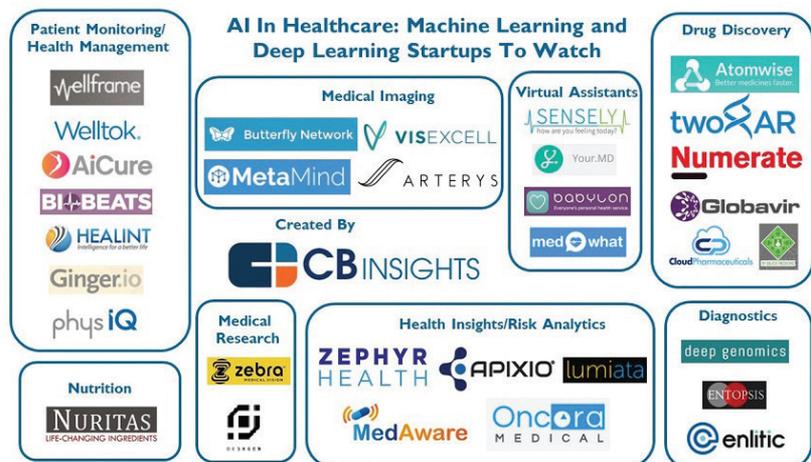
- 과거와 비교할 수 없을 만큼 폭발적으로 늘어나는 의료데이터를 어떻게 분석을 하고 이로부터 Insight를 얻어 문제를 해결할 것인가가 초미의 관심사

## ■ 미국의 헬스케어분야의 AI 스타트업 투자 현황

### ● 해외 투자자, AI 기반의 헬스케어 서비스/제품 기업에 투자 중

- CB Insight에서 제공하는 2016년 2월 기준 미국의 AI 관련 Healthcare Startup 투자를 살펴보면, 32개의 스타트업 기업에 총 \$534.58M(약 5800억 원)을 투자
- 투자 분야는 8가지로 분류할 수 있으며, 그 중 '환자 모니터링과 건강관리' 기업에 가장 많은 금액을 투자함

〈그림 2〉 미국 내 32개 AI관련 헬스케어 스타트업 투자 그룹



자료: CBInsight (2016.02), <https://www.cbinsights.com/blog/artificial-intelligence-startups-healthcare/>

〈표 1〉 각 분야별 투자 금액 (단위: 백만 달러)

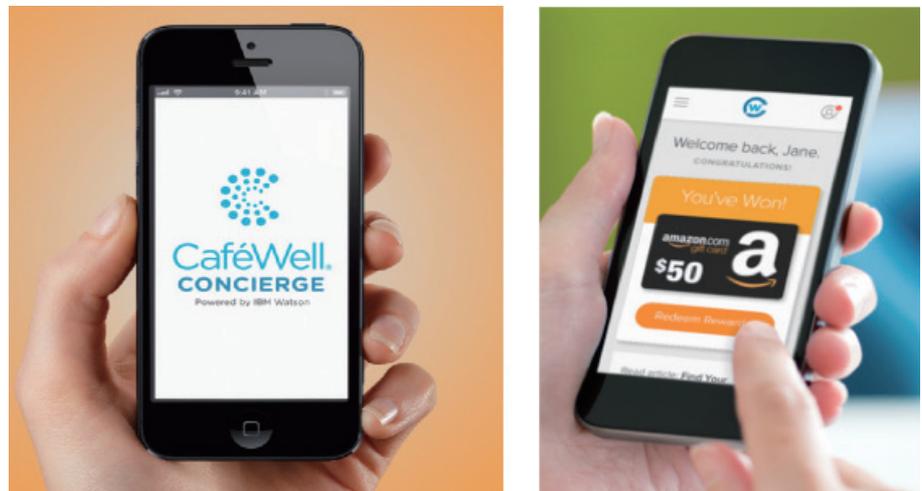
구분	금액	투자받은 회사 수	대표 회사사
환자모니터링/건강관리	252.66	7	WellTok, Ginger.io 등
의료이미지분석	115.25	4	Butterfly Net Inc., Arterys 등
의료 Insights/위험 분석	69.77	5	Zephyr Health, Apixio 등
가상 Assistant	35.42	4	Babylon Health
신약개발	29.7	6	Numerate
진단	17.43	3	Enlitic
Research	11.25	2	Zebra Medical Vision
영양(식품 관련)	3.2	1	Nuritas

자료: CBInsight 자료 재구성

● 32개 중 가장 큰 금액을 투자받은 곳은 건강관리 플랫폼회사인 'Welltok'

- Emergence Capital Partners, Flare Capital Partners를 비롯한 10개의 투자자로부터 총 \$191.9M의 자금을 유치 받은 환자 모니터링 및 건강관리 회사로 3년 성장률은 1,800%을 기록<sup>(2)</sup>
- IBM왓슨과 구동하여 모바일을 통한 지속적인 대화를 통해 건강을 최적화하는 지능성 개인 가이드(CafeWell Concierge)를 제공하고 결과 목표 성취 시 보상하는 시스템(CafeWell Rewards) 구축

〈그림 3〉 (좌) 웰톡 CafeWell Concierge, (우) CafeWell Rewards



자료: Welltok 홈페이지

(2) <http://withpress.co.kr/archives/1168>

## ■ 헬스케어에서의 인공지능 적용 분야

FROST & SULLIVAN의 보고서에 의하면, 헬스케어 분야에서 인공지능을 적용한 분야를 아래와 같이 8가지로 구분함

〈표 2〉 인공지능 적용 테마

기업	세부내용
	다양한 AI 기술 회사와 병원 관련 공급자 간 파트너십을 통해 중요한 치료 정보를 제공하고 병원 내 정보의 Workflow 문제 해결
	사진, SNS정보, 리뷰, 모바일 데이터 등의 여러 가지 형태의 비정형 데이터로부터 Insight를 얻음
	패턴인식과 머신러닝 AI 플랫폼을 통한 의료 이미지 분석으로 정확한 진단 지원
	개선된 안면인식과 모션센싱 SW를 활용하여 약물에 대한 환자의 순응도 관찰하는 등 자동화된 환자 가이드 제공
	AI 기반 솔루션으로 원활한 환자 흐름과 간호인력 유지 등의 병원 내 운영관련 Workflow를 효율화
	헬스케어 분야의 AI 시장 확대 효과 - 2021년까지 매년 40% 가량 성장률 기대
	데이터기반의 치료를 통해 일류 수준의 진료를 어디에서나 받을 수 있는 헬스케어 민주화 시대의 선도적 역할
	웨어러블 기기를 통한 실시간 데이터를 의료 서비스와 연결한 건강 관리 체계 구축

## ■ 시사점

- 국내 인공지능 관련 헬스케어 활성화를 위한 인식 제고 및 투자가 필요
  - 폭발적으로 늘어나는 의료 데이터 속에서 인공지능 기반의 헬스케어 기술은 이 데이터들을 분석하여 개인의 삶의 질을 향상 시킬 수 있는 돌파구
  - 다양한 AI기술 업체와 병원/제약/보험 관련 공급자간 교류를 통한 기술 발전 및 다양한 비즈니스 모델 창출 필요
  - 이를 위해 장기적 관점의 정책적 지원과 관련 Startup 기업에 대한 가치 투자가 보다 적극적으로 필요한 시점

# 소프트웨어 생산 현황

## 국내 소프트웨어 생산 현황

출처/시기 : KAIT · SPRi / 2016. 3

### 통계 내용

(표 1) 월별 소프트웨어 생산 동향(단위 : 십억 원, 전년 동월/동기 대비 증감률)

구분	2014년	2015년							
		1분기	2분기	3분기	10월	11월	12월	12월 누적	
패키지 SW	생산액	6,602	1,490	1,640	1,635	560	574	874	6,773
	증감률	1.9%	2.0%	2.9%	2.9%	4.3%	1.0%	2.3%	2.6%
IT 서비스	생산액	29,759	7,024	7,439	7,396	2,753	2,686	4,125	31,423
	증감률	2.5%	11.6%	6.9%	3.1%	2.0%	5.7%	0.8%	5.6%
게임	매출액	9,919	2,460	2,527	-	-	-	-	-
	증감률	-7.0%	2.2%	8.5%	-	-	-	-	-
소계 (게임 제외)	생산액	36,360	8,514	9,079	9,031	3,313	3,260	4,999	38,196
	증감률	2.4%	9.8%	6.2%	3.1%	2.4%	4.8%	1.1%	5.0%

※ 게임은 생산액이 아닌 매출액을 기준으로 집계되고 있으며, 분기별 매출실적 집계(KOCCA, 2분기 콘텐츠 산업 동향분석 보고서 2015. 9. 30)

### 분석

- 패키지소프트웨어 : 12월 패키지소프트웨어는 전년 동월 대비 2.3% 증가한 8,741억 원으로 집계, 12월 누적 패키지 소프트웨어 생산액은 전년 동기 대비 2.6% 증가한 6조 7,726억 원으로 나타남
  - 12월 패키지소프트웨어 생산액 중 시스템 소프트웨어 생산액은 전년 동월 대비 6.1% 증가, 응용 소프트웨어는 1.2% 감소
- IT 서비스 : 12월 IT서비스는 전년 동월 대비 0.8% 증가한 4조 1,252억 원이며, 12월 누적 생산액은 31조 4,235억 원으로 전년 동기 대비 5.6% 증가
  - 12월 IT서비스 생산액 중 IT컨설팅 및 시스템 통합 생산액이 전년 동월 대비 4.3% 감소, IT시스템 관리 및 지원서비스 부문이 5.1%, 기타 IT서비스가 40.1%로 증가하며 시장 성장을 견인

# 소프트웨어 수출 현황

## 국내 소프트웨어 수출 현황

출처/시기 : SPRi · KAIT/ 2016. 3

〈표 2〉 월별 소프트웨어 수출동향(단위 : 백만 달러, 전년 동월/동기 대비 증감률)

구분	2014년	2015년							
		1분기	2분기	3분기	10월	11월	12월	12월 누적	
패키지 SW	수출액	2,640	690	671	663	240	249	252	2,765
	증감률	25.4%	32.4%	-5.1%	10.2%	-8.6%	-8.5%	-7.9%	4.8%
IT 서비스	수출액	2,688	729	889	742	238	317	356	3,271
	증감률	28.8%	14.2%	48.2%	12.9%	11.1%	27.4%	8.1%	21.7%
게임	수출액	2,939	713	727	-	-	-	-	-
	증감률	8.2%	8.9%	4.6%	-	-	-	-	-
소계 (게임 제외)	수출액	5,327	1,419	1,560	1,405	478	566	608	6,036
	증감률	27.1%	22.3%	19.3%	11.6%	0.3%	8.7%	0.8%	13.3%

※ 패키지소프트웨어, IT서비스 수출액은 매월 입금된 금액을 기준으로 집계

※ 게임 산업은 분기별로 수출 실적을 집계하고 있으며, 수출액은 분기별 평균 환율을 적용하여 산출(KOCCA, 2분기 콘텐츠산업 동향분석 보고서 2015. 9. 30)

### 통계 내용

### 분석

- 패키지소프트웨어 : 12월 패키지소프트웨어 수출액은 전년 동월 대비 7.9% 감소한 2억 5,192만 달러, 12월 누적 수출액은 전년 동기 대비 11.6% 증가한 26억 3,954만 달러로 나타남
  - 12월 시스템소프트웨어 수출액(4,629만 달러)은 전년 동월 대비 122.3% 증가하였으며, 응용 소프트웨어 수출액이 2억 563만 달러로 18.6% 감소하였으나 패키지소프트웨어 전체 시스템 소프트웨어 시장 수출액이 반등함
- IT 서비스 : 12월 IT서비스 수출액은 3억 5,592만 달러로 전년 동월 대비 8.1% 증가하였으며, 누적 수출액도 전년 동기 대비 21.7% 증가한 32억 7,083만 달러 규모로 집계
  - 12월 IT컨설팅 및 시스템 통합 분야의 수출액(2억 4,187만 달러)은 전년 동월 대비 0.2% 증가하였고, IT시스템 관리 및 지원 서비스 분야의 수출액(1억 1,122만 달러)도 27.7% 증가하면서 전체 IT서비스 수출액이 상승세를 이어감



# 2016년 SW기업 수출, 고용 및 R&D 투자 전망 조사

## Executive Summary

본 연구소에서 수행한 2016년 SW기업 전망 조사 결과에 따르면 국내외의 어려운 경제여건 속에서도 SW산업은 수출(8.0%), 고용(2.0%) 및 연구개발투자(11.3%)에 있어서 전년대비 플러스 성장을 달성할 것으로 전망된다. 수출경기실사지수는 129.20이며, 업종별로는 게임SW와 인터넷서비스의 수출 증가율이 높을 것으로 예상된다. 주요 수출지역은 중국과 동남아시아로 조사되었다. 하지만 아직 수출 계획이 없는 업체가 62.7%에 달해 수출 성장의 기회가 많은 상황이다. 2016년 SW기업의 전체 신규채용 규모는 5만 9천여 명에 이를 것으로 추정되며, 직무별로는 SW기술인력의 비중이 45.7%로 집계되었다. SW기업들의 평균 고용실사지수는 130.1에 달했으며, 특히 게임SW 기업들이 고용시장을 이끌 것으로 보인다. 마지막으로 연구개발지수는 135.7으로 SW기업들의 R&D활동이 크게 개선될 것으로 보여 글로벌 경쟁력 강화에 도움이 될 전망이다. 업종별로는 IT서비스와 임베디드 SW 기업들의 연구개발 투자 증가가 두드러졌으며, 신사업 추진계획이 있는 기업들은 전체 조사업체의 42.2%에 달했다.

이동현 선임연구원

dlee@spri.kr



## 조사 개요

### (1) 조사 목적

- 경기동향의 판단지표 제공
  - 소프트웨어(SW)산업의 현재와 미래의 경기 동향과 관련 있는 수출, 인력채용, R&D투자 등의 전망에 관한 객관적인 판단지표를 제공하여 기업의 합리적인 경영계획과 정부의 정책수립의 기초자료로 활용

### (2) 조사 내용

- 일반, 수출, 인력, R&D 등 총 4개 분야에 대해서 조사 실시

〈표 1〉 조사 내용

구분	조사 내용
일반 현황	·업종 ·2015년 총 매출액
수출 현황 및 전망	·2015년 총 수출액 ·2016년 수출액 증감 전망 ·주요 수출 지역
인력 현황 및 전망	·총 인력 현황 ·SW 기술 인력 현황 ·2016년 인력 증감 전망 ·2016년 인력 채용 계획(전체 및 직무별)
R&D 현황 및 전망	·2015년 R&D 투자액 ·2016년 R&D 투자 증감 전망 ·주요 R&D 투자 분야 ·신사업 계획

### (3) 조사 기간 및 방식

- 설문항목 개발 및 예비조사 : 2016년 2월 15일 ~ 2016년 2월 19일
- 실시기간 : 2016년 2월 22일 ~ 2016년 2월 26일
- 조사방식 : 2015년 SPRi에서 진행한 '국내 소프트웨어 기업 현황 조사<sup>(1)</sup>'의 모집단에 속한 기업체를 대상으로 온라인 설문조사 실시 후, 전화로 보완조사 병행
- 유효응답수 : 322개

(1) 소프트웨어정책연구소, 국내 소프트웨어 기업 경쟁력에 관한 연구

## 응답기업 현황

### (1) 업종

- 패키지 SW - IT 서비스 - 인터넷 서비스 - 임베디드 SW - 게임 SW 순

- 패키지 SW 기업은 35.4%인 114개 기업, 그 다음으로 IT서비스 기업은 29.8%인 96개, 인터넷 서비스 기업이 13.7%인 44개 기업, 임베디드 기업이 10.9%인 35개 기업, 게임 SW 기업은 10.2%인 33개 기업 순으로 집계됨

〈그림 1〉 응답기업 업종 분포



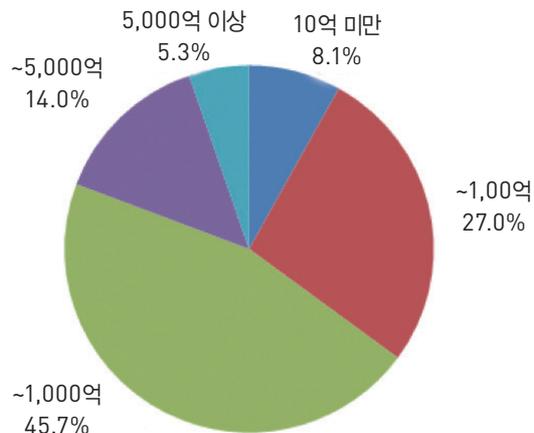
### (2) 매출액 규모

- 유효응답기업의 평균매출액은 1,308억 원

- 업종별 평균 매출액은 임베디드 업종이 2,408억 원으로 가장 높은 수준  
- 인터넷서비스 2,251억 원, IT서비스 1,765억 원, 게임 SW 1,342억 원, 패키지 SW 213억 원 순

- 100억 원 이상 1,000억 원 미만 기업의 비중이 45.7%, 147개 기업으로 가장 많은 것으로 나타남  
- 10억 원 이상 100억 원 미만 기업이 27.0%인 87개, 1,000억 원 이상 5,000억 원 미만 기업이 14.0%인 45개, 10억 원 미만 기업이 8.1%인 26개, 5,000억 원 이상 기업은 5.3%인 17개 기업 순으로 집계됨

〈그림 2〉 응답기업 매출액 현황



〈표 2〉 업종 및 매출액 규모별 응답 기업 분포

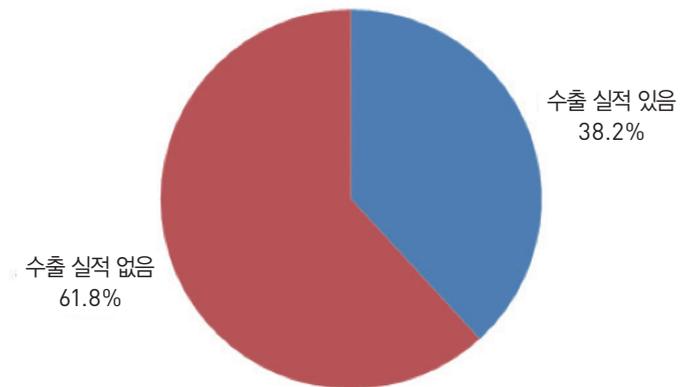
구분	10억 미만	~100억 미만	~1,000억 미만	~5,000억 미만	5,000억 이상	계	평균 매출액 (백만원)	1인당 매출액 (백만원)
게임SW	0	10	14	5	4	33	134,236	491.4
패키지 SW	11	46	53	4	0	114	21,321	227.3
IT서비스	10	22	43	16	5	96	176,507	392.5
인터넷 서비스	2	5	24	8	5	44	225,056	544.6
임베디드	3	4	13	12	3	35	240,789	636.8
계	26	87	147	45	17	322	130,854	446.8

- 1인당 매출액은 임베디드 업종이 6억 3,680만 원으로 가장 높은 수준  
 - 인터넷서비스 5억 4,460만 원, 게임 SW 4억 9,140만 원, IT서비스 3억 9,250만 원, 패키지 SW 2억 2,730만 원 순으로 나타남

(3) 수출액 규모 및 주요 수출 지역

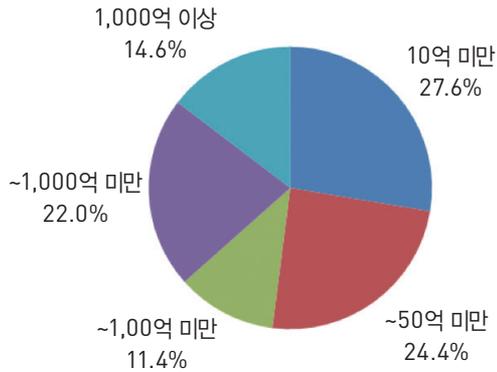
- 수출 실적이 있는 기업은 38.2%인 123개 기업

〈그림 3〉 2015년 수출 실적 현황



- 수출액 10억 원 미만 기업 비중의 27.6%로 가장 높게 나타났으며, 수출액 50억 원 미만 기업까지 합한 비중은 52.0%

〈그림 4〉 수출액 규모별 기업 분포



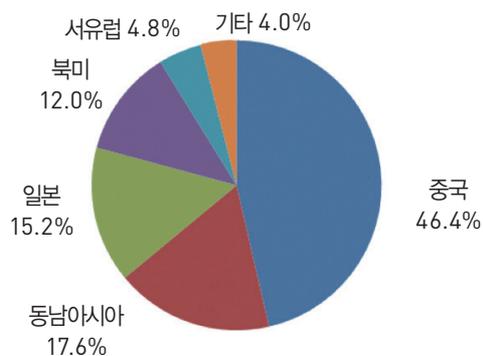
- 기업 당 평균 수출액은 894억 원
  - 업종별로는 IT서비스 업종이 2,372억 원으로 가장 많았으며, 임베디드 1,684억 원, 인터넷서비스 832억 원, 게임 SW 526억 원, 패키지 SW 40억 원 순

〈표 3〉 업종 및 수출액 규모별 응답 기업 분포 및 평균수출액

구분	10억 미만	~50억 미만	~1,000억 미만	~1,000억 미만	1,000억 이상	계	평균 매출액 (백만원)
게임SW	2	7	6	4	6	25	52,594
패키지SW	20	10	6	6	0	42	4,016
IT서비스	7	6	2	3	3	21	237,234
인터넷 서비스	3	7	0	4	2	16	83,199
임베디드	2	0	0	10	7	19	168,371
계	34	30	14	27	18	123	89,396

- 주요 수출지역은 중국이 46.4% (58개) 기업으로 가장 많은 비중을 차지 (단일 응답 기준)
  - 동남아시아 17.6%(22개) 기업, 일본 15.2%(19개) 기업, 북미 12.0%(15개) 기업, 서유럽 4.8% (6개) 기업 순으로 나타남

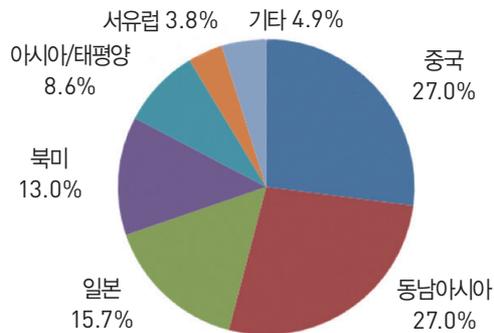
〈그림 5〉 주요 수출 지역 현황(단일응답기준)





- 1, 2, 3 순위의 주요수출지역을 모두 합한 기준으로는 27%(100개) 기업이 중국과 동남아시아를 선택 (복수 응답 기준)
  - 일본 15.7%(58개) 기업, 북미 13.0%(48개) 기업, 기타 아시아/태평양 8.6%(32개) 기업, 서유럽 3.8%(14개) 기업 순으로 나타남

〈그림 6〉 주요 수출 지역 현황 (복수응답기준)

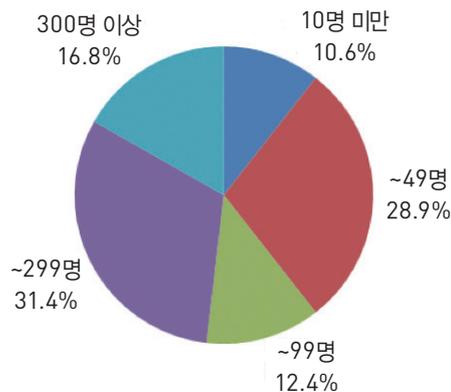


- 아시아권 국가에 대한 수출비중이 70% 후반대로 이 지역에 대한 수출의존도가 매우 높음

(4) 종업원 수

- 유효응답기업의 평균 종업원수는 293명
  - 100명 이상 300명 미만 기업의 비중이 31.4% 101개로 가장 높음
  - 10명 이상 50명 미만 기업 / 300명 이상 기업 / 50명 이상 100명 미만 기업 / 10명 미만 기업 순

〈그림 7〉 응답 기업 종업원 수 규모 분포



- 업종별 평균 종업원 수는 IT서비스 업종이 449.7명으로 가장 많은 것으로 나타남
  - 인터넷서비스 업종 413.3명, 임베디드 업종 378.1명, 게임 SW 업종 273.2명, 패키지 SW 업종 93.8명으로 나타남
  - 패키지 SW와 IT서비스 업종은 10명 이상 50명 미만 기업의 비중이 상대적으로 높게 나타나고, 나머지 업종은 100명 이상 300명 미만 기업의 비중이 상대적으로 높게 나타남

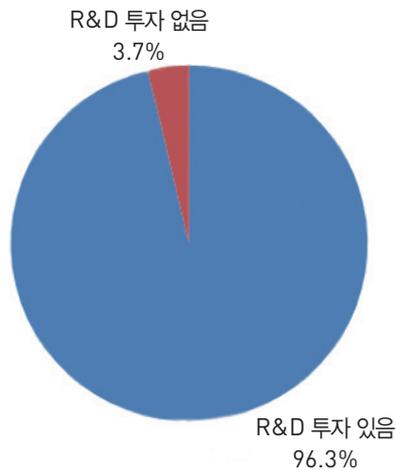
〈표 4〉 업종 및 종업원 규모별 응답 기업 분포

구분	10억 미만	10명~ 49명	50명~ 99명	100명~ 299명	300명 이상	계	평균 종업원수
게임SW	1	7	4	12	9	33	273.2
패키지SW	18	40	17	32	7	114	93.8
IT서비스	11	30	11	28	16	96	449.7
인터넷 서비스	2	8	6	15	13	44	413.3
임베디드	2	8	2	14	9	35	378.1
계	34	93	40	101	54	322	292.8

(5) R&D 투자 규모

- 2015년 기준 96.3%의 기업이 R&D에 투자

〈그림 8〉 R&D 투자 현황



- 기업 당 평균 R&D 투자 금액은 103억 원
  - R&D 투자 금액이 10억 원 미만인 기업이 46.1%
  - 20억 원 이상 50억 원 미만 기업이 17.7%, 10억 원 이상 20억 원 미만 기업이 16.1%, 100억 원 이상 기업이 10.6%, 50억 원 이상 100억 원 미만 기업이 9.4% 순

〈그림 9〉 R&D 투자 규모별 분포



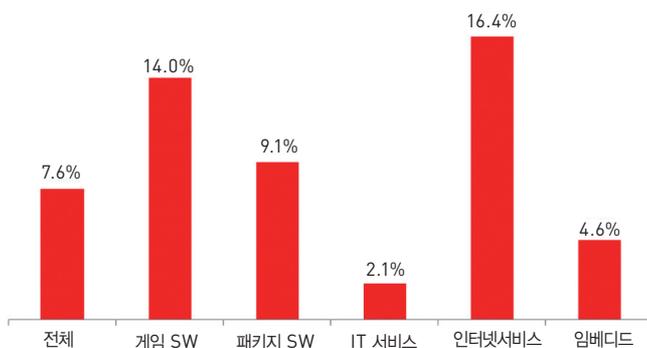
- 업종별로는 인터넷서비스 업종의 평균 R&D 투자 금액이 368억 원으로 가장 높고, 그 다음으로 게임 SW 업종 188억 원, 임베디드 업종 113억 원, IT서비스 업종 38억 원, 패키지 SW 업종 21억 원 순
  - 게임 SW와 임베디드 업종에서 R&D 투자 금액 100억 원 이상 기업들의 비중이 상대적으로 높음
  - 패키지 SW, IT서비스, 인터넷서비스 업종에서 R&D 투자액 10억 원 미만 기업들의 비중이 높게 나타남

〈표 5〉 업종 및 R&D 투자 규모별 응답 기업 분포

구분	10억 미만	~20억 미만	~50억 미만	~100억 미만	100억 이상	계	평균 R&D 투자
게임SW	8	4	4	6	11	33	18,757
패키지SW	54	18	23	7	3	105	2,103
IT서비스	57	15	15	3	4	94	3,806
인터넷 서비스	18	8	6	6	6	44	36,818
임베디드	6	5	7	7	9	34	11,316
계	143	50	55	29	33	310	10,330

- 매출액 대비 R&D 투자 금액 비중은 평균 7.6% 수준
  - 업종별로 살펴보면 인터넷서비스가 16.4%로 가장 높은 수준
  - 게임SW 14.0%, 패키지SW 9.1%, 임베디드SW 4.6%, IT서비스 2.1% 순

〈그림 10〉 업종별 매출액 대비 R&D 투자 금액 비중

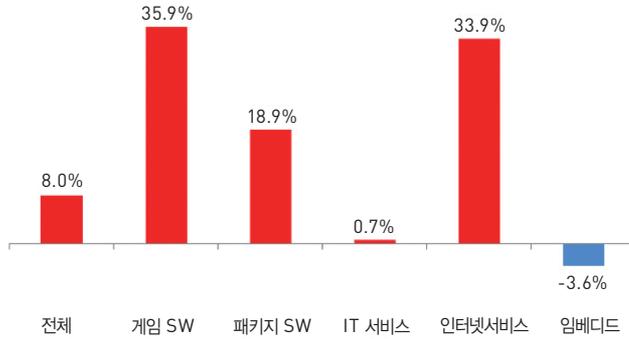


## 2016년도 전망조사 결과

### (1) 수출 전망

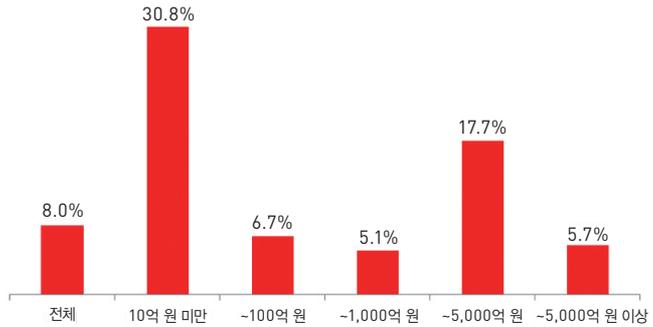
- 2016년 수출액은 전년대비 8.0% 증가할 전망
  - 업종별로는 게임SW와 인터넷서비스 업종이 각각 35.9%, 33.9% 증가할 것으로 예상

〈그림 11〉 업종별 2016년 수출액 증감률



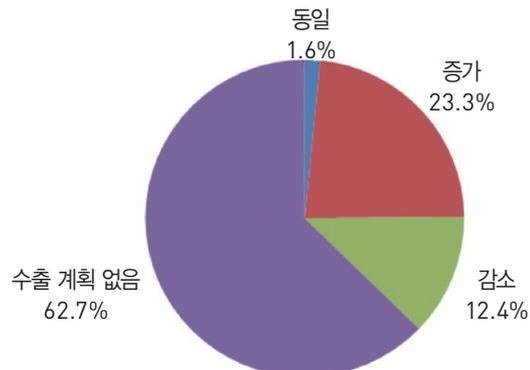
- 매출액 규모별로는 10억 원 미만의 소기업들의 수출증가 폭이 30.8%으로 가장 클 것으로 예상됨

〈그림 12〉 매출액 규모별 2016년 수출액 증감률



- 2016년 수출 증가를 예상하는 기업은 24.2%인 78개 기업
  - 감소를 예상하는 기업은 13.7%인 44개 기업이며, 동일한 수준을 예상하는 기업이 1.6%, 나머지 62.7%는 수출 계획이 없었음

〈그림 13〉 2016년 수출 증감 전망



- 업종별로는 게임 SW 업종은 증가가 45.5%, 인터넷서비스 업종은 증가가 34.1%로 상대적으로 높게 나타남
  - 임베디드 SW기업들은 수출 감소 예상 비중이 42.9%
- 전체 수출경기실사지수 평균은 129.2
  - 업종별로는 인터넷서비스 업종이 187.5로 가장 높은 수준
  - IT서비스 업종 142.9, 패키지 SW 업종 141.0, 게임 SW 업종 128.0 순
  - 임베디드 업종은 42.1으로 2016년 수출 전망이 어두운 것으로 나타남

〈표 6〉 업종별 2016년 수출 증감 예상 기업 현황

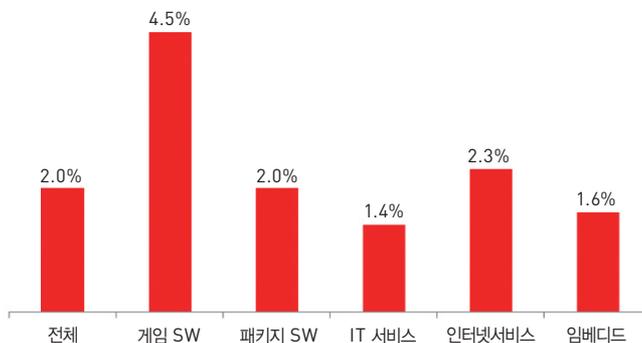
구분	동일	증가	감소	수출 계획 없음	계	수출경기 실사지수
게임SW	2	15	8	8	33	128.0
패키지SW	1	27	11	75	114	141.0
IT서비스	2	14	5	75	96	142.9
인터넷 서비스	0	15	1	28	44	187.5
임베디드	0	4	15	16	35	42.1
계	5	75	40	202	322	129.2

주 : 경기실사지수 100은 보합, 100보다 크면 호전, 100보다 작으면 악화를 의미

## (2) 인력 및 신규고용 전망

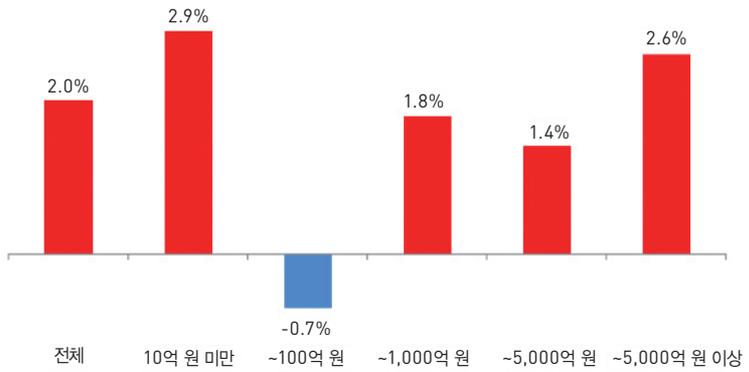
- 2016년 전체 인력은 전년대비 2.0% 증가할 전망
  - 업종별로는 게임 SW 업종의 인력 증가율이 4.5%로 예상되어 다른 업종에 비해 상대적으로 높게 나타났으며, 그 다음으로 인터넷 서비스 업종 2.3% 증가, 패키지 SW 업종 2.0%, 임베디드 업종 1.6% 증가, IT서비스 업종 1.4% 증가할 것으로 예상됨

〈그림 14〉 업종별 2016년 인력 증감률



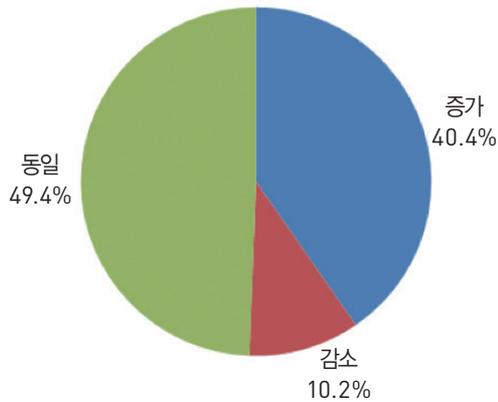
- 매출액 규모별로는 10억 원 미만 소규모 기업 및 5,000억 원 이상의 기업이 각각 2.9%, 2.6% 증가할 것으로 예상되나, 10억 원 이상 100억 원 미만 기업은 0.75% 매출 감소를 전망하고 있었음

〈그림 15〉 매출액 규모별 2016년 인력 증감률



- 40.4%의 기업이 전년도 대비 2016년 전체인력 수를 증가시킬 예정
  - 동일한 수준을 예상하는 기업이 49.4%, 159개로 가장 많은 비중을 차지하였으며, 감소를 예상하는 기업은 10.2%인 33개 기업

〈그림 16〉 2016년 인력 증감 전망



- 전체 고용실사지수 평균은 130.1
  - 업종별로는 게임 SW 업종의 고용실사지수가 169.7로 가장 높아 게임 SW 기업들이 고용시장을 이끌 것으로 예상

〈표 7〉 업종별 인력 증감 예상 및 인력경기실사지수

구분	동일	증가	감소	계	고용실사지수
게임SW	6	25	2	33	169.7
패키지SW	58	46	10	114	131.6
IT서비스	58	28	10	96	118.8
인터넷서비스	26	17	1	44	136.4
임베디드	11	14	10	35	111.4
계	159	130	33	322	130.1

주 : 고용실사지수 100은 보합, 100보다 크면 호전, 100보다 작으면 악화를 의미

- 2016년 신규 채용 예정 인력은 총 3,901명으로 집계됨
  - 신규 채용 규모는 전체종사자수 대비 4.1%에 이를 것으로 전망됨
  - 업종별로 살펴보면 IT서비스 업종에서 39.1%인 1,526명의 채용 계획이 있는 것으로 조사됨
  - 인터넷서비스 업종 22.1% 863명, 게임 SW 업종 15.9%인 622명, 패키지 SW 업종 14.1%인 550명, 임베디드 업종 8.7%인 340명 순으로 나타남
  
- 기업 1개당 평균 채용 인력은 14.0명
  - 업종별로는 인터넷서비스 업종이 평균 21.0명, 게임 SW 업종이 평균 20.7명, IT서비스 업종이 평균 17.5명, 임베디드 업종이 평균 11.0명, 패키지 SW 업종이 평균 6.1명인 것으로 집계됨

〈표 8〉 업종별 2016년 평균 및 전체 신규 채용 예정 규모

구분	채용 예정 인력 합계	평균 신규 채용 규모
게임SW	622	20.7
패키지SW	550	6.1
IT서비스	1,526	17.5
인터넷서비스	863	21.0
임베디드	340	11.0
계	3,901	14.0

- 직무별로 신규채용 분야를 구분해보면 SW 기술인력이 45.7%인 1,784명
  - 기타 인력이 30.8%인 1,201명, 마케팅/영업 인력이 11.0%인 431명, 연구직 인력이 7.1%인 276명, 일반사무/기획/관리 인력이 5.4%인 209명으로 집계됨
  - SW 기술 인력의 경우 게임 SW 업종에서 72.0%, 패키지 SW 업종에서 55.5%, IT서비스 업종에서 47.7%의 수요가 있는 것으로 조사됨
  - 인터넷서비스 업종의 경우 28.3%, 임베디드 업종에서는 17.4%으로 상대적으로 SW기술 수요 비중이 낮은 것으로 집계됨

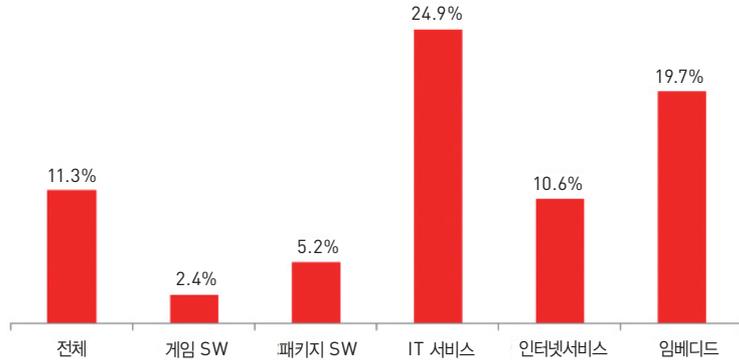
〈표 9〉 업종 및 직무별 2016년 신규 채용 예정 인력

구분		SW 기술	일반사무/ 기획/관리	마케팅/ 영업	연구직	기타	계
게임SW	채용인력수	448	33	60	49	32	622
	비중	72.0%	5.3%	9.6%	7.9%	5.1%	100%
패키지SW	채용인력수	305	42	91	86	26	550
	비중	55.5%	7.6%	16.5%	15.6%	4.7%	100%
IT서비스	채용인력수	728	66	123	83	526	1,526
	비중	47.7%	4.3%	8.1%	5.4%	34.5%	100%
인터넷서비스	채용인력수	244	50	113	31	425	863
	비중	28.3%	5.8%	13.1%	3.6%	49.2%	100%
임베디드	채용인력수	59	18	44	27	192	340
	비중	17.4%	5.3%	12.9%	7.9%	56.5%	100%
계	채용인력수	1,784	209	431	276	1,201	3,901
	비중	45.7%	5.4%	11.0%	7.1%	30.8%	100%

## (3) R&amp;D 투자 전망

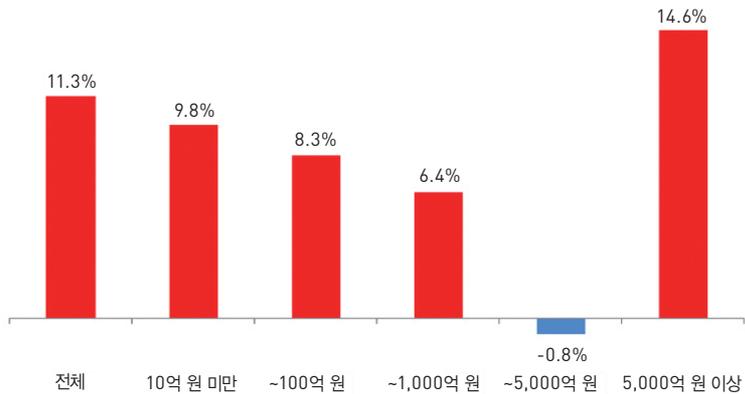
- 2016년 R&D 투자는 2015년 대비 11.3% 증가할 전망
  - 업종별로는 IT서비스 업종의 R&D 투자 증가율이 24.9%로 가장 높은 수준을 보일 것으로 나타남
  - 임베디드 업종 19.7% 증가, 인터넷서비스 업종 10.6% 증가, 패키지 SW 업종 5.2% 증가, 게임 업종 2.4% 순

〈그림 17〉 업종별 2016년 R&amp;D 투자 증감률



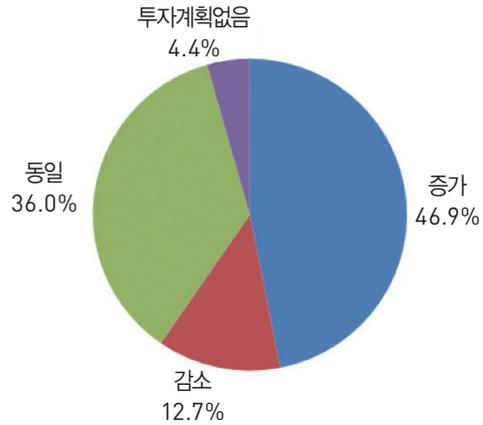
- 매출액 규모별로는 5,000억 원 이상의 대기업에서 14.6% 증가할 것으로 예상되나, 1,000억 원 이상 5,000억 원 미만 기업은 0.8% 감소할 것으로 전망

〈그림 18〉 매출액 규모별 2016년 R&amp;D 투자 증감률



- 2016년 R&D 투자 증가를 예상하는 기업은 46.9%인 151개 기업
  - 동일한 수준을 예상하는 기업은 36.0%인 116개, 감소를 예상하는 기업은 12.7%인 41개, R&D 투자 계획이 없는 곳이 4.4%인 14개 기업

〈그림 19〉 2016년 R&D 투자 증감 전망



- 업종별로는 게임 SW 업종이 R&D 투자 증가가 예상되는 기업이 60.6%으로 가장 크고, 나머지 업종은 비슷한 수준
- 전체 연구개발지수(R&D Survey Index, RSI) 평균은 135.7

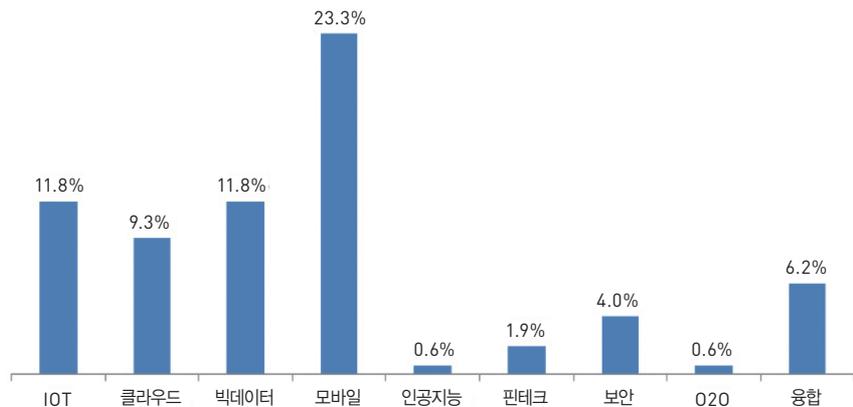
〈표 11〉 업종별 인력 증감 예상 기업 현황

구분	동일	증가	감소	투자 계획 없음	계	연구개발지수
게임SW	5	20	7	1	33	140.6
패키지SW	42	51	12	9	114	137.1
IT서비스	44	42	8	2	96	136.2
인터넷 서비스	18	21	4	1	44	139.5
임베디드	7	17	10	1	35	120.6
계	116	151	41	14	322	135.7

주 : 연구개발지수 100은 보합, 100보다 크면 호전, 100보다 작으면 악화를 의미

- 신(新)사업 추진계획이 있는 기업은 136개(응답 신사업분야는 244개)
  - 동일한 수준을 예상하는 기업은 36.0%인 116개, 감소를 예상하는 기업은 12.7%인 41개, R&D 투자 계획이 없는 곳이 4.4%인 14개 기업
  - IoT와 빅데이터 분야에 대한 신사업 추진 계획 비율은 각각 11.8%씩, 클라우드는 9.3%, 융합 분야는 6.2%으로 나타남
  - 기타 인공지능, 핀테크, 보안, O2O 신사업 추진 계획 비율은 5% 미만으로 조사됨

〈그림 20〉 신(新)사업 추진 분야 및 비중



주 : 복수응답 허용

## 시사점



- 2016년 SW 수출 성장률은 8%로 국내 SW시장 성장률 인 4.3%<sup>(2)</sup> 및 국내 경제 성장률 인 2.8%<sup>(3)</sup>를 훨씬 상회할 전망
  - 기업당 평균 수출액은 894억에서 966억 원으로 증가할 것으로 전망 : 평균수출규모에서는 IT서비스(2,389억 원), 임베디드SW(1,623억 원)가 주도하고, 수출성장률 측면에서는 게임SW(35.9%), 인터넷서비스(33.9%)가 호조를 보일 전망
  - 수출실적이 없는 SW기업이 61.8%에 달하고, 수출지역 역시 아시아 국가에 편중되어 수출성장의 기회가 많은 상황 : SW 수출 활성화를 통한 글로벌 시장의 확대를 위한 정책을 좀 더 적극적으로 펼쳐야 함
- SW기업의 인력은 전년대비 2.0% 증가하며, 신규채용규모는 전체종사자수 대비 4.1%에 이를 전망
  - SW기업의 고용실사자수는 130.1 : 전체산업을 대상으로 한 경제전망조사<sup>(4)</sup>에서 드러난 고용실사 지수 88.6보다 월등히 높아서 SW산업이 고용시장을 이끌 것으로 기대됨
  - 유효응답기업 322사의 신규 채용규모는 3,901명으로 기업별 평균 신규채용 규모는 14명 수준 : 직무 별로는 절반(45.7%)이 SW기술인력이며, 게임 업계가 SW기술인력 채용비중이 높은 것으로 드러남
  - 기업별 평균 R&D투자 예상 금액은 115억 원 : SW산업의 R&D 활동이 크게 개선될 것으로 보여 글로벌 경쟁력 강화에 도움이 될 것으로 전망

(2) 소프트웨어정책연구소, 2016 소프트웨어산업 전망과 대응

(3) 현대경제연구원, 2016년 한국 경제 전망

(4) 한국경영자총협회, 2016년 최고경영자 경제전망 조사 결과

## 조사의 한계

- 응답기업의 대표성 부족

- 조사과정에서 유효응답기업의 업종 및 매출액 구간별 분포가 전체 SW기업 모집단과 다소 차이가 발생  
: SW기업 모집단은 매출액이 100억 미만의 비중이 대다수이나 조사결과에 포함된 유효응답기업들은 100억 이상 1,000억 미만이 비중이 상대적으로 높음

〈표 12〉 업종 및 매출액별 SW기업 전체모집단 및 유효응답기업 분포

구분		10억 미만	~ 100억 미만	~ 1,000억 미만	~ 5,000억 미만	5,000억 이상	계
게임 SW	기업수	433 (0)	211 (10)	59 (14)	8 (5)	4 (4)	715 (33)
	비중(%)	60.6(0)	29.5 (30.3)	8.3 (42.4)	1.1 (1.5)	0.6 (1.2)	100 (100)
패키지 SW	기업수	6,049 (11)	4,668 (46)	363 (53)	9 (4)	0 (0)	11,089 (114)
	비중(%)	54.5 (9.6)	42.1 (40.4)	3.3 (46.5)	0.1 (3.5)	0 (0)	100 (100)
IT 서비스	기업수	3,024 (10)	2,862 (22)	512 (43)	66 (16)	8 (5)	6,472 (96)
	비중(%)	46.7 (10.4)	44.2 (22.9)	7.9 (44.8)	1.0 (16.6)	0.1 (5.2)	100 (100)
인터넷 서비스	기업수	954 (2)	293 (5)	89 (24)	18 (8)	7 (5)	1,361 (44)
	비중(%)	70.1 (4.5)	21.5 (11.4)	6.5 (54.5)	1.3 (18.2)	0.5 (11.4)	100 (100)
임베디드	기업수	47 (3)	96 (4)	106 (13)	24 (12)	6 (3)	279 (35)
	비중(%)	16.8 (8.6)	34.4 (11.4)	38.0 (37.1)	8.6 (34.3)	2.2 (8.6)	100 (100)
계	기업수	10,507 (26)	8,130 (87)	1,129 (147)	125 (45)	25 (17)	19,916 (322)

- 2015년 매출 및 수출액의 경우 실제 데이터보다는 응답자의 추정치로 집계

- 향후 실제 매출 및 수출액 데이터를 추적 조사를 통해 현재 조사 및 분석 결과의 보정 작업이 필요

## 참고문헌

### 국내문헌

- 소프트웨어정책연구소. (2015). 2016 소프트웨어산업 전망과 대응
- 소프트웨어정책연구소. (2016). 국내 소프트웨어 기업 경쟁력에 관한 연구
- 한국경영자총협회. (2015). 2016년 최고경영자 경제전망 조사 결과
- 현대경제연구원. (2015). 2016년 한국경제 전망



# 소프트웨어 산업 생산성 향상을 위한 원격개발환경 활용방안



## Executive Summary

- 정보기술의 발달로 산업 구성에서 지식산업이 차지하는 비중이 높아지고 있고, 기존 산업이 지식 산업으로 이행하는 현상이 가속화 되고 있음
- 지식 산업 내 지식 근로자가 건강, 육아, 가정 문제 등으로 경력이 단절되는 경우 산업 현장에 복귀하는 것에 많은 어려움을 겪는데, 이러한 현상은 소프트웨어 관련 종사자에게도 마찬가지로 적용 됨
- 소프트웨어 개발자의 경우 고객을 대면하거나 고객의 평판이 업무 생산성에 직접적으로 영향을 주는 분야가 아님에도 불구하고, 타 지식 산업 근로자 이상의 어려움을 겪고 있으며, 그 이면에는 소프트웨어 개발자를 채용할 때 발생하게 되는 기업의 부담이 자리 잡고 있음
- 지식 근로자로서의 소프트웨어 개발능력을 가진 조기퇴직, 경력단절 인력을 활용하기 위해서는 기업의 채용 부담을 줄이고, 그들이 가지고 있는 근무환경, 근무시간을 비롯한 장애요소를 해결하는 방안을 제안해야 함
- 최근 해외를 중심으로 사용량이 늘어나고 있는 클라우드(Cloud) 개발 환경은 원격 개발환경으로서의 기능을 탑재하고 있는 추세임
- 원격 개발 환경의 기술적 성숙도에 대해, 현장 적용의 실효성이 검증된다면 조기퇴직, 경력단절 소프트웨어 개발인력의 업무 장애요소를 해결할 수 있는 방안으로서 사용될 수 있을 것임
- 본 연구의 평가 결과는 다음과 같음
  - 웹 통합개발환경은 이미 소프트웨어 개발에서 원격 개발에 필요한 기능을 충분히 제공하고 있고 더 나아가서 개발의 생산성과 효율성을 제고할 가능성을 갖고 있음
  - 그러나 원격 개발 환경 조성은 기능적인 불완전성 및 개선 사항 등으로 인하여, 현재까지는 제한적인 플랫폼(소규모 개발 중심)에 한정하여 현장 적용이 가능할 것으로 보임
- 결론적으로 소프트웨어 개발기업이 효과적으로 자본을 투자하고 업무 효율을 올리면서, 소프트웨어 개발자들이 경력 단절을 두려워하지 않을 수 있도록 클라우드 기반의 소프트웨어 개발 시스템 활용을 촉진할 필요가 있음

김종규 국민대학교 교수  
jkkim.phd@gmail.com

김정민 연구원  
jungmink26@spri.kr

## 배경 및 필요성

정보기술의 발달로 선진국의 사회구조는 정보기술을 통해서 고부가가치를 생산하는 지식사회로 진입하고 있다. 지식사회에서 자신의 지식을 자본으로 하여 소득을 얻는 사람들은 지식근로자라고 부르는데, 전형적인 예로는 소프트웨어 개발자, 의사, 약사, 건축가, 엔지니어, 과학자, 회계사, 변호사, 학자 등이 해당된다. 지식근로자는 다른 표현으로 '생각하는 것으로 생계를 이어가는 사람들'로 정의되기도 한다.<sup>(1)</sup> 지식사회의 근간이 정보기술의 발전에 크게 의존하고 있기 때문에, 각국은 소프트웨어개발자를 포함한 다양한 분야의 지식근로자를 확보하기 위해서 많은 노력을 기울이며 경쟁하고 있다. 뿐만 아니라 이들의 생산성을 향상시키고, 교육하는 데도 많은 노력을 기울이고 있다.<sup>(2,3,4)</sup>

그러나 아이러니하게도 이렇게 많은 노력을 들여 확보한 지식근로자가 건강, 육아, 가정문제 등으로 업무를 중단하게 되면 지식산업현장으로 복귀하는 것도 어렵고 이들의 복귀를 촉진하는 제도나 정책도 매우 부족한 것이 현실이다.<sup>(5,6)</sup> 지식근로자들 중 의사나 변호사의 경우 고객의 평판이 전문 지식인으로 활동하는데 크게 영향을 주기 때문에 한동안 고객과의 관계가 단절된다는 것이 부정적으로 인식되어 경력단절로 연결되는 것은 쉽게 이해할 수 있는 일이지만, 소프트웨어개발자의 경우에는 전문지식을 습득하려는 자신의 노력에 따라 얼마든지 능력을 인정받아 현업에 복귀할 수 있음에도 불구하고 역량 있는 소프트웨어 개발자들이 현장으로 복귀할 때 겪는 어려움은 다른 지식근로자와 마찬가지로 더 높은 수준이다.<sup>(7)</sup> 그 이유는 무엇일까?

경력단절의 원인은 다양하지만 많은 경우 건강, 육아, 가정문제 등의 범주에 속한다.<sup>(8)</sup> 소프트웨어개발자도 예외는 아니지만, 다른 지식근로자와 비교하면 소프트웨어개발자는 프로젝트를 기반으로 잘 짜인 조직에서 활동하게 되고, 정보시스템을 통하여 실시간으로 교환하는 정보의 양이 다른 지식근로자에 비해 월등히 많고, 조직 내에서의 원활한 의사소통이 생산성에 미치는 영향이 매우 크다는 차이점이 있다. 따라서 대부분의 소프트웨어 개발조직에서는 많은 시간을 할애하여 개발조직에 쉽게 동화될 수 있는 사람을 선호하고, 실시간으로 많은 양의 정보를 공유할 수 있도록 특별히 구성된 작업공간에서 근무할 수 있는 사람을 선호하게 된다. 여기에 개발에 관한 전문 지식 뿐 아니라 문서 및 구두로 의사소통이 원활한 사람이라면 더 높은 평가를 할 수 밖에 없을 것이다. 이런 이유로 애초에 경력단절에 이를 수밖에 없는 사정이 있었던 소프트웨어 개발자라면 근무지를 선택하고 시간을 할애하는 측면에서 여러 가지 불리한 조건에 있게 됨을 부인할 수 없다.

(1) Thomas H. Davenport. Thinking For A Living: How to Get Better Performance and Results From Knowledge Workers. Boston: Harvard Business School Press, 2005.

(2) Managing knowledge workers – getting the most from them. [https://www.mindtools.com/pages/article/newTMM\\_45.htm](https://www.mindtools.com/pages/article/newTMM_45.htm).

(3) Gloria Ferguson Pobst. Meeting the challenge of knowledge worker shortages with strategic talent management. American Journal of Management, 14, 2014.

(4) Towards a knowledge society. [http://www.mssresearch.org/?q=Towards\\_a\\_Knowledge\\_Society#\\_Toc29534747](http://www.mssresearch.org/?q=Towards_a_Knowledge_Society#_Toc29534747).

(5) Career breaks: return journey. <http://www.lawgazette.co.uk/analysis/career-breaks-return-journey/70345.fullarticle>.

(6) When the doctor returns to doctoring. [http://well.blogs.nytimes.com/2013/01/10/when-the-doctor-returns-to-doctoring/?\\_r=0](http://well.blogs.nytimes.com/2013/01/10/when-the-doctor-returns-to-doctoring/?_r=0).

(7) How to become a software developer without a degree. <http://www.theguardian.com/careers/careers-blog/how-to-become-a-software-developer>.

(8) Why 43% of women with children leave their jobs, and how to get them back? <http://www.theatlantic.com/saxes/archive/2013/04/why-43-of-women-with-children-leave-their-jobs-and-how-to-get-them-back/275134/>, 2013.

## 목적

이 연구에서는 건강, 육아, 가정문제 등의 한시적인 이유로 경력이 단절된 소프트웨어 개발자들이 소프트웨어 산업현장에 효과적으로 복귀하도록 하여 소프트웨어 산업 전반의 생산성을 향상시키는 것을 목표로 소프트웨어산업의 생산 활동을 간략히 조망하고 경력단절이 있는 소프트웨어 개발자들이 좀 더 용이하게 소프트웨어 개발 현장에 복귀할 수 있도록 대안을 제시한다. 구체적으로는 소프트웨어 개발 조직에서 경력 단절이 있는 개발자를 채용하는데 있어 장애가 되는 요소들을 고찰하고 이 요소들을 제거 하거나 완화 하는데 클라우드 기반의 개발환경이 갖는 가치와 그것을 활용하는 방안을 제시한다.

클라우드 개발환경은 근무시간을 일별, 혹은 주별로 조절할 수 있고, 근무지의 선택에 따르는 장애요소를 대폭 완화시킬 수 있다는 측면에서 많은 가능성이 있다. 이러한 가능성을 지식근로자의 효율적이고 효과적인 활용이라는 구체적인 성과로 연결하기 위해서는 어떠한 노력이 필요한 지를 소프트웨어 개발이라는 지식산업분야에서의 적용현황을 통해 어느 정도 업계에 확산할 가능성이 있을지를 고찰해 본다. 이러한 고찰을 바탕으로 향후 클라우드 개발환경에서 기술적으로 개선되어야 하는 부분과 이를 수용하기 위해서 소프트웨어 개발조직의 구성이나 개발절차상 변화해야 하는 요소들을 제시한다. 이 글은 다음과 같은 측면에서 의의를 갖는다.

- 지식산업현장에 복귀하는 소프트웨어 개발자에게 적합한 근무환경을 제시하고, 이를 클라우드 개발환경을 이용하여 제공하는 방안을 제시한다.
- 기업이 클라우드 개발환경을 이용하여 생산성을 높일 수 있음을 설명하고, 도입 가능한 클라우드 시스템을 설명한다.
- 향후 소프트웨어 산업 전반에 클라우드 개발환경이 적용되기 위해서 클라우드 환경에 기반한 개발절차에 대하여 논의한다.

## 연구방법 및 범위

연구의 대상은 국내외 다양한 문서와 실태조사결과이다. 소프트웨어개발과 관련한 방법론을 기술한 문서와 소프트웨어개발을 위해 현장에서 사용하는 기술과 제품을 파악한 후, 이에 대응되는 클라우드 기반 소프트웨어 솔루션 등을 파악한다. 또한 개발자의 블로그, 개발자 커뮤니티의 질의응답 등을 참고하여 개발자들이 현업에서 활용하고 있는 기술이 무엇인지도 파악한다. 이렇게 파악된 제품과 서비스에 대해서는 각각에 대하여 사용자 매뉴얼과 사용 후기 등의 제품관련 문서도 조사의 범위에 포함한다.

연구방법은 문헌에서 조사된 내용을 바탕으로 문헌 조사 결과와 실제 시스템을 사용한 경험을 기초로 주관적인 평가를 진행한다. 소프트웨어개발은 분석, 설계, 구현, 검사, 출시의 단계로 구분할 수 있는데, 각 단계에서 사용되는 방법론과 이를 지원하는 기술들을 파악하여 클라우드 환경에서 활용 가능한지를 검토한다. 특히 소프트웨어 개발자들에게 원격 근무를 수행할 수 있는 환경을 제공하기 위해서 클라우드 개발 환경에 대한 분석과 비교평가가 필요하기 때문에 선택된 시스템을 시험해 볼 수 있는 경우 직접 시험계정을 생성하고 활용한 후 평가항목을 도출하여 이를 바탕으로 주관적인 평가를 진행한다.

각각의 클라우드 시스템들에 대해서는 시스템을 활용할 수 있는 위치, 소스 코드의 공개여부를 표시하고, 소스 코드가 공개된 경우 소스 코드를 검토할 수 있는 방안도 설명한다. 이와 같은 기술적인 분석결과에 더하여 서비스를 제공하는 기업의 사업적인 실행력과 향후 기술적인 비전을 고려하여 전반적인 성능과 기능에 대한 평가를 수행한다.



## 1. 소프트웨어 산업의 생산 활동

### (1) 소프트웨어 개발 프로세스

소프트웨어 개발을 매우 거칠게 단순화시켜 설명하면 크게 어떤 일을 수행하도록 할 것인지, 그 일을 어떤 방식으로 수행할 것인지를 결정한 후에 실제 동작할 수 있는 프로그램코드를 작성하는 단계를 거친다. 소프트웨어의 복잡성을 고려하면 이런 일련의 작업을 한 번에 성공시킬 수 없기 때문에 소프트웨어 개발자가 작성한 코드를 검증하고 발견된 오류를 수정하는 과정을 반복하여 제품으로서의 모든 요건을 충족했는지에 대한 최종적인 합의가 이루어진 후에 소프트웨어로서 시장에 나오게 된다.

그러나 시장의 성격, 사용자의 요구사항, 사용할 수 있는 기술의 제한이나 응답속도, 처리량, 보안 등의 기술적인 부분을 모두 고려하고 최선의 제품을 만들어내는 과정은 매우 많은 대안들을 고려하고, 전문가의 검토를 거쳐 의사결정을 내려야한다. 또한 이 사안들을 각각의 소프트웨어 개발전문가에게 전달하여 개발을 진행한 후 결과를 검토하는 매우 복잡한 과정도 뒤따르게 된다.

따라서 소프트웨어 개발에서는 특별한 소프트웨어 도구를 이용하여 소프트웨어와 관련된 의사결정과 진행사항을 관리하는 것이 보통이다. 소프트웨어 개발이 다른 개발 작업과 비교하여 특이한 점은 이와 같은 의사결정사항이나 진행사항, 담당자 등의 정보가 조직 내에 공유되는 정도가 높으면 높을수록 소프트웨어 개발의 생산성이 증가된다는 점이다. 이에 따라 개발 현황과 관련한 정보는 별도의 문서를 사용하는 것 보다는 <그림 1>과 같이 웹을 이용하여 누구나 편리하게 정보에 접근하고 필요한 경우 수정할 수 있도록 시스템을 갖추어 정보를 관리하는 것이 일반적이다.

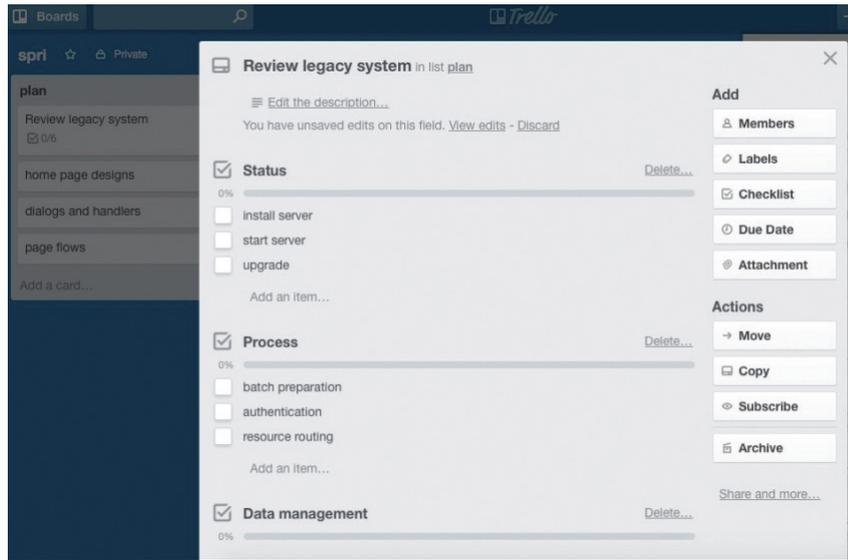
이런 시스템을 어떻게 활용할 것인가는 개발 조직의 문화에 크게 영향을 받는다. 예를 들어 가상의 어떤 조직에서는 단계별로 정보를 교환할 때 다음과 같이 템플릿 작성에 대한 규약을 정의하고 이에 맞추어 정보를 교환할 수 있다.

- **Work item (requirement)**

- title : 쉽게 이해할 수 있는 짧은 제목을 붙인다.
- description : 관심 있는 사람이 좀더 잘 이해할 수 있도록 한다.
- validation list : 주어진 기능이 구현되면 검토할 항목을 기술한다.
- founding facts : 주어진 기능을 작성하게 된 근거(email 요청 등)를 기술한다.



〈그림 1〉 Trello 작성템플릿예제



- Refine (design)
  - data model : ER diagram 등 data model의 위치와 참조한 항목을 기술한다.
  - input/process/output (I/P/O) : 입력과 출력으로 사용하는 데이터의 형식과 위치 등을 기술하고 이들을 변환하는데 사용할 절차를 간단히 기술한다.
  - verification list : 위의 작업이 정상적으로 이루어졌을 때 이를 확인 할 수 있는 방안을 기술한다. 필요한 경우 예제자료도 작성한다.
- Coding (implementation)
  - source code : 설계된 기능을 구현한다.
  - build script : 이 기능이 실행될 수 있도록 필요한 library를 확보하고 컴파일한다.
  - test data/code : 구현된 기능이 정상적으로 동작하는지 확인하기 위해 데이터를 입력하여 확인하고, 필요한 경우 동작확인을 위한 코드를 작성한다.
- Review (testing)
  - checklist (validation, verification) : 요구분석 및 설계 과정에서 작성한 list를 확인한다.
  - test script : 이 과정을 수행하거나 자동화하는데 필요한 스크립트를 공유 가능한 장소에 저장하고 그 위치를 기록한다.
- Release (quality assurance)
  - quality report : 성능, 보안 등 비기능적 성질을 포함하여 개발된 시스템의 전반에 대한 평가를 작성한다.
  - final approval : 고객, 개발자, 테스터 등이 모두 확인했다는 사실을 최종적으로 확인 받고 이를 문서로 저장한다.

## (2) 단계별 작업내용

지식사회에서 지식의 생산과 유통은 정보기술, 특히 소프트웨어에 의해서 이루어진다. 또한 소프트웨어 산업은 지식사회를 운영하는 데 필요한 요구를 파악하여 효율적으로 정보기술들을 활용하는 방안을 제공하는 데서 시작한다. 사용자의 요구를 파악하기 위해서는 현장방문, 인터뷰 등을 통한 시장조사를 선행하여 진행하는 것이 보통이고 이를 통해 바람직한 소프트웨어의 대략적인 형태를 도출한다. 그 후 이것이 현행 기술로 개발 가능할 것인지를 파악한 후 사업적으로 의미 있는 제품이나 서비스가 될 수 있을 것인지를 최종적으로 판별하게 된다. 소프트웨어 기업에서 주로 이 단계는 마케팅이나 기획 등의 사업부서 등이 주도하며 경력 있는 소프트웨어 아키텍트가 기술적인 검토를 함께 진행한다.

이 단계에서 소프트웨어 아키텍트는 해당 기업이 보유하고 있는 기술과 도입 가능한 기술, 새로 개발해야 하는 기술을 분류하여 사업성 검토에 반영한다. 따라서 능력 있는 소프트웨어 아키텍트의 경우에도 아직 등으로 조직의 역량에 대한 이해가 없으면 참여하기 어려운 분야이기 때문에 업무 인수인계를 위해 전임자와 후임자가 근무하는 기간의 중첩을 길게 잡거나 신입 아키텍트를 지원하고 업무 연속성이 유지되도록 엔지니어 그룹이나 보드의 형태로 조직하는 경우도 많다. 또한 영업 및 기술 관련하여 기밀로 분류되는 문서에 대한 접근권한도 제공되는 것이 일반적이기 때문에, 이전 경력에서 뛰어난 성과를 보인 소프트웨어 아키텍트라 하더라도 일정 기간 경력이 단절된 후에 다시 이 단계에서 공헌하기 위해서는 조직적인 지원은 물론 다양한 방법으로 기술 문서에 접근할 수 있도록 지원해야 한다.

소프트웨어의 수요와 사업성을 확인하면 소프트웨어의 구동방식을 구체화 시키는 단계가 필요하다. 어떠한 형식의 정보를 어떤 사람이나 정보시스템에서 획득하여야 하는가, 그 정보를 가공하여 어떤 형태로 사람이나 시스템에 전달하여야 하는가, 이 과정에서 법률적인 요건을 만족하기 위해서는 어떤 기록이 남아야 하는가와 같은 질문에 응답할 수 있어야 한다. 소프트웨어 개발자는 이 단계에서 정보시스템에 저장된 자료의 형식과 의미를 분석하고 이를 변환하기 위한 절차에 대한 개략적인 방안을 제시하는 등의 역할을 한다. 이 작업은 형식 없이 문서형태로 진행될 수도 있으나 이 경우 실제 소프트웨어 개발에 들어갔을 때 애매한 부분이 뒤늦게 발견되어 사업화에서 문제가 되는 경우가 종종 발생한다.

따라서 오늘날에는 이 부분을 명확히 하기 위해서 프로토타입을 작성하는 방법, 개발의 한 부분으로 포함하여 지속적으로 개선하는 방법 등이 사용되는 것이 일반적이다. 따라서 일정 기간 경력이 단절된 개발자가 이 단계에서 참여하기 위해서는 각종 자료나 정보를 명확하게 전달받을 수 있어야 하고, 의문사항은 쉽게 질문할 수 있어야 한다. 또한 작업결과가 구동되는 소프트웨어의 형태로 전달될 것이기 때문에 결과를 받아서 활용하는 부서나 담당자가 손쉽게 개발된 소프트웨어를 구동시켜서 확인할 수 있는 환경이 갖춰져야 한다.

소프트웨어 시스템이 정보를 획득하고 처리하는 방안이 구체화되면 이를 사업성 있는 소프트웨어 시스템으로 개발하기 위한 설계과정을 거치게 된다. 전통적인 공학 설계와 달리 소프트웨어는 사용과정에서 지속적인 개선이 이루어져야하기 때문에 성능뿐 아니라 운영 및 관리측면에서의 고려가 설계에서 중요한 요소가 된다. 많은 경우에 성능을 희생시키고 운영성이나 관리성을 의미 있게 높일 수 있는 방안이 있다면 이 방안이 선택되는 것이 보통이다.

소프트웨어 설계는 규범화된 방식이 있는 것이 아니기 때문에 다양한 방식이 사용된다. 전통적으로는 블록 다이어그램으로 전반적인 구조를 설명하고 사용자 눈높이에서 시스템의 활용법을 기술하는 방식으로





상위구조를 설계하는 것이 보통이었으나 이 때 엄밀한 검증이 이루어지지 못해서 구현단계에서 설계를 대폭 수정해야 하는 상황이 발생하기도 한다.

따라서 현대적인 설계기술은 블록다이어그램을 이용하여 대략적인 구조를 설계하고 데이터의 형식과 저장 등 매우 상세한 명세가 필요한 부분은 ER 다이어그램 등 상위 수준 기술뿐 아니라 프로토타입 시스템을 구현하여 사용자의 검증을 받는 방식으로 진행하는 것이 보통이다. 이 때 다른 정보시스템에서 획득하는 정보의 형태, 처리를 위해서 저장하는 방식이나 포맷 등은 재활용하고 정보의 처리 방식 자체는 성능이나 보안 등을 고려하지 않고 진행한다.

따라서 이후 구현과정에서 재활용하지 않고 지속적으로 재개발하는 것이 보통이다. 경력 단절이 있는 개발자의 경우 시스템의 전반적인 요구사항을 명확히 파악할 수 있고, 설계를 구체화하기 위해서 필요한 정보를 명확히 얻을 수 있다면 정보의 저장방식이나 포맷 등을 설계하고 이를 처리하는 과정을 실제 동작하는 코드로 보여줄 수 있을 것이다.

소프트웨어 시스템의 설계가 완료되면 어디에서 어떤 정보를 획득하여 어떻게 처리해야 하는지, 그 과정에서 정보는 어떻게 보호할 것인지, 그리고 성능 면에서 어떤 수준을 기대하는지가 결정된다.

다음 단계는 이러한 내용을 기초로 원하는 기능과 성능을 만족시키는 시스템을 구현하는 것이다. 이를 완수하기 위해서 개발과정에는 기능이 정상적으로 동작하는지, 원하는 성능이 도출되는지를 검사하는 테스트작업이 함께 지속적으로 반복된다. 또 향후 소프트웨어의 개선이나 관리를 위해서 소프트웨어에 대한 문서를 작성하는 작업도 진행하게 된다. 이 단계는 일정을 줄이기 위해 병행적으로 진행되는 것이 보통이고 많은 인력이 투입된다. 따라서 전체 작업은 하나의 컴퓨터에서 진행되기보다는 다수의 컴퓨터를 이용하여 진행하게 되는데 여기서 많은 문제가 발생한다. 예를 들어 컴퓨터의 용량이나 설치된 소프트웨어의 버전, 작업 중인 소스 코드의 상태 등이 일치하지 않는 경우 소프트웨어가 오동작하거나 아예 동작을 시작하지도 못하는 등의 문제가 발생하게 된다.



이와 같이 다수의 컴퓨터를 이용하여 병행적으로 소프트웨어를 구현하는 데서 발생하는 문제를 줄이기 위해서 현대적인 소프트웨어 기업은 개발에 필요한 도구를 표준화하여 보급하고, 이를 이용하여 작업하는 설정도 표준화한다. 또 소프트웨어 구동에 필요한 runtime 환경도 표준화하여 제공하는데, 상업용 소프트웨어 개발 경험을 돌이켜 보면 이런 정보를 하나의 파일로 표준화하는 것은 네트워크로 공유하기에는 부담스러운 크기가 된다. 더구나 구현이 진행되면서 이 내용들은 자주 변경되는 것이 보통인데, 그 때마다 동시에 많은 인원이 해당 내용을 내려 받아야 한다면 네트워크 지연으로 업무 생산성이 크게 저하된다.

이런 부분은 개발자가 원격지에서 근무하기를 원하더라도 이를 소프트웨어 개발사가 지원하는데 큰 장애가 되는 요소이다. 경력단절자가 복귀 초기에 구할 수 있는 직장은 이와 같이 단기간에 많은 사람이 필요한 부분에서 발생할 가능성이 높는데, 이런 업무는 영구적이라기보다는 일시적이고 업무 필요성이 종료되면 다른 직장을 구해야 하는 상황이 될 수 있다. 따라서 직장이 구해졌다고 쉽게 거주지를 변경할 수 있는 것은 아니다. 이 경우 근무지와 거주지 간의 쉼이 발생하게 되는데, 이 때 원격근무가 가능하다면 경력단절자가 복귀 초기에 직장을 구하는데 큰 도움이 될 것이다.

소프트웨어가 개발되어 충분히 테스트된 후에는 다양한 고객사에 소프트웨어의 실행본이 전달되고 이를 컴퓨터에 설치하고 운영하는 단계를 거치게 된다. 일반 사용자용 소프트웨어의 경우에는 그 필요성이 낮은 편이지만 비즈니스 소프트웨어의 경우 설치와 운영에 대한 전문적인 지식이 필요한 경우가 많기 때문에 별도의 인력을 파견하여 이를 지원하는 것이 업계의 표준 관행이다. 또 일반 사용자용 소프트웨어의 경우에도 사용자 문의를 처리하기 위해서 헬프 데스크 정도는 상시 운영하는 것이 일반적이다.

소프트웨어 설치와 운영에서 발생하는 문제의 해결에는 현장의 정보가 매우 중요한 역할을 한다. 이 정보는 현장에 방문하여 획득하는 것이 가장 정확하겠지만 비용 상의 이유로 원격지에서 컴퓨터의 화면이나 키보드를 제어하는 방식을 사용하기도 하는데, 오늘날 적용 분야가 점점 증가되는 추세에 있다. 따라서 경력단절자가 소프트웨어 설치 및 운영과 관련된 업무를 재개할 때는 이렇게 원격시스템의 도움을 받을 수 있는 분야가 적합하다.

문제는 소프트웨어 개발과 관련한 전문지식과 소프트웨어의 설치 운영과 관련한 전문 지식에 공통요소가 상대적으로 적다는 점에 있다. 따라서 소프트웨어 개발자가 설치 운영 경험을 통해서 소프트웨어 개발 업무에 복귀하는 것은 예상보다 대단히 어려운 과제가 된다.

소프트웨어 운영과 관련한 문제 중에서 가장 극복하기 어려운 부분은 소프트웨어의 성능과 관련된 부분이다. 갑작스럽게 소프트웨어가 동작하지 않게 되거나 단위 시간당 처리하는 정보의 양이 급격하게 감소하는 경우, 혹은 정보를 잘못 처리한 것으로 의심이 되는 경우가 발생하는데 원인을 찾지 못하는 경우가 발생하게 되면 지식 정보를 처리하는데 심각한 장애요인이 된다. 상업용 소프트웨어 시스템은 특성상 한 두 사람이 구현하기에는 업무가 나지 않는 매우 복잡한 문제를 해결하는 것이기 때문에 설계도 매우 복잡할 수밖에 없어 제품화 초창기에 이런 문제가 발생하는 것은 피할 수 없는 일이다. 하지만 이 상태가 장기간 지속되지는 않도록 조치해야 하는데, 이 때 필요한 정보를 수집하고 원인을 추정하여 재현하고, 문제를 해결하는 과정에서 소프트웨어 개발자가 매우 중요한 역할을 한다.

소프트웨어에서 성능과 관련한 문제는 피할 수 없는 것이기 때문에 소프트웨어는 설계와 개발과정에서 이런 문제가 발생할 때 도움이 되는 정보를 생성하고 저장하도록 고려하는 것이 일반적이다. 이 때 수집된 정보를 이해하고, 그 때의 상황을 재구성하기 위해서는 수집된 정보 뿐 아니라 소프트웨어를 소스 코드 수준에서 접근하여 해석하는 것도 필요하다. 또 수집된 정보를 기초로 상황을 구성하고 소스 코드를 실행할 수 있는 환경도 필요하다. 이런 작업은 해당 모듈을 잘 알고 있는 개발자만 가능한 작업이고 가능한 원저자가 수행하는 것이 바람직하다. 만일 해당 부분이 원격지에서 근무하는 개발자가 작성했다면, 성능 문제에 대한 검토와 실험을 원격지에서 실행할 수 있도록 지원하지 못할 경우 문제 해결에 소요되는 시간이 많이 증가할 우려가 있다.



## 2. 클라우드 환경을 활용한 소프트웨어 개발

육아, 건강, 가정 문제 등으로 경력이 중단된 지식노동자가 산업현장에 복귀하는 초기에 요구하는 사항은 대체로 재택근무와 명확한 성과지표, 그리고 명확한 마감일 등이다.<sup>(8)</sup> 이에 반하여 현재 소프트웨어 산업의 문화는 사회에 진입한 초기부터 지속적으로 업계에서 근무한 인력을 선호하고, 작업하는 방식도 같은 사무실에서 가능한 대면하여 의사를 교환하는 것을 선호한다. 그러나 인터넷으로 전송 가능한 정보의 양이 증가하고, 이를 활용하여 화상통신 등 다양한 방식으로 의사소통이 가능한 상황에서 대학이 배출하는 제한된 인력으로 소프트웨어 산업을 확장시키는 것에 소프트웨어업계에 종사하는 많은 사람들이 한계를 느끼고 있다.

따라서 화상통신 등 다양한 의사소통 도구와 클라우드에 기반 한 개발도구를 이용하여 원격지 개발을 지원하게 되면 역량은 있지만 경력 단절로 산업 현장에 복귀하지 못한 개발자들을 활용하여 소프트웨어 산업 전반의 생산성을 높일 수 있을 것이다. 이 절에서는 클라우드 기반 온라인 도구들을 이용하여 개발생산성을 높이는 방안에 대해서 설명한다.

### (1) 소프트웨어 개발방식의 변화

소프트웨어 개발에는 하드웨어와 소프트웨어가 모두 필요하다는 것이 상식이었지만, 가상화 기술의 발전으로 이와 같은 전제는 현재 빠르게 사라지고 있다. 전통적으로 소프트웨어의 개발은 사용자에게 실시간 피드백(Feedback)을 제공하는 워크스테이션(Workstation)에서 진행하고 실행은 메모리나 프로세싱 파워, 혹은 I/O 성능에서 우월한 서버(server)에서 수행했다. 그러나 적어도 서버의 경우에는 Amazon web service(AWS)와 같은 클라우드 환경을 활용할 경우 초기 진입비용이 낮기 때문에 상당수의 스타트업(Start-up)들이 클라우드 환경을 이용하여 서비스를 제공하고 있다.<sup>(9)</sup>

소프트웨어 개발에서는 고급 언어로 작성된 프로그램코드를 하드웨어에서 실행할 수 있는 형태로 번역하는 컴파일러라는 소프트웨어가 반드시 사용되는 데, 컴파일러의 품질에 따라 개발된 소프트웨어의 실행속도 등 성능이 영향을 받는다. 컴퓨터 산업의 초창기에는 컴파일러는 하드웨어 업체가 일종의 부가상품으로 제공하던 것이었지만 이제는 소프트웨어 전문업체가 독립된 상품으로 개발하여 보급하는 중요한 소프트웨어 제품이 되었다.

이 컴파일러라는 소프트웨어는 최종 개발되는 제품의 성능에 큰 영향을 주고, 편리하게 활용할 수 있도록 개발하는 데 많은 노력이 소요되기 때문에 전문업체가 개발하여 고가에 판매하는 것이 일반적이었으나, 현재는 무료로 사용할 수 있고 누구나 개선해갈 수 있도록 오픈소스로 개발된 컴파일러의 품질이 급격히 향상되어 별도의 비용 없이 고급언어로 작성된 소프트웨어 코드를 실행코드로 번역하는 것이 가능해졌다. 따라서 클라우드 환경에 설치된 서버에 오픈소스 컴파일러를 설치하여 서비스를 운영하는 것이 비용 면에서 매우 효과적인 방안으로 부상하였다.

(9) Amazon web service. <http://aws.amazon.com>. [Online; accessed 14Dec-2015].



전문 업체는 이와 같은 오픈소스 컴파일러의 도전에 대응하고자 단순한 번역 기능뿐 아니라 소프트웨어 개발을 편리하게 하도록 편집기, 디버거 등을 통합하고, 소프트웨어 개발에 사용된 함수나 변수를 손쉽게 추적할 수 있도록 IDE 를 제공하고 있다. 이와 같은 모든 기능을 통합하여 개발자에게 제공하는 경우 오픈소스 컴파일러만을 제공하는 경우에 비해서 개발생산성이 크게 높아지는데, 무료로 사용할 수 있는 오픈소스 도구들의 품질은 컴파일러 자체에 비해서는 조악한 실정인어서 전문개발자들에게 외면을 받았다. 그러나 최근 클라우드 환경에서도 편집, 오류추적, 프로그램관계분석 등 개발생산성에 도움을 주는 도구들이 상당한 개선을 이루어 고품질로 제공되기 시작하고 있다.

소프트웨어는 출시 당시의 편의성이나 생산성뿐 아니라 사용상 발생하는 문제점을 신속하게 해결하는 것도 제품으로써 성공하는데 중요한 요소가 된다. 출시된 제품, 혹은 테스트 중인 제품의 동작에서 이상한 현상이나 오류가 발생하는 경우, 프로그램의 수행을 추적하고 분석할 수 있는 도구는 소프트웨어 개발 생산성에 큰 영향을 끼친다. 소프트웨어제품이 PC 용 application과 같이 비교적 간단한 형태인 경우에는 컴파일러에서 제공하는 오류분석기를 이용하여 간편하게 프로그램의 수행 과정을 추적할 수 있지만 소프트웨어가 독립된 서버에서 실행되는 경우에는 네트워크를 통하여 실행을 추적할 수 있는 기능이 없이는 오류 추적이 매우 번거롭고 어려운 일이 된다. 이와 같은 기능을 원격 디버깅이라고 하는데, 원격 디버깅은 상대적으로 소수의 개발자가 사용하는 기술이었으나 현재는 여러 분야에서 이용되고 있고, 클라우드 환경에서 원격 디버깅 기능은 필수 불가결한 요소가 되었다. 대다수의 오픈소스 컴파일러 및 관련 소프트웨어 시스템에서도 지원하고 있다. 다만 아직까지는 일부 서버 개발자들에게만 알려진 기능인어서 보급과 확산에는 아직도 시간이 더 소요될 것이다.

효용성 있는 소프트웨어 시스템은 매우 많은 기능을 담게 된다. 이런 이유로 대부분의 소프트웨어 개발 작업은 개발자 한 사람이 담당하기에는 버거운 작업이 된다. 이렇게 여러 개발자가 소프트웨어 코드를 작성하는 경우 타인이 작성한 코드를 참조하거나 변경해야 하는 경우도 발생하고, 자신이 작성한 코드에서 호출하여 활용하는 경우도 발생한다. 이 때 다른 사람이 작성한 코드를 함부로 수정해서도 안 되지만, 여러 부분이 수정되지 않으면 자신이 담당한 과업을 완성할 수도 없다는 딜레마가 수시로 발생한다. 이와 같은 문제는 전문 소프트웨어 시스템을 이용하여 해결할 수 있는데, 그 중 가장 중요한 도구가 소스 코드 관리 시스템이다.

소스 코드 관리 시스템은 네트워크를 이용하여 개발자들이 작성한 모든 소스 코드를 한 장소에 저장하는 것을 기본으로 한다. 이 때 어떤 개발자가 소스 코드의 특정한 부분을 이미 변경하고 있는 경우, 다른 사용자들은 그 변경사항에 영향을 받지 않고 예전 버전으로 작업해야 하는데, 소스 코드 관리 시스템은 개발자들이 특별히 약속하지 않아도 이런 작업이 자연스럽게 진행될 수 있도록 지원한다. 이 시스템을 사용하면 변경된 내용이 있다는 사실이 최종적으로 다른 개발자에게 알려지는 것은 변경이 완료되고 변경된 내용에 대한 테스트까지 완료된 후이나 이루어진다. 그리고 이렇게 알리는 과정도 소프트웨어 개발자가 직접 수행하는 것이 아니라 시스템에서 알림 기능을 이용하여 자동으로 전달되어 안전하고 편리하게 소스 코드를 변경할 수 있도록 지원한다.

소스 코드의 변경뿐 아니라 화면의 구성이나 사용자의 요구사항, 혹은 일정상의 변경이 발생한 경우에도 소프트웨어 개발에 참여하는 모든 사람, 그러니까 개발자, 테스트담당자, 관리자, 그리고 고객에게까지도 정보가 공유되는 것이 소프트웨어 개발을 원활하게 진행하는데 크게 도움이 된다. 이러한 정보의 공유를 용이하게 하기 위해서 여러 가지 전문화된 소프트웨어 관리 도구들이 개발되고 보급되어 왔는데, 현재는 클라우드를 기반으로 web 에서 비전문가도 쉽게 접근하여 소프트웨어 개발과 관련한 정보에 접근하고 의견을 남기거나 필요한 경우 변경을 가할 수 있도록 지원하고 있다.



## (2) 가상컴퓨터와 원격접근

현대적인 소프트웨어 시스템은 인터넷으로 접근 가능한 서버에서 운영되는 것이 보통이다. 이와 같은 변화를 가장 극적으로 보여주는 산업 분야가 금융이다. 전통적으로 은행원의 책상에서 운영되던 은행의 정보 시스템은 인터넷의 보급과 함께 거의 대부분 웹 브라우저에서 동작하는 시스템으로 전환되었다. 이에 따라 이제는 인터넷이 연결되는 어디든지 정보시스템을 설치, 운영할 수 있게 되었고, 따라서 지점을 개설하거나 운영하는데 들어가는 비용을 획기적으로 낮출 수 있게 되었다. 한 걸음 더 나아가서 이제는 고객들이 은행 지점을 거치지 않고 직접 자신의 업무를 처리하는 단계까지 발전하여 인터넷 전용 은행의 탄생도 눈앞에 보게 되었다.

이와 같은 변화는 일반 사용자를 대상으로 하는 소프트웨어 스타트업의 경우 더 극적으로 반영되는데, 오늘날 대다수 스타트업들은 서버 운영에 필요한 초기 비용을 줄이기 위해서 클라우드 기반 시스템에 서비스를 설치/운영하고 있다. 가상 컴퓨터에 오픈소스 기반의 운영체제와 컴파일러 등을 설치하여 운영하게 되면, 많은 고객을 상대하는 서비스를 유지하는 경우에도 비용이 거의 들지 않는다. 따라서 요즘 스타트업들이 초기서비스에 소요하는 비용은 사실상 가상 서버의 임대료와 네트워크 비용 정도이고, 독자적인 서버를 구입하고 소프트웨어를 구입해야 했던 과거와 비교하면 초기 투자와 비용이 대폭 축소되었다.

외부에서 가상컴퓨터를 안전하게 관리하는 문제는 암호화 기술의 발전으로 많은 부분이 해결되었다. 더구나 최고 수준의 암호화 기술도 오픈소스 형태로 개발되어 거의 무료로 보급되고 있기 때문에 이 기술의 활용에도 별도 비용은 거의 소요되지 않는다. 따라서 이제는 개인정보와 같이 민감한 정보를 전송하거나 처리하는 서비스도 적절한 네트워크 설정을 거쳐 클라우드에 설치된 서버에서 제공할 수 있다.



## (3) 컴파일러와 패키지 라이브러리

소프트웨어는 C, Java, Python 등의 이름으로 불리는 고급언어로 개발된 후 컴파일러라는 소프트웨어를 통해서 구동 가능한 실행 파일을 변환된다. 특별한 영역에서 사용되는 특수한 종류의 프로그래밍언어를 제외하면 대다수 프로그래밍언어의 컴파일러로는 높은 품질의 오픈소스 제품도 존재한다. 그러나 소프트웨어개발에는 컴파일러 외에도 다양한 패키지라이브러리가 필요하고, 이 라이브러리들은 개발 과정 뿐 아니라 운영과정에도 큰 역할을 한다.



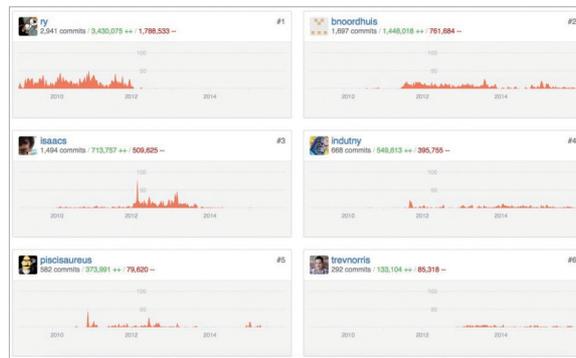
이러한 패키지 라이브러리는 전문업체에서 고가에 판매하는 것이 일반적이었다. 그러나 웹 기반 응용 프로그램이나 모바일 기반 응용 프로그램의 개발에 필요한 상당수 라이브러리들은 현재 많은 부분 오픈소스의 형태로 개방되고 있는 추세이다. 따라서 일반 사용자를 대상으로 하는 대부분의 소프트웨어 제품이나 서비스를 개발하는 데는 소프트웨어 개발자의 수와 역량을 확보하는 것 외에는 별다른 투자가 필요하지 않다. 다만 특별한 산업분야에 적용되는 특수한 소프트웨어 개발의 경우에는 일반적인 하드웨어나 소프트웨어 런타임을 활용할 수 없기 때문에 상대적으로 많은 규모의 투자가 필요할 수도 있다.

## (4) 클라우드 기반 소프트웨어 개발 도구

소프트웨어 개발은 많은 사람들이 참여하고 매우 복잡한 절차를 거쳐 이루어진다. 소프트웨어 개발은 소프트웨어가 어떤 일을 할 것인지, 어떻게 그것을 하도록 할 것인지를 결정할 후에 소프트웨어 코드를 작성하고 이를 실행할 수 있는 형태로 변환하는 것이다. 따라서 소프트웨어 개발에는 컴파일러, 편집기, 디버거, 라이브러리, 소스 코드 관리 시스템 등 많은 소프트웨어 부품과 시스템들이 사용된다. 여기서 소프트웨어 개발의 각 단계와 사용되는 정보 시스템을 간략히 설명하면 다음과 같다.

- 개발단계 : 대부분의 소프트웨어 개발에는 공통적으로 requirement, design, implementation, testing, release 등의 단계를 거친다.
- 산출물 : release를 제외하면 목표로 하는 소프트웨어 시스템이 동작하는 모습을 확인할 수 없다. 따라서 최종 목표물이 잘 제작되고 있는지 확인할 수 있도록 논의 내용이나 설계 내용 등을 중간 산출물로 작성하여 검토한다.
- documentation : 중간 산출물을 정확하고 손쉽게 이해하고 다음 작업을 진행할 수 있도록 정보를 생성하여야 한다. 이를 위해서 단계별로 어떤 정보가 필요한 지를 정의하고 이를 파악하여 기록한다.
- communication : 소프트웨어 개발의 성공은 효과적인 의사소통에 기반을 둔다. 생산성을 유지하면서 의사소통을 원활하게 유지할 수 있는 방안이 필요하다.
- automation : 소프트웨어 개발의 후반부로 가면 테스트와 개선을 지속적으로 반복하게 되는데, 이것은 매우 기계적인 작업으로 사람이 작업하는 경우 업무 만족도도 낮고 실수할 확률도 크다. 따라서 소프트웨어 후반부 작업의 많은 부분은 자동화 시스템의 도움을 받는다.

〈그림 2〉 Progress



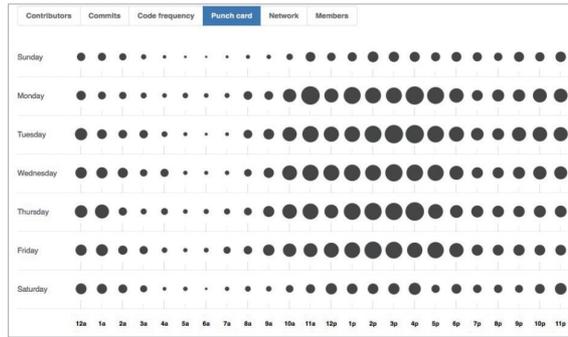
소스 코드를 공유하고 개발 현황을 공유하는 데는 GitHub라는 시스템이 사용된다.<sup>(10)</sup> GitHub는 git이라는 소프트웨어 시스템을 클라우드 환경에서 제공하는 서비스이면서 동시에 프로젝트 관리에 필요한 여러 가지 기능들을 제공하는 프로젝트 관리 사이트이기도 하다. GitHub에서는 소스코드의 변화뿐 아니라 프로젝트관리, 개발자간의 의사소통 등을 지원한다. GitHub는 오픈소스 프로젝트나 소규모 프로젝트의 경우 무료로도 사용이 가능하다.

예를 들어 〈그림 2〉는 GitHub를 이용하는 프로젝트의 한 예로 개발자가 얼마나 활발하게 소스 코드를 변경하고 있는지를 보여주고 있다. 소프트웨어 코드는 라인 별로 하나의 작업을 수행하기 때문에 개발자가 얼마나 많은 수의 라인에 변화를 주었느냐를 측정하면 얼마나 개발이 활발하게 진행되고 있는 지를 추정할 수 있게 되는데 이러한 변화량을 날자 별로 수집하여 그래프로 그리면 프로젝트의 진행이 전반적으로 얼마나 활발한 지를 유용하게 추정하는 정보가 된다.

또한 일주일 중, 혹은 하루 중 어떤 시간에 가장 높은 생산성이 도출 되는지도 프로젝트 관리측면에서 중요한데, 회의 등 부가적인 작업은 가능한 생산성이 높은 시간을 피해서 수행하는 것이 효과적이기 때문이다. GitHub는 〈그림 3〉과 같이 전반적으로 개발자들이 언제 많은 양의 프로그램 코드라인을 수정하는지를 한눈에 파악할 수 있도록 지원한다.

(10) GitHub. <https://github.com>. [Online; accessed 14-Dec-2015].

〈그림 3〉 Efficiency

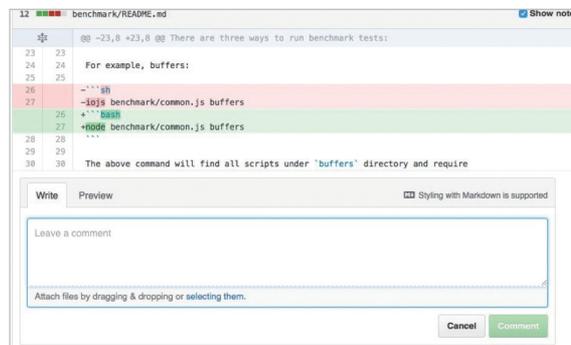


소프트웨어는 고도의 논리적인 사고를 반영하여 작성된다. 이런 이유로 소프트웨어 개발자들이 소프트웨어 개발과정에서 문의사항이 발생하면 이 문제는 대단히 논리적인 이해가 필요하고, 이것을 명확하게 기술하는 것은 매우 중요한 일이다. 따라서 소프트웨어 개발자들이 의견을 교환하는 경우에는 일상적인 언어보다는 고도의 논리적인 사고를 표현하는 도구인 프로그램코드를 포함하는 경우가 많은데, GitHub를 이용하면 이런 과정을 좀더 효과적으로 수행할 수 있게 된다. 예를 들어 GitHub를 이용하여 소스코드를 관리하는 경우 소프트웨어 개발자가 다른 사람이 작성한 코드에 의문이 있는 경우 〈그림 4〉와 같이 프로그램 코드에 직접 문의 사항을 기록하여 공유할 수 있다. 이렇게 하면 프로그램 코드를 복사하는데 소요되는 노력이 줄어들게 되고, 더 나아가서는 복사 과정에서 발생할 수 있는 오류도 원천적으로 제거된다.

소프트웨어 개발과정에서는 의사소통이 중요하다. 소프트웨어 개발의 후반부에는 소프트웨어 개발자간의 의사소통의 중요성과 빈도가 높지만, 초기에는 소프트웨어 개발자와 마케팅 등 타 부서와의 의사소통이 차지하는 비중이 높다. 전통적으로 이 작업은 대면회의, 전화, 이메일 등으로 진행되었지만 최근에는 Slack<sup>(11)</sup>이라는 클라우드 시스템의 활용도가 높아지는 추세이다.

Slack은 기관별로 독립된 서버를 구성하여 사용하는 일종의 채팅서비스이다. 이 시스템이 소프트웨어 개발에서 유용한 이유는 업무와 관련하여 협의해야하는 사람들을 파악하고 그룹을 구성하여 정보를 공유하는 것이 편리하기 때문이다. 이것은 기업이 구성원들에게 전화를 제공할 때 PBX 시스템의 형태로 제공하는 것과 유사한데, Slack에서는 사용자들을 쉽게 찾아 연락할 수 있는 기능과 부재 시 메시지를 남기는 기능, 그리고 확장기능을 설치할 수 있는 방안 등을 제공한다.

〈그림 4〉 Notes on source code



(11) Skype. <http://www.skype.com>. [Online; accessed 2015-12-14].

Slack의 communication 기능과 전문가 시스템 기술을 결합하면 매우 흥미로운 활용이 가능하다. 예를 들어 인력이 부족한 소규모 벤처에서는 전문가시스템을 이용하여 질의응답이나 소프트웨어 컴파일 등의 간단한 작업을 컴퓨터가 대행하도록 할 수도 있다. 이와 같은 방식은 실리콘밸리의 첨단기업에서는 이미 일상화되어 예를 들어 Hubot이라는 이름의 표준 인터페이스모듈도 오픈소스로 개방되어 개발되어있다.

Hubot은 Coffeescript라는 확장성이 높은 프로그래밍언어로 개발되어 있는데, 소프트웨어를 컴파일한 후 기초적인 테스트를 수행하는 작업을 담당하는 가상의 직원으로 활용할 수 있다. 이 경우 소프트웨어개발자는 chatting 프로그램에 다음과 같은 메시지를 Hubot에게 보냄으로써 빌드 업무를 요청하고 그 결과를 통보받을 수 있다.

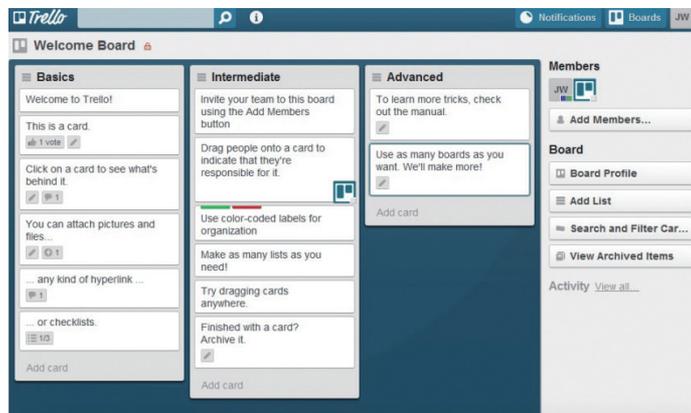
〈표 1〉 Hubot 커뮤니케이션(예)

나 : @Hubot 도와줘 Hubot : 뭘요?  
 나 : @Hubot 현재버전릴리즈  
 Hubot : 에러났네요. :P redmine #115292

소프트웨어 개발 과정에서는 고객이나 외부의 개발자와 개발화면을 공유하면서 실시간으로 의사소통을 할 필요도 있는데, 전통적으로는 고가의 전용장비를 구매해야하는 투자가 많이 소모되는 부분이였다. 또한 시스템의 운영에도 많은 비용이 소모되었다. 그러나 오늘날에는 클라우드 기반으로 많은 서비스들이 매우 저렴하게 제공되고 있다. 한 예로 Skype<sup>(12)</sup>는 음성통신만을 지원하는 인터넷 전화로 출발하였으나 현재는 비디오 통신기능뿐 아니라 화면 공유를 지원하여 외부의 상황을 좀더 정확히 교환할 수 있게 되었다.

일반적으로 소프트웨어 개발과 관련한 현황은 Project Manager (PM)가 수집하여 공유하게 된다. PM은 시기별로 해야할 일들을 계획하고 진행상황을 점검한 후에 계획대비 차이가 발생하는 부분을 파악하여 적절한 행동을 취하는 역할을 담당하는데, 이 과정에서 소프트웨어 개발에 참여하는 수 많은 이해 당사자들, 소프트웨어개발자, 테스트 담당자등과 대단히 많은 양의 정보를 적시에 공유해야 하는 어려움이 있다.

〈그림 5〉 Trello



(12) Skype. <http://www.skype.com>. [Online; accessed 2015-12-14].



소프트웨어 산업의 역사를 돌이켜 보면 프로젝트 관리에 필요한 정보를 수집하고 관리하는데 들어가는 노력을 줄이기 위해서 많은 수의 프로젝트관리 도구들이 제안되어 개발되었다. 그러나 이런 도구들은 대체로 매우 큰 규모의 프로젝트를 위해 개발된 배경을 갖고 있고, 가격도 고가인 관계로 초기 스타트업이나 규모가 작은 소프트웨어 개발업체에서 활용하는 예는 흔치 않았다. 그러나 최근에는 클라우드 기반으로 프로젝트를 관리하는 시스템이 제안되어 많은 기업이 이용하기 시작하고 있다.

그 중 Trello 라는 클라우드 기반 시스템은 실제로 많은 기업에서 이용되고 있는데, 이 시스템은 <그림 5> 과 같이 커다란 보드에 카드를 붙이는 것처럼 정보를 작성할 수 있도록 지원하는데, 인터페이스로는 웹을 사용한다. 이 시스템이 소규모 소프트웨어 개발사에서 유용하게 사용되는 이유는 프로젝트 관리자가 이 보드를 웹에 공개하는 것이 간편하고, 일단 공개되면 누구나 어디서나 접근하여 정보를 공유할 수 있고, 나아가서 권한이 있는 사람들은 참조하거나 수정할 수 있도록 할 수 있기 때문이다. 이 와 같이 누구나 참조할 수 있고 필요한 경우 수정할 수 있도록 정보를 공유하는 방식을 kanban 방식이라고 하는데, 이 방식을 사용하는 경우 PM 은 정보가 최신 내용을 담고 있도록 하는데 더 많은 노력을 기울이고 현황을 공유하는데 들어가는 노력은 상대적으로 많이 절감할 수 있어서 전체적으로는 PM의 업무 생산성이 높아지게 된다.

소프트웨어 개발의 많은 부분이 클라우드로 전환되고 있지만, 실시간으로 소스 코드를 편집하고 관련 정보를 참조해야 하는 코딩부분은 클라우드로 전환하기에는 응답 속도 등 많은 도전 과제가 있었다. 그러나 컴퓨터의 용량 처리속도 뿐 아니라 인터넷의 연결 속도가 증가한데다 web browser의 성능까지 증가하면서 소스 코드 편집과 디버깅과 같은 소프트웨어 개발에서 가장 빈번하게 이루어지는 작업을 지원하는 클라우드 서비스도 출현하게 되었다.

클라우드 9<sup>(13)</sup>는 웹 브라우저를 통해서 원격지에 설치된 가상컴퓨터에 소스 코드를 작성하고 이를 컴파일한 후 실행할 수 있는 환경을 제공한다. 이 시스템을 사용할 때, 사용자는 웹 브라우저만 있으면 된다. 사용자는 웹 브라우저를 이용하여 클라우드 9의 홈페이지에 접근하고, 적절한 방식으로 로그인하면 웹 브라우저 화면에는 클라우드 9가 제공한 파일 목록과 편집창 등 프로그램 작성에 필요한 정보가 나타난다. 이 시스템에서 제공하는 서버는 Amazon Web Service에 설치되어 있는데, 사용자는 자신의 컴퓨터에서 ssh로 독자적인 연결을 맺을 수도 있다.

클라우드 9에서 제공하는 서버는 Linux 서버의 일종으로 사용자가 <그림 6>과 같이 직접명령을 입력할 수 있도록 지원한다. 여기서 사용자는 주어진 컴퓨터에 대해서 제한 없이 명령을 실행할 수 있는 권한을 갖게 되는데, 이 권한은 새로운 소프트웨어시스템을 설치하거나 네트워크 설정을 변경하는데 사용되고 명령창에서 직접 입력하여 사용할 수 있도록 개발되어 있다.

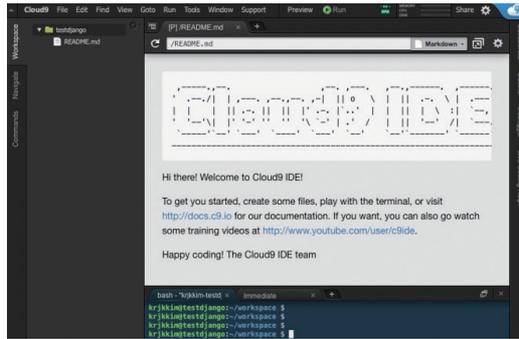
<그림 6> Compilation

```

bash - "krjkkim-testd" x Immediate x +
krjkkim@testdjangoo:~/workspace $
krjkkim@testdjangoo:~/workspace $
krjkkim@testdjangoo:~/workspace $ ls
krjkkim@testdjangoo:~/workspace $ README.md
krjkkim@testdjangoo:~/workspace $ vi a.c
krjkkim@testdjangoo:~/workspace $ vi a.c
krjkkim@testdjangoo:~/workspace $
krjkkim@testdjangoo:~/workspace $ vi a.c
krjkkim@testdjangoo:~/workspace $ gcc a.c
krjkkim@testdjangoo:~/workspace $ ./a.out
Welcome
krjkkim@testdjangoo:~/workspace $ █
    
```

(13) 클라우드 9. <https://c9.io/>. [Online; accessed 2015-12-14].

〈그림 7〉 Command line



이 시스템의 기본 인터페이스는 웹 브라우저에 기반한 일종의 터미널 에뮬레이터이다. 따라서 편집된 파일을 명령창에서 Linux 명령어를 통해 확인할 수 있을 뿐 아니라 해당시스템에 설치된 컴파일러를 〈그림 7〉와 같이 직접 호출하여 실행파일을 작성할 수도 있다.

이 시스템에서 제공하는 터미널 에뮬레이터는 매우 수준이 높아서 전용 에뮬레이터의 대응으로 사용해도 무방할 정도의 수준이다. 예를 들어 클라우드 9 의 한 서버에 접근한 후 그 안에서 〈그림 8〉과 같이 vi 나 emacs와 같은 편집기를 구동시킨 경우에도 큰 문제없이 소프트웨어 작성을 수행할 수 있다.

〈그림 8〉 Separate editor



오늘날 위에서 설명한 내용은 대부분 클라우드 환경에서 운영하는 것이 가능하다. 각 단계별로 사용되는 도구와 그 때 기대되는 산출물, 이를 관리하기 위한 간단한 문서화 규칙, 그리고 각 단계에서 효과적으로 사용될 수 있는 communication 및 automation 도구들을 정리하면 다음 〈그림 9〉과 같다.

〈그림 9〉 개발 프로세스와 개발 환경

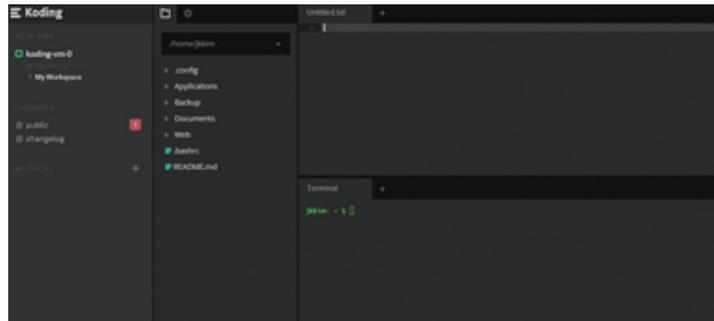
단계	산출물	documentation	communication	automation
work item (requirement)	title, description, validation list	trelo, excel, google docs	slack, messenger, email	hubot, redmine
Refine (design)	data model, i/p/o, verification list	trelo, redmine	slack, messenger, (email)	hubot, github, redmine
Coding (implementation)	Source code, build script, test data	trelo, redmine, github	slack, redmine	hubot, github, jenkins
Review (testing)	checklist (verification, validation), test script	trelo, redmine, slack, github, jenkins	redmine, slack, messenger	hubot, jenkins
Release (QA)	performance test, official approval	github (branch), slack, trelo	email, slack, messenger	hubot, jenkins

### 3. 클라우드 기반 소프트웨어 개발 환경 분석

소프트웨어 개발자는 소프트웨어 개발의 모든 단계에서 중요한 역할을 한다. 그러나 소프트웨어 개발자의 역할이 가장 중요하고 대체 불가능한 지점은 설계된 내용을 실행 가능한 코드로 전환하는 구현 (implementation) 단계이다. 따라서 지식 근로자로서 소프트웨어 개발자의 요구를 반영하여 원격 근무와 같은 소프트웨어 개발자에게 친화적인 환경을 구축하는 데는 소스 코드의 편집, 디버깅 등을 지원하는 클라우드 기반 개발환경의 역할이 매우 중요하다. 소규모 창업 벤처의 요구사항을 중심으로 클라우드 개발환경이 제공하는 기능과 품질을 개략적으로 평가하고, 전문적인 소프트웨어 개발에 활용할 수 있는지의 여부를 파악하기 위해서 평가항목에 적용될 내용을 파악하고 이를 기초로 현재 시장에서 활용 가능한 클라우드 개발환경의 특성을 분류한다.

클라우드 개발환경은 사용자에게 가상 컴퓨터를 할당하여 <그림 10>과 같은 환경을 웹브라우저로 접근하여 사용할 수 있도록 지원하는 것이다. 이 때 가상 컴퓨터의 성능은 어느 정도인지, 확장이 가능한지, 충분히 커스터마이징하여 활용할 수 있는지, 제공하는 컴파일러에 제약은 없는지, 편집기가 충분히 편리하게 활용할 만 한지, 그리고 원격지에서 실행중인 프로그램의 오류를 찾고 수정하는 디버깅 과정이 충분히 편리한지의 여부가 중요한 판단 여부가 된다.

<그림 10> 온라인 개발 환경



#### (1) 클라우드 개발 환경에 대한 평가항목

클라우드 개발환경은 다양한 서비스 제공자에 의해서 제공되고 있으나 사용 목적과 지원 방식에서는 약간씩 차이를 보이고 있다. 초창기에는 사업화보다는 연구목적으로 개발되었기 때문에 거의 대부분 오픈 소스로 제공되었으나 최근에는 제품의 완성도가 증가함에 따라 소스 코드는 비공개인 경우도 많다. 클라우드 개발환경에 대한 관심이 최근 증가하고는 있지만 아직 독자적인 시장으로 성숙한 상황은 아니기 때문에 Gartner<sup>(14)</sup>나 IDC<sup>(15)</sup>와 같은 시장조사기관에서도 거의 언급되지 않고 있다. 따라서 제품이 제공하는 기능이나 한계점 등은 사용자의 입장에서 직접 경험하면서 판단할 수밖에 없는 것이 현실이다.

클라우드 개발 환경으로 개발 가능한 영역을 조사한 결과 교육용 App 에서 상용 웹 서비스까지 다양하였다. 이 중 교육과 웹 서비스에 대한 개발자와 IT 전문가들의 관심이 높은 점을 반영하여 현재 서비스나 오픈 소스 형태로 제공되는 클라우드 개발환경을 선정하였다. 현재 활발한 연구가 이루어지고 있다는 점과

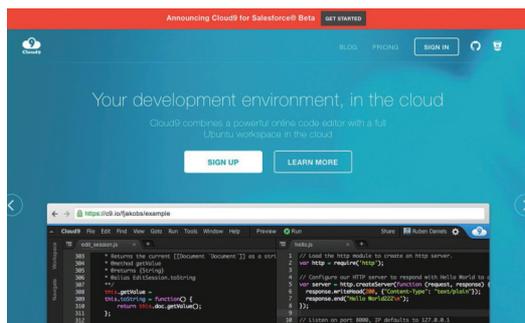
(14)artner. Gartner. [Online; accessed 2015-12-14].

(15)Idc. <https://www.idc.com>. [Online; accessed 2015-12-14].

오픈소스로 제공되어 변종이 출현할 수 있다는 점을 감안하면 향후 새로운 서비스나 제품이 출현할 가능성은 높다. 그러나 현시점에서 활용가능한 서비스를 모아 목록을 작성하고 이들에 대한 개괄적인 평가를 수행함으로써 향후 클라우드 개발 환경을 좀더 깊이 있게 분석하고 비교하는 토대를 마련하는 것에도 의미가 있다. 본 연구에서는 다음과 같은 관점에서 클라우드 개발 환경을 평가하였다.

- 클라우드 features : 제공되는 서버의 성능이나 확장성이 충분한가?
  - on-demand : 프로세스나 메모리, 디스크와 같은 서버관련자원을 추가 요청하여 할당 받을 수 있는가?
  - instant-access : 새로 프로젝트나 테스트 환경을 구축하고자 할 때, 온라인으로 손쉽게 서버를 요청하여 활용할 수 있는가?
  - real-time response : 편집이나 명령어 실행의 결과를 실시간으로 받아볼 수 있는가?
  - collaborative : 다수의 사용자가 할당된 서버에서 공동으로 작업할 수 있는가?
- Supported domains : 서버, 컴파일러, 편집기 등 제공되는 개발환경이 지원하는 소프트웨어의 종류는 어떤 것인가?
  - Mobile apps : 모바일 소프트웨어의 개발을 지원하는가?
  - Web 서버 apps : 웹 기반 서버프로그램의 개발을 지원하는가?
- Open system : 개방형으로 구성되어 세부사항을 확인하여 오류에 대처할 수 있는가?  
새로운 기능의 추가가 자유로운가?
- Usability : 사용하기에 편리한가? 예를 들어 drag and drop 과 같이 전통적인 웹 응용 프로그램에서 지원되지 않던 부분도 프로그래머에게 불편을 주지 않고 매끄럽게 활용할 수 있도록 하는가?
- Database support : 서버에 데이터베이스를 설치하고 관리할 수 있는가?
- Mashable : 외부 프로그램이나 서비스와 연계하여 활용이 가능한가?
- Language support : 지원하는 프로그래밍 언어에 제약은 없는가? Java, C 등 널리 사용되는 프로그래밍 언어가 잘 지원되는가?
- Extensible to professional development : 상용 서비스를 운영하는데 제약은 없는가?  
예를 들어 서버 관리에 필요한 ssh 계정을 운영하는데 제약을 두지는 않는가?

〈그림 11〉 클라우드 9



- Support : 서비스 제공자가 고객에게 충분하고 신속한 서비스를 제공하는가?
- Security/Availability : 상용 서비스 제공에 필요한 네트워크 및 보안관련 서비스들이 제공되는가?  
예를 들어 DoS 공격의 위협에 대처할 수 있고 사용자가 폭증할 때 서버를 증설하여 load balancing 을 운영할 수 있는 방안을 제공하는가?

## (2) 클라우드 개발 환경의 개괄적 평가

이 절에서는 클라우드 개발환경에 대한 사용경험을 토대로 앞에서 언급한 평가항목들을 고려하여 클라우드 개발 환경에 대한 평가를 진행한다. 평가의 결과를 사업성, 기술의 비전, 그리고 소스 공개 여부로 정리한다.

- 클라우드 9 : 사용자가 다수의 프로젝트를 개설하고 운영할 수 있도록 지원한다. <그림 11>과 같이 웹에서 클라우드 개발 환경과 운영환경을 시험할 수 있다. 특이하게도 하나의 사용자에게 하나의 가상 machine 을 할당하는 방식이 아니라 하나의 워크스페이스(프로젝트)에 하나의 가상머신을 할당한다. 따라서 다른 프로젝트에서 개발하는 시스템과 네트워크 혹은 데이터베이스 문제로 충돌이 발생하는 것을 원천적으로 봉쇄하고 있다.
  - 클라우드 개발 환경 제공<sup>(16)</sup>
  - 소스 코드는 초기에는 공개되었으나 현재는 공개하지 않음.  
초기버전의 경우 deprecated 된 상태이지만 github에서 history 형태로 확인할 수 있다.<sup>(17)</sup>

<그림 12> Codeenvy



- Codenvy : 클라우드 서버를 제공하여 인터넷에서 상용서비스를 제공할 수 있도록 지원한다. <그림 12>과 같이 웹에서 클라우드 개발환경을 시험할 수도 있다. php, nodejs 등 서버 응용프로그램을 작성할 때 필요한 기본적인 기능과 database 연결 기능 등을 갖추고 있다. 포트포워딩을 통해서 클라우드 서버에 접근할 수 있기 때문에 지속적으로 서버의 상황을 모니터링할 수 있어 상용서비스의 운영도 가능하다.
  - 클라우드 개발 환경 제공<sup>(18)</sup>
  - 소스 코드 공개<sup>(19)</sup>

(16) 클라우드 9. <https://c9.io/>. [Online; accessed 2015-12-14].

(17) 클라우드 9 source code. <https://github.com/ajaxorg/클라우드9>. [Online; accessed 2015-12-14].

(18) Codenvy. <https://codenvy.com/>. [Online; accessed 2015-12-14].

(19) Codenvy source code. <https://codenvy.com/products/che>. [Online; accessed 2015-12-14].

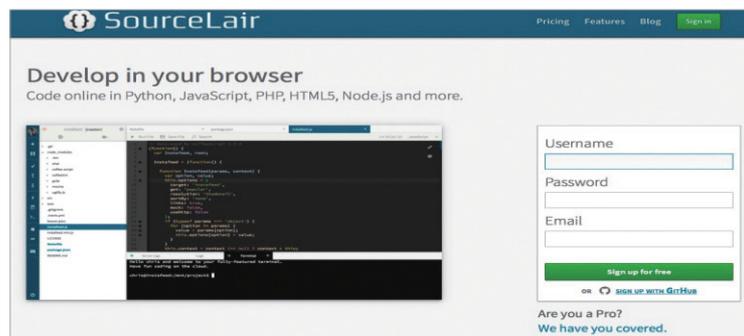
- Codeanywhere : 클라우드서버를 제공하여 인터넷에서 상용 서비스를 제공할 수 있도록 지원한다. <그림 13>과 같이 웹에서 클라우드개발환경을 시험할 수도 있다. 편집기 등은 깔끔하게 정리되어 있지만 데이터 베이스 연결은 프로그램코드를 직접 작성해야 하고 운영 관련한 도구가 부족한 편이다.
  - 클라우드 개발 환경 제공<sup>(20)</sup>
  - 소스 코드 공개<sup>(21)</sup>

<그림 13> Codeanywhere



- SourceLair : 간단한 웹 응용프로그램을 작성하여 사이트를 운영하는 것을 목적으로 php, nodejs, html5 등을 지원한다. 데이터 베이스 연결은 가능하지만 무료버전에서는 30일 한정판으로 사용해야 하는 등의 제약이 있다. <그림 14>와 같이 웹에서 클라우드 개발환경을 시험할 수도 있다. 소스 코드 공개는 언급이 없다.
  - 클라우드 개발 환경 제공<sup>(22)</sup>

<그림 14> SourceLair



- Koding : 역시 교육에 적용할 목적으로 개발된 서비스로 <그림 15>과같이 웹에서 클라우드 개발 환경을 시험할 수 있다. 소스 코드는 공개하지 않고 있으며 사용자당 하나의 가상 서버를 제공한다.
  - 클라우드 개발 환경 제공<sup>(23)</sup>

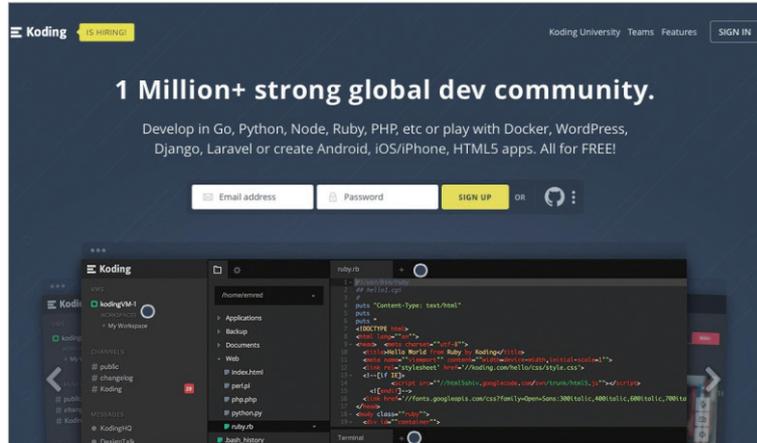
(20) Codeanywehre. <https://codeanywhere.com/>. [Online; accessed 2015-12-14].

(21) Codeanywheresourcecode.<https://code.google.com/p/codeanywhere/source/checkout>. [Online; accessed 2015-12-14].

(22) Sourcelair. <https://www.sourcelair.com/home>. [Online; accessed 2015-12-14].

(23) Koding. <https://koding.com/>. [Online; accessed 2015-12-14].

〈그림 15〉 Koding



- Nitrous : 클라우드 개발 환경에 중점을 둔 환경으로 〈그림 16〉과 같이 시험 환경을 제공한다. 그렇지만 신용카드 정보를 넣기 전에는 시스템을 활성화 시키지 않는 등 운영 면에서는 부족한 점이 보인다.
  - 클라우드 개발 환경 제공<sup>(24)</sup>

〈그림 16〉 Nitrous



- AppInventor : MIT 에서 mobile application 을 초보자도 쉽게 작성할 수 있도록 지원하기 위해서 제작한 시스템이다. 〈그림 17〉와 같이 웹 을 통해서 접근할 수 있다. 사용자에게 독자적인 서버를 제공하지 않고 교육용으로 설계된 특수한 언어로 소프트웨어를 구현하도록 되어 있어 상업용 서비스의 제작에는 부족한 점이 있다. Android 응용프로그램만 작성이 가능하다.
  - 클라우드 개발 환경 제공<sup>(25)</sup>
  - 소스 코드 공개됨<sup>(26)</sup>

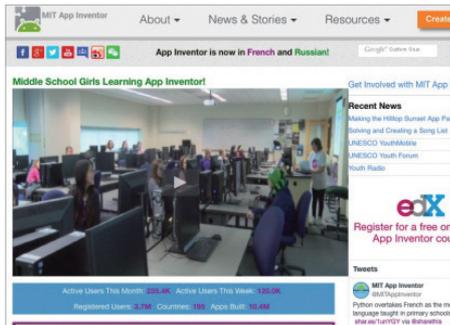
(24) Nitrous. <https://www.nitrous.io/>. [Online; accessed 2015-12-14].

(25) Appinventor. <http://appinventor.mit.edu/explore/>. [Online; accessed 2015-12-16].

(26) Appinventor source code. <http://appinventor.mit.edu/appinventor-sources/>. [Online; accessed 2015-12-14].

- Scratch : 프로그래밍 개념을 이해하는데 도움을 주도록 MIT 에서 개발한 사이트. <그림 18>에서와 같이 웹을 통해서 접근할 수 있다. 클라우드 서버를 제공하지 않고 교육용으로 작성된 특수한 언어로 소프트웨어를 구현하도록 되어 있어 상업용 서비스를 개발하는 데는 부족한 점이 있다.
  - 클라우드 개발 환경 제공<sup>(27)</sup>
  - 소스 코드 공개됨<sup>(28)</sup>

<그림 17> AppInventor

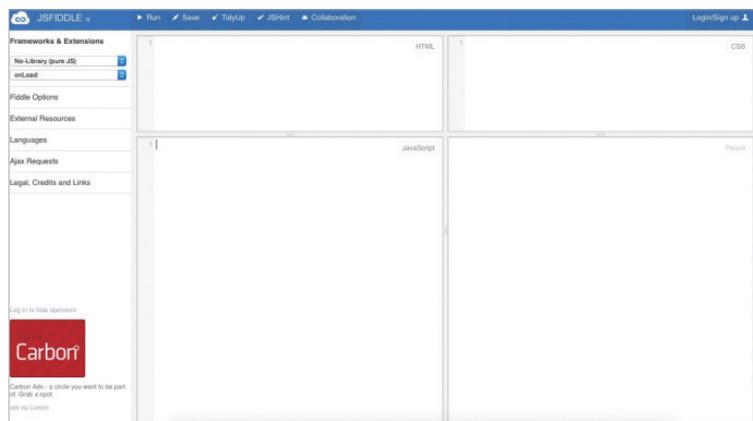


<그림 18> Scratch



- JSFiddle : JavaScript를 학습하고 간단한 서비스를 구축하는 것을 목표로 한다면 매우 훌륭한 환경이다. 가상 서버가 할당되는 것은 아니지만 운영환경을 JavaScript로 한정하였기 때문에 하나의 서버에서 여러 개의 응용 프로그램을 동작시키는 등 저비용으로 운영하고 있는 것으로 판단된다. <그림 19>과 같이 웹에서 클라우드 개발 환경을 시험할 수 있는데, HTML, CSS 등도 함께 편집할 수 있도록 지원하고 있다. JavaScript 프로그래밍언어는 CoffeeScript 등 여러 가지 변종이 존재하고, 지원하는 application framework의 종류도 많은데 이 시스템은 상당히 폭넓은 분야의 변종 언어와 application framework을 지원하고 있다. 개인이나 소규모 회사의 홈페이지를 작성하고 관리하는 경우, 작은 기능을 담당하는 web component 를 작성하는데 유용할 것으로 판단된다.
  - 클라우드 개발 환경 제공<sup>(29)</sup>

<그림 19> JSFiddle



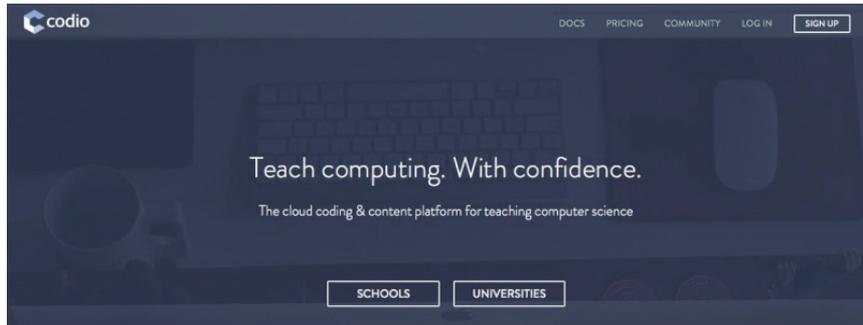
(27) Scratch. <https://scratch.mit.edu/>. [Online; accessed 2015-12-14].

(28) Scratch. [http://wiki.scratch.mit.edu/wiki/Scratch\\_1.4\\_Source\\_Code](http://wiki.scratch.mit.edu/wiki/Scratch_1.4_Source_Code). [Online; accessed 2015-12-14].

(29) JsFiddle. <https://jsfiddle.net/>. [Online; accessed 2015-12-14].

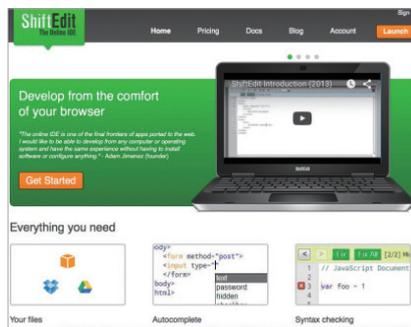
- Codio : 소프트웨어 교육을 목적으로 개발된 서비스로 <그림 20>과 같이 웹에서 개발환경을 시험할 수 있다. 시험기간이 14일로 제한되고 일종의 가상 교실 환경에서만 사용할 수 있는 등의 제약이 있어서 본격적인 서비스나 제품의 개발에 적용되기에는 제약이 있다.
  - 클라우드 개발 환경 제공<sup>(30)</sup>

&lt;그림 20&gt; Codio

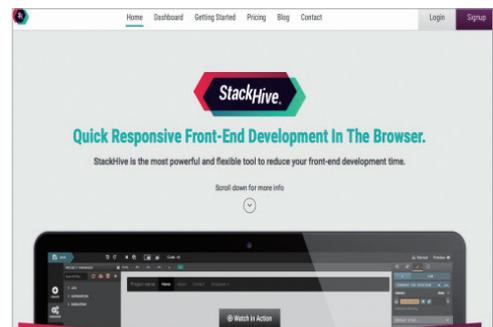


- ShiftEdit : 주로 웹 사이트 개발을 염두에 두고 개발된 시스템으로 <그림 21>과 같이 웹에서 클라우드 개발 환경을 시험해볼 수 있다. 매우 다양한 프로그래밍 환경이 제공되나 서버 운영과 관련한 부분의 지원이 부족하고 시험 사용 기간도 30일로 제한되어 있어서 서비스 운영에 적용한다는 측면에서는 제약이 크다고 판단된다.
  - 클라우드 개발 환경 제공<sup>(31)</sup>
- StackHive : 다양한 웹 클라이언트에 접근하는데 필요한 기능과 홈페이지 개발에 필요한 다양한 템플릿을 제공하는 것을 목적으로 하는 서비스 <그림 22>와 같이 웹에서 클라우드 개발 환경을 시험해볼 수 있는데, 서버 관련된 기능은 거의 없어서 주로 JavaScript를 활용하여 클라이언트를 개발하는데 사용하면 적합할 것으로 판단된다.
  - 클라우드 개발 환경 제공<sup>(32)</sup>

&lt;그림 21&gt; ShiftEdit



&lt;그림 22&gt; StackHive

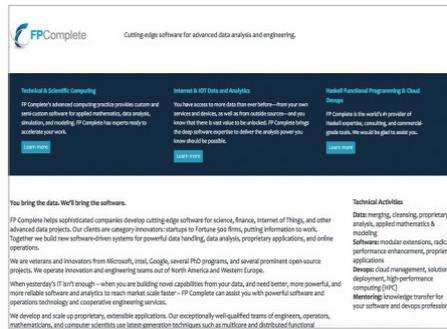
(30) Codio. <https://codio.com/>. [Online; accessed 2015-12-14].(31) Shiftedit. <https://shiftdit.net/>. [Online; accessed 2015-12-14].(32) Stackhive. <http://stackhive.com/>. [Online; accessed 2015-12-14].

- FP Complete : 함수형 언어의 교육을 목적으로 개발된 시스템으로 (그림 23)과 같이 웹에서 클라우드 개발 환경을 시험해볼 수 있다. 대중적인 시스템의 개발보다는 과학 기술이나 금융 등의 개발에 특화되어 있음에도 클라우드 개발환경을 제공하고 있고, 직관적이고 편리한 인터페이스를 제공하고 있다. 다만 가입 절차가 복잡하고, 사용자층이 넓지 않아 서비스가 지속될 수 있을지는 의문이 든다.

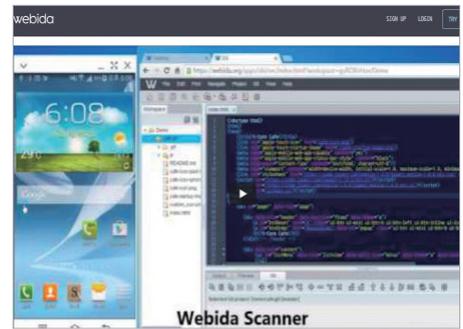
- 클라우드 개발 환경 제공<sup>(33)</sup>
- 소스 코드 공개됨<sup>(34)</sup>

- Webida : Tizen를 등 HTML5 기반의 mobile application을 개발하기 위한 개발 환경으로 개발되었다. 개발 환경 자체도 JavaScript 로 개발되어 있고 브라우저 호환성도 제공하기 때문에 일반적인 웹 응용 시스템을 개발하는데도 충분히 활용할 수 있다.

<그림 23> FP Complete



<그림 24> Webida



- Webida는 (그림 24)과 같이 웹에서 클라우드 개발 환경을 시험해볼 수 있는데 Code completion의 수준이 매우 높고 소스 관리도 편리하게 수행할 수 있다. 다만 아직 개발 중인 시스템이라 상용 서비스에 필요한 서버 설정이나 네트워크 관리 등의 기능은 제공하지 않고, 전용 클라우드 서버를 할당하고 관리하는 작업은 사용자가 수작업으로 진행해야 하는 등의 한계가 있지만 본격적인 서비스가 시작되면 개선될 수 있을 것으로 판단된다.

- 클라우드 개발 환경 제공<sup>(35)</sup>
- 소스 코드 공개됨<sup>(36)</sup>

- Goorm : 교육을 목적으로 개발된 클라우드 기반 프로그래밍 환경으로 (그림 25)와 같이 클라우드 환경에서 접근하여 활용할 수 있다. 클라우드 개발 환경 부분은 많은 부분 완성되었고 가상 교실 운영 등의 시스템은 아직 개발 중이다. 사용자마다 가상 서버를 제공하기 때문에 상용 서비스로 활용하는 것도 가능하다. 다만 교육에 중점을 두고 있기 때문에 계정당 하나씩의 가상머신이 할당될 뿐이고, 초기 사용자가 루트로 설정되어 보안상 문제가 발생할 여지가 있지만 교육용으로 사용하기에는 충분하다.

- 클라우드 개발 환경 제공<sup>(37)</sup>

(33) Fp complete. <https://www.fpcomplete.com/>. [Online; accessed 2015-12-14].

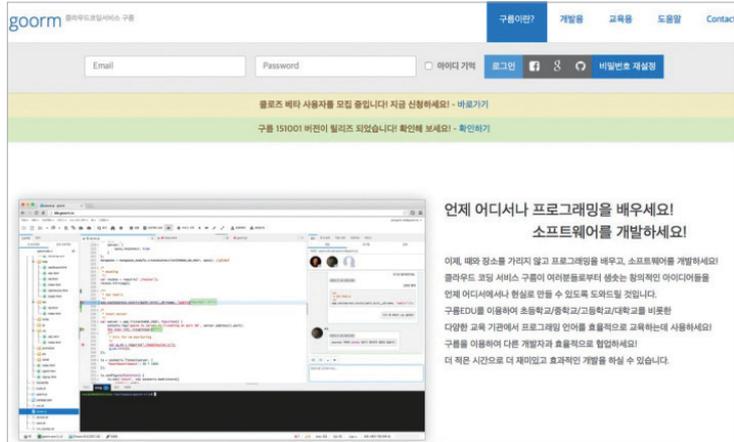
(34) Fp complete. <https://github.com/fpco/ide-backend>. [Online; accessed 2015-12-14].

(35) Webida. <https://webida.org/apps/site/>. [Online; accessed 2015-12-14].

(36) Webida source code. <https://github.com/webida>. [Online; accessed 2015-12-14].

(37) Goorm. <http://goorm.io/>. [Online; accessed 2015-12-14].

〈그림 25〉 Goorm



- Vizwik : 모바일 응용프로그램을 작성하는 경험을 통해서 프로그래밍 교육을 진행하겠다는 것을 목적으로 개발된 시스템으로 관련 분야를 장기간 연구한 연구 그룹이 운영하고 있는 사이트이다. 〈그림 26〉과 같이 웹으로 클라우드 개발 환경에 접근할 수 있다. 상용 서비스보다는 교육에 중점을 두고 있다.
  - 클라우드 개발 환경 제공<sup>(38)</sup>

〈그림 26〉 Vizwik



중소 규모의 개발자가 클라우드 개발 환경을 선택할 때 고려할 것은 개발 환경의 편의성 뿐 아니라 서비스 사용자로서 기업의 연속성이나 장래성도 고려하여야 한다. 소프트웨어 개발 환경은 지속적으로 변화하고 있는데, 이런 변화를 잘 이끌어갈 기술적인 비전이 있는지, 그 비전을 수행할 사업적인 역량이 있는지를 함께 고려하지 않으면 새로운 변화에 적응하지 못하는 기업이 되거나 서비스를 운영하는데 어려움을 겪는 상황에 처할 수 있기 때문이다.

여기에 한 가지 더 고려할 사항은 시스템에서 발생하는 문제를 정확하게 파악하고 신속히 개선할 여건이 갖추어져 있어야 한다는 것이다. 소프트웨어 개발의 역사를 통해서 소스 코드에 접근하고 이를 개선할 수 있다는 것은 소프트웨어의 생명을 장기적으로 유지하고 지속적인 개선을 통해 효용성을 증대시키는데 매우 효과적인 정책임은 아주 잘 검증되었다. 이런 이유로 오늘날 리눅스를 비롯한 대다수의 소프트웨어 프로젝트

(38) Vizwik. <https://vizwik.com/>. [Online; accessed 2015-12-14].



트는 오픈 소스 형태로 진행되고 있다. 클라우드 개발환경을 소프트웨어 개발의 최첨단인 상업용 서비스나 제품을 개발하는데 적용하기 위해서는 오픈 소스로 개발환경 시스템의 개발을 진행하는 것이 바람직하다. 이런 이유로 많은 수의 기업들이 실제로 오픈 소스로 클라우드 개발환경 시스템을 개발하고 있기도 하다.

그러나 클라우드 개발 환경을 서비스 형태로 운영하는 사업자의 입장에서 보면 소프트웨어 코드를 공개하면 사업측면에서 차별화 요소가 대부분 사라질 수 있다는 문제가 있다. 최악의 경우 우리가 공개한 소스 코드로 경쟁 서비스가 출현할 수도 있는 것이다. 이런 이유로 초창기 오픈 소스 프로젝트로 시작했던 많은 수의 클라우드 개발 환경 서비스가 현재는 소스를 공개하지 않고 있다. 예를 들어 클라우드 9, Goorm 등의 서비스는 오픈 소스라는 점을 강조하며 서비스를 시작했지만 현재 정책을 변경하여 소스 코드에 접근이 불가능하다.

현재 서비스 중인 클라우드 시스템들이 사업적으로 얼마나 많은 사용자를 모집하였는지, 매출이 어느 정도 인지는 아직 대부분의 기업이 사업 초기 단계이기 때문에 공식 등을 통해서 정량적으로 판단하는 것은 불가능하다. 그러나 전문지의 기사나 블로그 등 인터넷에서 언급되는 정도 등 간접적인 방법으로 파악해보면 사업적으로 성공적인 기업들이 여럿 있다고 판단된다. 이중 클라우드 9 등 몇몇 기업은 투자 측면에서 의미 있는 성과를 거둔 것으로 뉴스를 통해서 추정할 수 있었다.

클라우드 기업들의 성과와 역량을 판단하기 위해서 본 연구에서는 사업적인 측면과 기술적인 측면을 동시에 고려하여 평가한다. 이 평가 방법은 Gartner<sup>(14)</sup>에서 제시한 Magic Quadrant 방식과 유사한데, Gartner는 새로운 산업계에 참여한 신생기업의 경쟁력을 분류하는데 이 방법을 사용하고 있다. Gartner의 분류법은 크게 두 축으로 구성되는데, 먼저 새로운 산업 분야에 대한 분석인 만큼 기술적인 창의성과 진보성을 하나의 축으로 하고, 기업으로서의 실행력과 자금력 등을 다른 하나의 축으로 한다. 이 두 측면이 모두 성공적인 경우 leader group에 소속이 되고, 기술적인 측면에 부족한 점이 있는 경우 challenger로, 실행력에 부족한 점이 있는 경우 visionary로 분류한다. 그 외에 해당기술을 특정한 분야나 시장에 적용시켜 생존하는 기업들은 niche player로 구분한다.

클라우드 개발 환경의 측면에서 보면 얼마나 좋은 기술을 얼마나 효과적으로 시장에 보급하고 있는가라는 측면만큼이나 해당 서비스를 제공하는 소프트웨어 프로젝트가 얼마나 지속될 수 있는가 하는 문제도 매우 중요한 요소가 된다. 따라서 이 연구에서는 Magic Quadrant 분류에 소스 코드 공개 여부를 함께 고려하여 <그림 27>과 같이 클라우드 개발 환경을 제공하는 사업자들을 분류하였다.

클라우드 9, Codenvy, Codeanywhere 등은 사업적인 측면에서 성공적으로 운영되고 있는 것으로 판단된다. 여기에 가상 서버를 제공하는 점, Github 등 외부에 저장된 소스 코드 저장소에 접근하여 소프트웨어개발을 수행할 수 있는 점, 상용 서비스를 위한 서버 운영에 필요한 네트워크 및 서버 관리 관련 기능 등을 통합하여 제공하는 점 등을 고려하면 사업적 측면이나 기술적 전망측면에서 업계의 리더라고 판단할



수 있다. 그러나 사업적 측면을 고려하여 소스를 공개하지 않는 쪽으로 정책을 전환한 점은 아쉬운 측면이 있다. 그러나 Codenvy의 경우 오픈 소스화의 중요성을 인식하고 편집기 등 일부 시스템에 대해서는 소스를 공개하고 Eclipse 재단에 관련기술을 기부하였다.

반면 ApplInventor, Scratch, Webida 등의 시스템은 소스가 공개되어 있기는 하지만 상용 서비스를 운영하기에는 부족한 점이 있다. 지원하는 프로그래밍 언어가 너무 제한적이고 무엇보다 서비스 운영에 활용할 클라우드 서버를 별도로 마련하여 관리해야 하는 점은 다른 시스템과 비교하여 큰 단점이다. 또 제공하는 개발환경도 데이터베이스 연결이나 프로젝트 관리 측면에서 제약이 있기 때문에 상용 서비스의 개발에는 적합하지 않은 부분이 많다. 이 중 Webida의 경우 서비스 운영에 필요한 기능, 예를 들어 가상 서버의 기동, 운영 모니터링, 네트워크 설정 등을 시행할 수 있는 방안이 마련된다면 JavaScript 기반으로 서비스를 개발하고 운영하는데 필요한 기능을 매우 세련되게 제공할 수 있을 것이다.

그러나 ApplInventor, Scratch, Codio 등을 교육 측면에서 보면 직관적으로 이해할 수 있는 형태로 소프트웨어를 개발할 수 있는 장점도 있고, Codio의 경우 가상 교실 환경에서 학생과 교사의 관계를 개선하는 방안을 제시하는 등 향후 기업에서 소프트웨어 개발자의 재교육등에 활용할 경우 유용하게 활용할 부분이 있다. 교육과 관련한 서비스는 대체로 오픈소스로 프로젝트를 진행하고 있다. 다만 Codio의 경우 소스 코드 공개 여부가 명확하지 않다.

Goorm, JSFiddle, Koding 등은 많은 기업에 적용된 사례가 기사에서 언급되고 제공하는 기능이 조사 기간 중에도 지속적으로 개선이 이루어지고 있는 것으로 미루어 사업적으로 상당히 적극적인 개선이 이루어지고 있다고 판단된다. Goorm의 경우 개발 환경의 기능이나 동작 방식 등을 Leader 그룹을 참조하여 적극적으로 개선하고 있고, 교육 분야에서 적용하기 위해서 실시간 chatting 기능 등도 추가하고 있다. JSFiddle의 경우 HTML/CSS, JavaScript 등을 하나의 화면에서 편집할 수 있도록 지원하여 웹 페이지를 간편하게 구성할 수 있다는 장점이 있다. Koding은 오랜 기간 교육 분야에서 사업을 유지해 온 것으로 미루어 상당한 사용자층을 확보하고 있을 것으로 추정된다.

이번 조사를 통해서 모바일 응용프로그램에 기반 한 소프트웨어 개발 교육이나, 클라우드 개발 환경의 편이성 개선 등 특정한 분야나 시장에 중점을 둔 기업들도 의외로 많다는 사실도 확인할 수 있었다. 이러한 변화는 JavaScript에 기반한 편집기 기술이 전에는 진입 장벽이 매우 높은 분야였지만 이제는 상당부분 오픈 소스화가 진행되었기 때문에 가능해진 것으로 판단된다. 예를 들어 Atom editor의 경우 전적으로 JavaScript로 개발되었음에도 전문적인 수준의 편집기능을 제공하고 있고, 이 시스템에 적용된 기술은 손쉽게 웹 브라우저에서 동작하는 서비스로 전환이 가능하다.





## 결론

인터넷의 보급에 따라 다양하게 발전한 클라우드 환경은 이제 소프트웨어 개발 방식 자체를 변화시킬 수 있는 수준으로 발전하였다. 클라우드 기반 개발 환경은 소프트웨어 개발자들이 요구하는 품질과 생산성 향상을 가져올 수 있는데, 언제 어디서나 접근할 수 있다는 특성은 소프트웨어 개발에서 지리적, 시간적 제약을 상당 부분 제거하기 때문에 원격 근무제나 자율 시간 근무제와 같이 소프트웨어 개발자들의 생산성을 향상시킬 수 있는 근무환경을 제공하는데 크게 기여할 수 있기 때문이다.

이 연구에서 조사한 바에 따르면 클라우드 기반 소프트웨어 개발 시스템은 이미 소프트웨어 개발에서 원격 개발에 필요한 기능을 충분히 제공하고 있고 더 나아가서 개발의 생산성과 효율성을 제고할 가능성을 갖고 있다. 다양한 클라우드 기반 개발 환경을 활용하면 소프트웨어 개발에 참여하는 모든 사람들이 간편하게 의사소통을 수행할 수 있다. 이미 다양한 의사소통 도구들이 클라우드 서비스의 형태로 제공되고 있기도 하지만 이를 활용하여 소프트웨어 개발 생산성을 높인 사례들이 다양하게 보고되고 있다. 또 서비스를 운영하는데 필요한 서버도 소규모 스타트업을 중심으로 클라우드 서버를 임대하여 운영하고 있으며, 소프트웨어 개발의 주요한 부분인 소스 코드의 생성과 오류 추적, 그리고 개선된 서비스의 배치도 클라우드를 통해서 수행할 수 있게 되었다.

클라우드 기반의 의사소통, 문서관리, 프로젝트관리, 소스코드편집, 서버 관리 시스템을 잘 활용하는 기업 이라면 이제는 지리적으로 접근이 어려운 곳에 근무하는 소프트웨어 개발자, 혹은 매일 출퇴근하는 것이 어려운 소프트웨어 개발자와 같이 각자 개인적인 사유로 동일한 장소와 시간을 공유하기 어려운 개발자라고 하더라도 충분히 개발팀의 일원으로 포함시킬 수 는 환경을 마련할 수 있게 되었다. 따라서 경력 단절을 겪은 소프트웨어 개발자라고 하더라도 원격지 근무를 통해서 새로운 기술을 익히고 새로운 동료들과 팀을 구성할 수 있을 것이다. 요컨대 기본적인 능력을 유지하고 있다면 소프트웨어 개발자는 언제든지 경력을 복구할 수 있다는 것이다.

물론 이를 실현하기 위해서는 소프트웨어 개발 기업의 입장에서 많은 부분이 변화해야 한다. 예를 들어 이 연구에서 다루지는 않았지만 소프트웨어 개발 시에 개발 장소를 지정하는 관행과 법규 등의 장애물도 엄연히 존재한다. 그러나 적어도 기술적인환경, 그러니까 의사소통, 프로젝트 현황 관리, 정보공유, 그리고 개발환경과 서비스의 운영환경이 모두 클라우드 환경에서 수행될 수 있는 기술적인 기반은 갖추어졌다. 따라서 정책적 지원을 통해 소프트웨어 개발기업이 효과적으로 자본을 투자하고 업무 효율을 올리면서 소프트웨어 개발자들이 경력 단절을 두려워하지 않을 수 있도록 클라우드 기반의 소프트웨어 개발 시스템을 활용할 수 있도록 촉진할 필요가 있다.

2016

04

APRIL

## 이호섭 연구소장(인코어드테크놀로지스) 초청 강연

일 시 2016. 03. 14(월) 09:30 ~ 12:30  
장 소 소프트웨어정책연구소 회의실  
주 제 빅데이터와 스마트 그리드  
참석자 SPRi 연구진

- ▶ 신기술을 통해 생활의 많은 부분에서 진화가 일어났지만, 에너지 분야에서는 신기술 도입과 확산이 더디게 나타남
  - 특히, 전력 계량기의 경우 기술의 진보가 더디어 100여 년 전에 개발된 전력 계량기를 현재에도 사용함
- ▶ 최근 스마트 계량기가 개발되어 가정의 전력 수요를 체크할 수 있게 됨
  - 하지만, 스마트 계량기는 15분 단위로 전력의 사용량을 수집하여 가정 전력 수요의 실시간 변화를 파악할 수 없다는 문제가 있음
- ▶ 인코어드테크놀로지스의 스마트 계량기는 초단위로 전력수요를 수집하는데, 이를 통해 전력 소비의 패턴 분석, 가전 기기별 수요 예측, 기기의 이상 징후 포착 등이 가능함
  - 가정의 전력 데이터를 클라우드 기반으로 수집하여 빅데이터 분석을 통해 실시간으로 분석하여 전력의 수요 예측이 가능함
- ▶ 과거에는 에너지 관리는 불확실성이 비교적 적은 공급 조절에 초점이 맞춰졌는데, 최근에는 전력의 공급에서도 불확실성이 발생하여 에너지 관리의 복잡성이 증대함
- ▶ 스마트 계량기를 통해 전력 수요를 수집함으로써 전력 수요의 예측과 관리가 가능해졌음
- ▶ 전력 수요 관리와 더불어 수요 절감을 위해 행동과학을 도입하여 자발적인 전력 감축이 가능함
- ▶ 에너지 플랫폼은 전력 수요 절감뿐만 아니라, 시니어 케어 등 다양한 분야에서 활용이 가능하여 미래 유망 플랫폼으로서의 기능을 할 것으로 예측됨



(좌)  
이호섭 연구소장의 강의 모습



(우)  
경청중인 SPRi 연구진

# 이희우 대표(Cognitive Investment) 초청 강연

일 시 2016. 03. 21(월) 09:30 ~ 12:30  
 장 소 소프트웨어정책연구소 회의실  
 주 제 린스타트업의 이해와 Case Study  
 참석자 SPRi 연구진

- ▶ 린스타트업(lean startup)이란 : 불확실성 속에서 새로운 제품이나 서비스를 만들기 위해 군살을 빼 빠르게 움직이는 초기단계의 조직이나 기업
- ▶ 린스타트업 정신
  - Think Big, Start Small, Scale Fast
- ▶ 5가지 원칙
  - 1) 초기 단계라면 모두 적용 가능
  - 2) 불확실성 속에서 성공하려면 새로운 방식의 관리 요구(검증된 학습과 측정에 의해 정의된 것들을 관리)
  - 3) 돈벌이를 위해 존재하는 것이 아니라 지속 가능한 사업을 어떻게 만들지 학습하기 위해 존재
  - 4) 만들고, 측정하고, 배우는 피드백 순환을 빨리 돌려야 함
  - 5) 스타트업을 위한 새로운 측정방식이 필요(혁신회계)
- ▶ 린스타트업의 3단계
  - 1) Problem, Solution Fit
  - 2) Solution/Market fit
  - 3) Scale up
- ▶ Lean Canvas
  - 문제가 가장 중요(문제, 솔루션, 핵심지표, 고유의 가치제안, 경쟁우위, 채널, 고객군, 비용구조, 수익원)
  - 회의한다고 답 안나옴
  - 수많은 튜닝을 거친 수십장의 기획서 필요 없다
  - 핵심지표 타협하면 안된다
  - 시장에서 배우자



(좌)  
이희우 대표의 강의 모습

(우)  
경청중인 SPRi 연구진

## 제21회 SPRi Forum

일시	2016. 03. 22(화) 18:30 ~ 20:30
장소	소프트웨어정책연구소 회의실
주제	한국형 CS50(K-CS50) 도입 방안
발표	한태숙 교수(KAIST), 김재현 교수(성균관대), 이호 선임연구원(SPRi), 김형주 교수(서울대), 김두현 교수(건국대)
참석자	관련 종사자 약 50명

발표 제목	발표자	시간
KAIST에서의 프로그래밍 강의 무엇이 중요한가?	한태숙 교수(KAIST)	15분
비전공자를 위한 컴퓨터과학 교육 혁신 방안 - 성균관대 사례를 중심으로	김재현 교수(성균관대)	15분
한국형 CS50(K-CS50) 캠페인 도입 방안	이호 선임연구원(SPRi)	15분
패널토론 : 김형주 교수(서울대), 김두현 교수(건국대) 및 상기 발표자(3명)		45분

※ 발표자료는 연구소 홈페이지([www.spri.kr](http://www.spri.kr))에서 다운받을 수 있습니다.

- ▶ 첫 번째 세션에서는 “KAIST에서의 프로그래밍 강의: 무엇이 중요한가?”를 발표했다.
  - 한태숙 카이스트 교수는 1976년 이후의 카이스트의 프로그래밍 강의 연혁과 Language Syntax가 아닌 Computational Thinking를 지향하는 CS101 프로그래밍 기초과정에 대해 설명하였다.
  - 2006년 이후 같은 문제를 낸 적 없을 만큼 시험문제의 교환 준비와 실습의 어려움이 있으므로 이를 준비하는 교수의 자세와 강의실 장비 등의 학교의 지원체계, 실습을 받아들이는 학생의 정서가 바뀌는 등의 대학의 문화가 전반적으로 바뀌어야 한다고 강조했다.
  - 또한 강의 교재의 구성이 매우 중요하고 특히 프로그래밍 경험이 적은 학생은 초반 4주내에 흥미를 끌어내는 것이 중요하다고 밝혔다. 뿐만 아니라 학생들의 Cheating 방지 방법과 성적 관리 방안을 마련하는 것도 필요하다고 주장했다.
- ▶ 두 번째 세션에서는 '비전공자를 위한 컴퓨터과학 교육 혁신 방안 - 성균관대 사례를 중심으로'를 발표했다.
  - 김재현 성균관대 교수는 전 대학생을 대상으로 SW 비전공자를 위한 성균 소프트웨어 교육원을 설립하여 운영하고 있는 경험을 공유하였다.
  - 대학 입학 전 사전 교육으로 '컴퓨팅과 기초SW'라는 온라인 강의를 제공하고 입학 후 1학기에는 '컴퓨팅사고와 SW코딩'을, 2학기에는 '문제해결과 알고리즘'을 강의한다. 이를 통해 컴퓨팅사고력 기반 과목의 선수체계를 구성할 수 있도록 교과과정을 개편하여 문제해결형 인재를 양성한다고 밝혔다.
  - 인프라 구성을 위해 중등교사 자격증을 보유한 강사를 대상으로 2번 가량의 강사 워크숍에서 강의 및 실습과정을 통해 전교생을 대상으로 표준화된 교육과정을 제공한다.
  - 2016학년도 신입생 전체를 대상으로 1학기 교양 필수로 개설하여 총 16개 강좌 개설하였으나 전교생을 같은 시간대에 같은 강의를 배정하기 위해 각 전공대학에 해당 강좌를 선통보하고 전공 수업을 동 시간에 개설하지 않도록 하는 등의 지원을 받아야 하는 어려움이 있었다고 언급하였다.

- ▶ 세 번째 세션에서는 ‘한국형 CS50(K-CS50) 캠페인 도입 방안’을 발표했다.
  - 이후 소프트웨어정책연구소 선임연구원은 SW융합인재 수요 급증에 대응하기 위해 SW 중심대학에서 비전공자 SW 교육 강화 중이나 일부 대학에 국한된 지원으로 전국 규모의 SW 교육 확산으로는 이 어지지 못하고 SW 중심대학의 비전공자 교육은 전교생을 대상으로 진행되기 때문에 깊이 있는 수 업이 어렵다고 주장했다.
  - 이를 해결하기 위한 대안으로 하버드대학의 인기 과정으로 급부상한 CS50을 소개하였다. CS50의 핵심 성공 요인으로는 문제해결형 기반의 강의로 철저하게 실습위주의 현실에 밀접한 문 제 해결에 초점을 맞추고 풍부한 맞춤형 자료제공과 학교의 적극적 지원으로 꼽았다. 한국에서도 CS 입문 교육에 대한 민간 SW교육이 확산되고 있으며 이를 기반으로 한국형 CS50캠페인을 진행하여 심도 있는 맞춤형 CS입문 교육의 필요성을 주장했다.
  - 이를 위해서 거꾸로 수업 방식을 도입하여 스스로 학습할 수 있는 환경을 제공하고 매년 전국 규모의 데모데이를 개최하는 등 참여를 확산하는 것이 중요하다고 언급했다.
  
- ▶ 강연 종료 후, 서울대학교 김형주 교수, 건국대학교 김두현 교수가 패널로 참여하여 보다 심도 있는 토론 을 진행했다.
  - 김형주 서울대학교 교수는 서울대학교의 경우 컴퓨터학과를 제외하고 공과대학과 자연대, 일부 인문과학을 중심으로 CS강좌가 개설되었으나 각 학과 내에서 자체 개설되어 운영 중이라고 밝히며 비전공자 를 위한 CS강좌는 앞으로 많은 고민이 더 필요할 것이라고 언급했다.
  - 김두현 건국대학교 교수는 건국대학교에서는 컴퓨팅적 사고라는 과목을 교양필수 과목으로 개설하여 1600명을 대상으로 동영상과 실습을 Flipped Learning방식으로 진행하고 있다고 밝혔다.
  - 또한, 인문사회예체능계열(30개반)과 정보통신계열(6개반)로 나누어서 진행하고 있으나 아직은 학교 내에서 풀어야 할 문제가 다수 존재한다고 언급했다.
  - K-CS50 캠페인에 대해서는 학생 - 교수 - 학교(인프라)의 3개 측면의 입장을 보다 상위레벨에서 고려 하여야 한다고 주장했다. 즉, 세밀하게 학생을 이해하고 분석하여 디자인하고 수업을 위한 시설과 강사진 을 확보하기 위해 노력해야 한다고 강조했다.
  - 또한 “기존의 SW중심대학에서 진행하는 비전공자를 대상으로 하는 강좌와는 차별성을 가지고 보다 심도 있는 수업 중심으로 진행하는 것이 좋을 것”이라고 주장했다.
  - 김재현 성균관대학교 교수는 “컴퓨팅적 사고력을 위해서 학생의 흥미와 동기부여가 중요하다”고 강조하였다.
  - 고건 이화여자대학교 교수는 “전공과 상관없이 공통으로 듣는 기본 강좌와 다양한 전공자를 위해 각 전공에 맞는 융합형 실습체계를 갖추어야 할 것”이라고 주장하였다.



발제자 발표모습 및 패널토론

# 2016 SPRi Spring Conference

## - AI is here. 성큼 다가온 인공지능

일 시 2016. 03. 08(화) 13:00 ~ 17:00  
 장 소 경기창조경제혁신센터 국제회의장  
 주 제 AI is here. 성큼 다가온 인공지능  
 주 최 미래창조과학부  
 주 관 소프트웨어정책연구소  
 참석자 SW관련 종사자 약 500명



- ▶ 소프트웨어정책연구소(SPRI)는 미래창조과학부 주최, 경기과학기술진흥원과 한국정보과학회 인공지능소사이어티 후원의 ‘2016 SPRi Spring Conference’가 “AI is Here, 성큼 다가온 인공지능”이란 주제로 3월 8일 판교 테크노밸리 경기창조경제혁신센터 국제회의장에서 500여 명의 인파가 몰려 성황리에 개최됐다.
- ▶ 이번 컨퍼런스에 참석한 최양희 미래창조과학부 장관은 축사에서 “인공지능은 미래 사회의 변혁을 가져올 창조엔진이다.”며 “이 번 컨퍼런스가 인공지능을 통해 다양한 문제를 발굴하여 산업과 사회의 혁신을 견인하는 첫단추 역할을 했다”고 말했다.
- ▶ 특히, 이세돌 9단과 세기의 대결을 앞두고 있는 구글 인공지능 ‘알파고’ 개발자 데이비드 실버 박사가 직접 이를 소개하여 참석자의 많은 관심과 흥미를 끌었다. 이세돌 9단과의 대결에 따로 준비한 것이 있는가에 대한 질문에는 “이세돌 9단에 특화된 알고리즘을 준비하지는 않았다”고 밝혔다.
- ▶ 독일 인공지능 연구소(DFKI) 기술책임자이자 이사회임원인 안드레아스 덴겔 교수도 인공지능연구소 소개와 더불어 DFKI에서 진행하는 인공지능 연구 사례에 대해서 강연했다. 특히, ‘데이터를 기반으로 한 인식’을 주제로 사람의 감정적인 생각조차 모델링할 수 있다는 가능성을 소개했다.
- ▶ 딜로이트의 정성일 전무(인공지능과 경영), IBM 이강윤 상무(왓슨 소개 및 인지컴퓨팅 비즈니스 전략), 가트너 최윤석 상무(가트너의 미래예측), 소프트웨어정책연구소 김석원 실장(미래일자리 변화) 등 국내 전문가들도 인공지능 관련한 다양한 주제를 발표했다.
- ▶ 김진형 소프트웨어정책연구소(SPRI) 소장은 “(이세돌 바둑대결에서 보듯) 인공지능은 이미 우리 실생활에 매우 가깝게 다가와 있다”며 “이번 컨퍼런스를 통해 일반 대중에게 성큼 다가온 인공지능에 대한 막연한 궁금증을 풀 수 있는 계기가 됐다.”고 말했다.

발표 제목	발표자
인공지능과 우리의 미래	김진형 소장(소프트웨어정책연구소)
Mastering the game of Go with deep neural networks and tree search	Dr. David Silver (구글 DeepMind)
Augmenting Human Mind by Artificial Intelligence	Prof. Dr. Andreas Dengel (독일인공지능연구소)
Business Impact from AI	정성일 전무(딜로이트)
Watson, Cognitive Computing Business Strategy	이강윤 상무(한국IBM)
Gartner Top Predictions	최윤석 상무(가트너)
Changes in future jobs	김석원 실장(소프트웨어정책연구소)

발표모습 및 행사전경

※ 발표자료는 연구소 홈페이지(www.spri.kr)에서 다운받을 수 있습니다.(일부 발표자는 비공개)