

엑셀 VBA DAO 기능을 이용한 효율적인 데이터 검색 기능 설계

장승주^{1*} · 류대현²

Design of Efficient Data Search Function using the Excel VBA DAO

Jang, Seung Ju^{1*} · Ryu Dae-Hyun²

¹Department of Computer Engineering, Dongeui University, Busan 176, Korea

²Department of Computer IT, Hansei University, Gyeonggi-do 604-5, Korea

요 약

본 논문은 마이크로소프트 엑셀에서 데이터 분할 알고리즘을 이용한 효율적인 데이터 검색 시스템을 제안한다. 엑셀에서 VBA 기능을 이용하여 해당 데이터를 빠르게 검색하는 알고리즘을 제안한다. 본 알고리즘은 찾고자 하는 데이터가 있는 시트를 지정하게 된다. 해당 시트가 지정이 되고나면 해당 시트 내의 데이터에 대한 시작과 끝을 찾게 된다. 시작 셀의 위치에서부터 원하는 키워드와 중간 값을 비교하게 된다. 이런 식으로 데이터 끝까지 검색을 하게 된다. 본 논문에서 제안하는 알고리즘을 실제 VBA 프로그램을 이용하여 구현하고 실험을 하였다. 실험 결과 기존의 방식과 비교했을 때 일반적인 순차 검색 방식보다 성능이 좋음을 확인할 수 있었다.

ABSTRACT

In this paper, I propose an efficient data search system using data partitioning algorithm in Microsoft Excel. I propose searching algorithm to retrieve data quickly using VBA functioning in the Excel. This algorithm is to specify the sheet you are looking for. Once the sheet is specified, the algorithm searches the beginning and the end of the data in the sheet. The algorithm compares intermediate values and key words, from the starting position of the cell. In this way, it will search data to the end. This proposed algorithm was implemented and tested in the Excel system using VBA program. The experimental results showed that the performance was better than that of the conventional sequential search method.

키워드 : 데이터 검색 알고리즘, 엑셀 데이터 검색, 엑셀, DAO, VBA

Key word : Data Search Algorithm, Excel Data Search, Excel, DAO, Visual Basic for Application

접수일자 : 2013. 08. 21 심사완료일자 : 2013. 10. 04 게재확정일자 : 2013. 10. 18

* **Corresponding Author** Jang, Seun Ju(E-mail:sjjang@deu.ac.kr, Tel:+82-51-890-1710)

Department of Computer Engineering, Dongeui University, Busan 176, Korea

Open Access <http://dx.doi.org/10.6109/jkice.2014.18.1.217>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

우리가 일반적으로 소량의 데이터 처리를 위해서 개인이 가장 많이 사용하는 것이 엑셀 프로그램이다. 엑셀 프로그램은 전 세계적으로 가장 널리 사용되고 있는 데이터 처리 시트이다. 엑셀은 대차대조표, 성적표, 고객 관리, 재고 관리 및 재무제표 등과 같은 행과 열로 구성된 데이터가 서로 연관을 갖는 수치 분석에 유용하게 활용된다. 특히, 공식을 사용자가 정의하여 사용할 수 있을 뿐만 아니라 사전에 정의된 수많은 함수들을 활용하여 효과적으로 분석할 수 있다. 또한 다양한 서식을 통해 알기 쉽고 보기 좋은 보고서를 작성할 수 있다. 둘째, 엑셀은 그래픽 프로그램이다. 2차원, 3차원 차트의 작성, 다양한 차트 서식 기능 등 엑셀의 차트 작성 능력은 어떤 스프레드시트 프로그램보다 뛰어나다. 프로그래밍 언어에는 여러 가지가 있지만 그 중에서도 비주얼 베이직(Visual Basic)은 사용하기가 편리하고 개인용 컴퓨터에 적합하다. 비주얼 베이직은 베이직을 기반으로 만들어진 언어이다. BASIC(Beginner's All-purpose Symbolic Instruction Code)은 1960년대 미국에서 학생들에게 프로그래밍 개념을 쉽게 소개할 목적으로 개발된 언어로서 명령의 종류가 적고, 문법이 간단하여 컴퓨터에 전문적인 지식이 없어도 쉽게 배울 수 있는 언어이다.

본 논문에서 구현한 검색 기능은 엑셀의 VBA기능을 이용한다. 엑셀의 VBA 기능에서 DAO(Data Access Objects) 기능을 이용하여 간단한 DB 기능을 활용하여 검색 기능을 구현한다. 본 논문에서 제안한 검색 기능은 데이터를 보다 편리하고 효율적으로 검색이 가능하도록 한다. 본 논문에서 제안하는 데이터 검색 기능은 DAO 기능을 이용하여 보다 빠른 검색을 보장한다. 본 논문은 2장은 관련연구, 3장은 데이터 검색 기능 설계, 4장 실험, 5장 결론의 순으로 구성되어 있다.

II. 관련 연구

엑셀은 전 세계 많은 사람들이 이용하는 범용 데이터 처리 소프트웨어이다. 엑셀을 활용하여 할 수 있는 주요 작업은 다음과 같다. 첫째, 엑셀은 스프레드시트 프로그램이다. 주어진 데이터를 단순히 숫자만을 나열하는 것이 아니라 차트를 활용하여 제시한다면 훨씬 더

이해하기 쉬운 것이다. 둘째, 엑셀은 데이터베이스 관리 도구이다. 엑셀은 레코드 및 필드에 관련된 자료들에 대해 레코드의 관리, 탐색, 정렬, 자동 필터, 보고서 작성 등의 작업을 지원하며 수행한 작업 결과를 워크시트나 표준 데이터베이스 파일에 저장할 수 있다. 특히 자동 필터 기능을 활용하여 복수의 레코드나 필드를 중심으로 복수의 조건을 만족하는 자료를 정렬할 수 있으며 피벗 테이블을 활용하여 손쉽고 유용한 방법으로 자료를 정리, 요약할 수 있다[1-2, 5-6].

비주얼 베이직의 특징으로는 시각적인 개발 환경을 가지고 있다. 비주얼 베이직은 원도 환경에서 각종 응용 프로그램을 개발할 때 눈으로 보면서 작업을 할 수 있도록 하는 편리한 도구를 지원한다. 또한, 객체 지향적 개발환경을 갖고 있다. 이때 객체지향(object orientation)이란 독립적 객체로 된 기능들을 조립식 주택을 조립하듯 맞추어서 하나의 프로그램을 완성하는 것이다. 비주얼 베이직은 프로그램을 구성하는 기본요소를 하나의 객체로 제공하는 객체 지향적 개발환경을 제공하기 때문에 프로그램을 작성하기 쉽고 생산성이 높다. 엑셀은 자동화된 개발환경이다. 비주얼 베이직에서는 코드작성과정에서 명령문 또는 함수를 사용할 경우 구문정의 박스가 자동으로 화면에 나타나고 한 문장의 입력이 끝날 때마다 자동으로 오류를 최소화할 수 있다 [3, 4-7].

마이크로소프트사의 오피스97에서부터는 Excel, Word, Powerpoint, Access 등에서 공통으로 사용되는 개발 언어가 되었으며, VBA는 이러한 응용 프로그램에서 사용하는 비주얼 베이직 언어이다. 따라서 VBA나 비주얼 베이직은 같은 언어인 것이다. 다만 편집기가 비주얼 베이직은 독립적으로 즉, 하나의 프로그램으로 사용되는 것이고, VBA는 Excel, Word, Access, Powerpoint 등 마이크로소프트사의 오피스 제품에서 사용되는 비주얼 베이직인 것이다 [8-13].

VBA는 다른 프로그램에서 비주얼 베이직 코드를 이용할 수 있도록 비주얼 베이직 코드를 적절히 개조한 것이다. Excel을 포함한 오피스의 모든 제품에는 비주얼 베이직 엔진이 내장되어 있어서 비주얼 베이직 코드로 사용자 프로그램을 작성할 수 있다. 매크로는 일련의 원하는 작업을 코드로 기록해 두었다가 필요할 때 이 코드를 다시 실행하는 것으로 비주얼 베이직 코드로 작성된다. VBA는 Excel 매크로와 같은 사용자 프로그램을 작성할 수 있도록 엔진이 내장되어 있기 때문에

오피스 제품을 효율적으로 사용할 수 있는 유용한 프로그램을 만들 수 있다[13]. VBA는 Excel이나 OLE automation을 지원하는 윈도우 프로그램에서 제공하는 수많은 기능을 갖는 부속품들 즉, 작은 부속 프로그램들을 다시 만들 필요 없이 제공되는 부속품을 조작할 수 있는 장점을 가지고 있다.

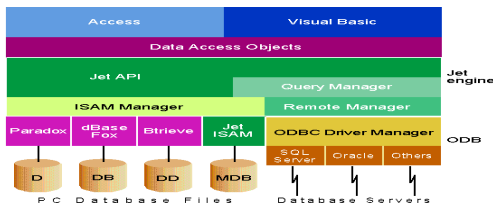


그림 1. DAO와 관련된 시스템 구조
Fig. 1 DAO Related System Structure

그림 1은 DAO와 관련한 시스템 구조를 나타낸다. DAO는 Jet 엔진이라 부르는 DLL의 집합들을 사용한다. 이러한 DLL은 MDB 파일들이나 다양한 데이터베이스 포맷 파일에 접근할 수 있도록 제공한다.

III. 데이터 검색 기능 설계

본 논문에서 개발하고자 하는 시스템 구조는 다음 그림 2와 같다.

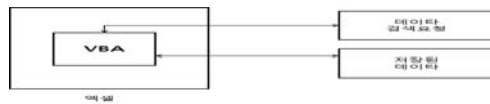


그림 2. 개인별 데이터 관리 시스템 내부 구조
Fig. 2 Internal Structure of the Personal Data Management System

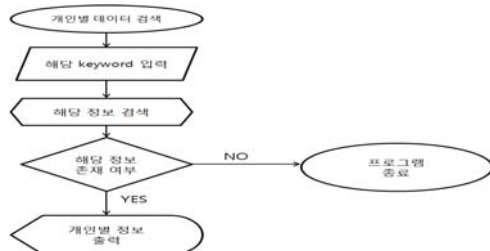


그림 3. 개인별 데이터 검색 기능 순서도
Fig. 3 Flow Chart of the Personal Data Search Function

위 그림 3은 개인별 데이터 검색 기능 순서도이다. 사용자 데이터 관리는 기존 엑셀 시트를 이용한다. 엑셀 시트에 저장된 데이터는 VBA를 이용하여 관리 알고리즘을 설계한다. 개인별 데이터 검색 기능 흐름도는 그림 3과 같다. 개인별 정보 검색을 하게 되면 엑셀 VBA의 DAO 기능을 이용하여 수행하게 된다. 엑셀 DAO 기능은 일반 데이터베이스와 같이 대용량 자료를 처리하는데 효율적이다. 개인별 데이터 검색 요청이 발생하게 되면 해당 키워드를 입력하게 된다. 본 논문에서 제안하는 모듈에서 해당 키워드 입력을 받게 되면 DAO 모듈에서 해당 키워드를 이용하여 정보를 검색하게 된다. 해당 정보 검색에서 해당 정보 존재 여부를 찾게 된다. 개인 정보 조회에서 일치하는 데이터가 있을 경우 상세 개인 정보를 출력하게 된다. 그렇지 않을 경우에 실행 프로그램을 종료하게 된다. 본 논문에서 제시하는 VBA 기능을 이용한 효율적인 검색 알고리즘의 동작 과정은 다음 그림 4와 같다.

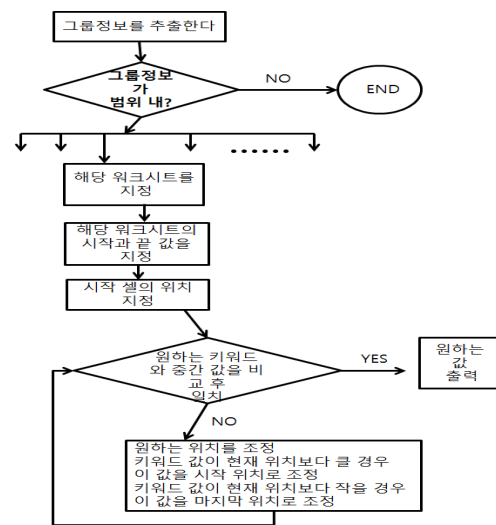


그림 4. DAO 기능을 이용한 분할 검색 알고리즘
Fig. 4 Partition Search Algorithm Using DAO Function

먼저 수행하고자 하는 그룹 정보를 추출한다. 그룹 정보가 범위 내에 없을 경우에는 알고리즘 동작을 중지한다. 그룹 정보가 범위 내에 있을 경우는 해당 시트를 지정하게 된다. 해당 시트가 지정이 되고나면 해당 시트 내의 데이터에 대한 시작과 끝을 찾게 된다. 시작과 끝의 위치를 찾게 되면 시작 셀의 위치를 지정하게 된

다. 시작 셀의 위치에서부터 원하는 키워드와 중간 값을 비교하게 된다. 이때 비교하는 값과 값이 일치할 경우는 원하는 자료를 찾은 경우이다. 그렇지 않을 경우는 두 가지 경우로 나누어진다. 키워드 값이 현재 위치보다 클 경우는 시작 위치를 현재 위치로 조정한다. 키워드 값이 적을 경우는 끝나는 위치 값을 현재 위치 값으로 조정한다. 알고리즘은 이와 같이 반복해서 동작이 이루어지게 된다.

실제 이 프로그램에 많은 사람의 데이터가 등록되어 있다. 이처럼 많은 데이터를 수작업으로 하기에는 한계가 있다. 따라서 해당 사람의 데이터를 찾고 싶으면 일일이 찾아 볼 수가 없으므로 본 프로그램에서 제공하는 검색 기능을 이용하면 된다. 다음 그림 5는 본 논문에서 제안하는 검색 기능에 대한 동작 화면이다.

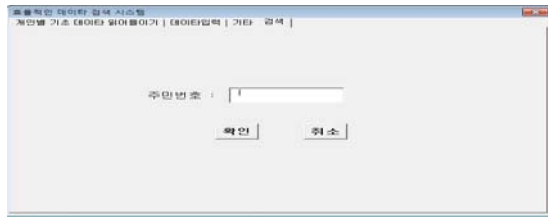


그림 5. 데이터 검색 화면
Fig. 5 Data Search Screen

다음과 같이 '검색' 탭을 선택하면 데이터를 검색할 수 있게 텍스트 박스에 찾고자하는 키값을 입력 할 수 있다. 가령 예를 들어 '장승주' 라는 사람의 데이터를 검색 한다고 하자. 그러면 아래와 같이 해당 사용자에게 대한 주민번호를 입력하게 된다.



그림 6. 검색 기능에 주민번호 입력 화면
Fig. 6 User Identification Input Screen in Search Function

원하는 사용자를 검색하고 난후 정상적으로 원하는 데이터가 찾아졌을 경우에는 다음 그림 7과 같은 메시지 박스를 확인할 수 있다.

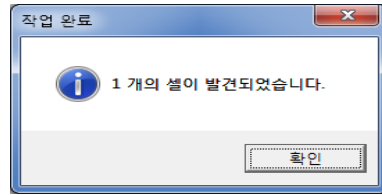


그림 7. 원하는 개인 정보 발견
Fig. 7 Match the Exact Personal Cell

만약 찾고자 하는 사람이 발견되지 않았을 경우는 다음 그림 8과 같이 메시지 박스를 띄우게 된다.

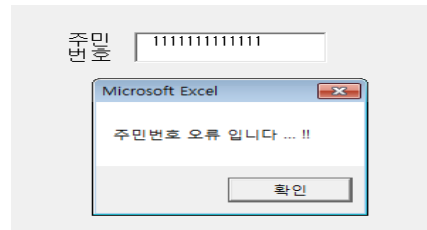


그림 8. 잘못된 개인 정보 입력시 오류 메시지 화면
Fig. 8 Error Message Screen in Unnamed User Identification Input

그림 5, 그림 6, 그림 7, 그림 8과 관련한 구현 소스 중 일부 코드는 다음과 같다.

```

.....
If rngcell Is Nothing Then
    MsgBox "해당 주민번호의 셀을 찾지 못했습니다.", vbInformation
    GoTo out
End If
.....
If Not rngcell Is Nothing Then
    strAddress = rngcell.Address
    Do
        rngcell.Select
        Selection.Interior.ColorIndex = 4
        '중간값과 키워드를 비교한다.
        .....
    Loop While rngcell.Address <> strAddress
    .....
    Msg = Msg & i & " 개의 셀이 발견되었습니다." &
    .....
    MsgBox "소요시간"&Format(Timer - oldTime, "#0.00")

```

그림 9. 구현된 소스 코드
Fig. 9 Implementation Source Code

위 그림 9는 앞의 그림 5, 그림 6, 그림 7과 관련한 프로그램 소스 코드이다. 그리고, 그림 10의 소요시간을 측정하는 부분도 포함되어 있다.

IV. 실험

본 논문에서 개발 환경으로 엑셀을 사용한다. 특히, 엑셀 내의 VBA 기능을 이용하여 개발한다. 그리고 VBA내의 데이터 처리를 위하여 DAO(Data Access Objects) 기능을 이용한다. 본 논문에서 개발한 내용에 대하여 정상적인 동작 여부를 체크하기 위한 실험을 수행하였다. 본 논문의 구현 결과를 실제 실험을 수행하였다. 실험은 실제 윈도우 운영체제가 탑재된 시스템 환경에서 엑셀 프로그램을 이용하여 이루어졌다.

실험을 위하여 개인별 데이터를 1000건 정도 입력하였다. 실험을 위한 컴퓨터 시스템 환경은 윈도우 7 운영체제를 사용하였다. 윈도우 7 운영체제 상에 엑셀은 마이크로 소프트 엑셀 2010 버전을 사용하였다. 1000건의 데이터 입력을 통해서 본 논문에서 제시하는 검색 알고리즘이 DAO를 통한 검색 기능이 우수함을 보인다.

입력된 데이터에 대해서 VBA상에서 일반적인 순차 데이터 검색 알고리즘을 이용한 경우와 DAO를 이용하여 본 논문에서 제안한 검색 알고리즘을 이용한 경우에 대하여 실험을 수행하였다. 기존의 순차 데이터 검색 알고리즘은 데이터 검색을 할 경우에 처음부터 차례로 내려가면서 원하는 데이터 인지를 판단하는 방식이다. 다음 그림 10은 400개 데이터에 수행 시간 측정 화면이다.

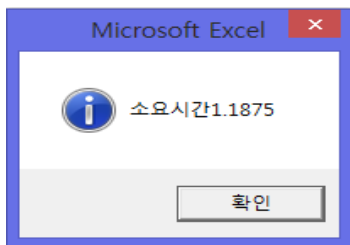


그림 10. 400개 데이터에 대한 시간 측정 결과 화면
Fig. 10 Time Check Screen for 400 Data

표 1. 일반 순차 데이터와 DAO를 활용한 경우의 수행 시간 (단위 : 초)

Table. 1 Execution Time for General Environment Data and DAO Data(Unit : Second)

	300	400	500	600	700	800	900	1000
일반 순차 데이터 검색	1.11	1.23	1.28	1.35	1.39	1.50	1.58	1.65
DAO를 사용한 분할 데이터 검색	1.01	1.15	1.20	1.27	1.3	1.43	1.5	1.55

표 1은 일반 데이터에 대해서 DAO를 활용한 경우의 수행 시간을 측정한 결과이다. 위 <표 1>의 수행 시간은 각 데이터 개수에 대해서 10회 이상 수행한 결과에 대한 평균값이다. 위 결과에서 보듯이 일반 순차 데이터 검색 시간이 DAO를 사용한 분할 검색의 경우보다 전반적으로 많이 소요됨을 알 수 있다. 이러한 이유는 분할 검색 알고리즘을 사용한 원인이 있지만, 분할 검색 알고리즘과 DAO 기능을 적절히 조합한 결과로 볼 수 있다. 위 실험 결과 기존 엑셀 데이터를 단순히 VBA 프로그램을 이용한 경우와 VBA에서 DAO를 이용한 검색 알고리즘을 사용한 경우에 다소 성능의 차이가 있음을 확인할 수 있었다.

V. 결론

본 논문에서는 엑셀 데이터 DAO 기능을 이용하여 빠른 검색을 할 수 있도록 시스템을 설계 및 구축하였다. 본 논문에서 개발된 기능은 사용자의 데이터를 보다 효율적으로 검색하고 관리하기 위한 것이다. 본 논문에서 사용한 개발 환경은 엑셀을 이용한다. 특히, 엑셀 내의 VBA 기능을 이용하여 개발한다. 그리고 VBA 내의 데이터 처리를 위하여 DAO(Data Access Objects) 기능을 이용한다. 이러한 DAO 기능을 통해서 보다 대용량 데이터를 신속하고 정확하게 처리할 수 있다.

본 논문에서 개발한 기능의 상세 내용은 먼저 수행하고자 하는 그룹 정보를 추출한다. 그룹 정보가 범위 내에 없을 경우에는 알고리즘 동작을 중지한다. 그룹

정보가 범위 내에 있을 경우는 해당 시트를 지정하게 된다. 해당 시트가 지정이 되고나면 해당 시트 내의 데이터에 대한 시작과 끝을 찾게 된다. 시작과 끝의 위치를 찾게 되면 시작 셀의 위치를 지정하게 된다. 시작 셀의 위치에서부터 원하는 키워드와 중간 값을 비교하게 된다. 이때 비교하는 값과 값이 일치할 경우는 원하는 자료를 찾은 경우이다. 그렇지 않을 경우는 두 가지 경우로 나누어진다. 또한, 본 연구 과제에서 개발한 내용에 대하여 정상적인 동작 여부를 체크하기 위한 실험을 수행하였다. 일반 환경 데이터 검색 시간이 DAO를 사용한 경우보다 전반적으로 많이 소요됨을 알 수 있다. 기존 엑셀 데이터를 단순히 VBA 프로그램을 이용한 경우와 VBA에서 DAO를 이용한 검색 알고리즘을 사용한 경우에 다소 성능의 차이가 있음을 확인할 수 있었다.

REFERENCES

- [1] Park, Hae won, "Program Development and Distribution Analysis Using Excel VBA", Seoul Nat'l Univ. M.S Theses, 2011.
- [2] Park, Jun Wong , "Logistics Regression Analysis Program Using Excel VBA", Seoul Nat'l Univ. M.S Theses, 2012.
- [3] Lee, Ahn Na, "Sample Extraction Program Using Excel VBA", Seoul Nat'l Univ. M.S Theses, 2013.
- [4] Kim, Sung Min, "Log Linear Regression Analysis Using Exce VBA", Seoul Nat'l Univ. M.S Theses, 2013.
- [5] Choi Hyun Seok, Park Chul Yong, "Implementation Education Tool of the Variable Regression Model Using Excel VBA", *Korea Data Information Journal*, 24(3) pp.593- 601, 2013.
- [6] Choi Hyun Seok, Ha Chung Cheol, "Implementation Statistics Education Tool of the Procedural Base Using Excel Macro", *Korea Data Information Journal*, Vol. 22 No 4, pp.643-650, 2011. 7.
- [7] Choi Jun Sun, *Excel Macro VBA Bible*, HanBit Media, 2012.6.
- [8] John Walkenbach, *Excel 2007 Power Programming With VBA*, Predeck, 2010.11.
- [9] Lee Chong Seok, *Excel 2010 Macro & VBA*, Digital Media Books, 2012.3.
- [10] Lee Gang Jae, *200 Excel Functions for Field Workings*, Sejin Books, 2011.2.
- [11] Richard Mansfield, *Mastering VBA for Office 2010*, SYBEX, 2010. 8.
- [12] Ko Eung Nam, *Visual Basic Programming*, HanBit Media., 2007. 10.
- [13] Kim Eun Jung, *Advanced Excel & VBA Programming for Expert*, E-Han Media, 2012. 2.



장승주(Jang, Seung Ju)

1985년 부산대학교 계산통계학(전산학) 학사
 1991년 부산대학교 계산통계학(전산학) 석사
 1996년 부산대학교 컴퓨터공학 박사
 1987년 ~ 1996년 한국전자통신연구원(ETRI) 시스템 SW연구실
 1993년 ~ 1996년 부산대학교 시간강사
 2000년 ~ 2002년 Univ. of Missouri at Kansas City, visiting professor
 1996년 ~ 현재 동의대학교 컴퓨터공학과 교수
 ※ 관심분야 : 운영체제, 임베디드 운영체제, 분산시스템, 시스템 보안, 스마트 폰 시스템 운영체제



류대현(Dae-Hyun Ryu)

1983년 부산대학교 전기기계공학과 졸업(공학사)
 1985년 부산대학교 대학원 전자공학과 졸업(공학석사)
 1997년 부산대학교 대학원 전자공학과 졸업(공학박사)
 1987 ~ 98 전자통신연구원 선임연구원
 1998년 한세대학교 IT 학부 교수
 ※ 관심분야 : IoT, M2M, 정보보호, 영상처리