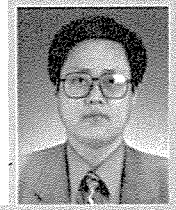


# COBOL 세대들의 귀향을 바라며



(주)객체정보기술/대표이사 김 계 철

**김**퓨터와 통신의 급격한 발전은 현대사회의 구조를 변화시켜 나가기에 이르렀으며 특히 인터넷 환경의 보편화는 기존의 수직화된 정보 질서체계를 무너트렸다. 또한 최근들어 신문이나 잡지를 보면 객체지향 방법론이니 객체지향 언어니 그리고 소프트웨어 컴포넌트니하는 새로운 용어들이 자주 등장하게 된다.

이러한 신 지식관련 용어들은 일칭 COBOL 세대라고 하는 불과 몇 년 전까지 정보화 세계의 한 일익을 담당했던 즉 기존 일선에서 소프트웨어 개발의 귀재라고 일컫는 선배들을 일거에 컴맹(?)으로 만들어버리고 20대의 신세대 기술 인력만이 마치 새로운 정보화 세계를 이끌어갈 수 있는 것처럼 혼동을 야기시키는 아이러니한 현상을 나타나게 하였다.

그렇다면 이들 신세대와 IMF와 함께 소프트웨어 개발 세계에서 떨어져 간 COBOL 세대의 엔지니어가 힘을 모아 더욱 큰 시너지 효과를 발휘할 수 있는 방안은 없을까 ?

아마 어떤 프로젝트를 개발할 시 누구나 가장 중요하다

고 인정하는 분야가 시스템 분석 및 설계 분야일 것이며 바로 이 부분은 실 생활 속에서 흘러가는 업무의 Flow를 신세대들이 선배들을 따라가기가 벅찰 것이다. 그리고 이 유야 어떠하든지 간에 선배들은 계속적으로 발전하고 새로 생기는 프로그램 언어를 습득하기에 또한 벅찰 것이다.

컴퓨팅 시스템의 초창기에는 프로그래머들이 대개 면밀한 문제 분석에 의존하지 않은 것이 보통이었다. 문제 분석이 있었다 해도 거의 잊어 버리거나 컷등으로 흘려 보내기 일수였다. 그들은 프로그램을 밑바닥부터 만들어 올렸으며, 필요한 코드 소스를 스스로 만들었다. 그 당시에는 이러한 것들이 추억담이나 무용담이 될 수 있었겠지만 오늘날과 같이 위험도 높은 업계에서는 부적당한 것이다. 점점 복잡해지는 세상에 컴퓨터 기반 시스템 역시 덩달아 복잡해지고 있다. 여러대의 하드웨어와 소프트웨어는 기본이고 하드웨어를 연결하기 위한 네트워크 시스템과 데이터베이스도 필요해졌다.

여기서 소프트웨어의 위기란 말이 강하게 들어온다. 지난 30년간 컴퓨터의 속도와 성능은 수백배 이상의 놀란만큼 발전하고 있지만 소프트웨어의 생산성은 지난 30년간 거의 진전이 없었다. 즉 소프트웨어의 위기는 역설적으로 말해 지난 30년간 전환점이 없었다는 것에 기인한다.

이를 해결하기 위한 가장 강력한 대응책으로 소프트웨어 개발의 전환점을 마련해 줄 수 있는 소프트웨어 컴포넌트 개발 기술이 등장하고 있다.

소프트웨어 컴포넌트란 소프트웨어를 마치 레고 블록처럼 하나의 특정 기능을 가진 소프트웨어를 CORBA, COM+, EJB 등 국제표준에 입각하여 부품과 같이 만들어, 이들 각각의 기능을 가진 소프트웨어들을 결합시켜 하나의 응용 소프트웨어를 만드는 것으로 사용자는 자기의 사용환경에 맞는 응용소프트웨어를 개발하기 위해 저렴한 가격으로 신속히 커스터마이징 할 수 있고 실무 적용 후 유지보수를 손쉽게 할 수 있는 새로운 소프트웨어 개발 기술이다. 그리고 소프트웨어 컴퍼넌트 그 자체를 전자상거래 대상의 한 콘텐츠로 유통시킬 수 있어 향후 시장을 지배할 새로운 상품 및 기술로 정부에서도 국가산업으로 육성하고 있다.

소프트웨어 컴포넌트는 객체지향적인 소프트웨어 모듈이며 이는 객체지향 언어로 개발되어 있다. 객체지향 언어란 과거 COBOL, FORTRAN 등 소프트웨어 개발 언어가 고가의 하드웨어 이용의 효율성을 높이기 위한 구조로 만들어진 반면, 이미 저가격 고 성능화된 하드웨어 이용의 효율성을 높이기 보다는 소프트웨어의 재사용을 통한 개발 생산성 향상 및 유지보수의 편의성 위주로 만들어진 컴퓨터 언어이다.

여기서 주목하여야 할 부분은 아무리 소프트웨어 컴포넌트가 소프트웨어의 위기를 해결하기 위한 가장 강력한 대응책이라고 하지만 그 컴포넌트를 만들기 위해선 사전에 업무 분석 및 설계를 하여야 하며 이는 컴퓨터 지식과 개발 프로젝트의 업무를 잘 알고 있는 사람이 맡아야 할 부분이다. 이 부분은 신 세대 기술 인력들이 단순간에 습득

하기는 어려울 것이다.

그래서 이 업무 분석/설계부분을 객체지향언어에 알맞은 새로운 분석 설계 기법인 UML(Unified Modeling Language)을 적용하여 실전 프로젝트 경험이 많은 COBOL 세대가 이 부분을 습득하여 맡는 것이 소프트웨어의 품질을 높이는데 더욱 이상적이라고 생각된다. 또한 이들이 이 부분을 습득하는 데는 그리 긴 시간을 요하지 않을 것이다. 불과 3개월에서 최대 6개월 정도면 이들은 이를 완전 소화해 낼 것이다. 그리고 개발 부분은 새로운 언어의 지식을 가진 신세대들에게 맡겨야 할 것이다.

지난해 11월 산,학,연, 공동으로 한국소프트웨어컴포넌트 컨소시엄(KCSC, 회장 오길록)이 결성되고 여기서 소프트웨어 컴포넌트 유통시장을 형성하고 공용 컴포넌트와 컴포넌트 기반 소프트웨어 구축 기술의 개발로 우리나라를 세계적 수준의 컴포넌트 기술 국가로 발전시키려는 활동을 하고 있다. 하지만 객체기술의 산업계정착에 따른 기술 인력은 매우 부족한 현실이며 교육 또한 몇몇 기관 및 업체에서 간헐적으로 하는 등 거의 전무한 실정이다. 이 기술 인력은 현재 고급 엔지니어로서 인정을 받고 있다. 짧은 기일 내에 전문가를 양성한다는 것은 무리일 것이고 대안으로서 IMF로 소프트웨어 개발의 세계에서 멀어져 간, 그리고 Y2K와 관련해서 일시적으로 해외로 취업한 COBOL 세대들을 모아 이들을 상대로 UML 방법론을 교육시켜 신세대 엔지니어와 함께 신기술을 발전 시켜나가야 한다는데 공감하길 기대한다.