

내가 프로그래밍을 처음 시작한 지 벌써 20년 가까이 된다. 처음 개발을 시작할 당시에는 지금처럼 많은 사람이 컴퓨터를 쓰게 될 것이라고는 상상조차 못했다. 8비트 컴퓨터인 애플 컴퓨터를 사용하던 시절이었으니 가정에서 취미로 가지고 노는 정도였지 업무용으로 사용하다는 생각은 꿈에도 하지 못할 상황이었다. 지금은 컴퓨터뿐만 아니라 인터넷이 보편화하면서 가정마다 없어서는 안 될 필수품으로 자리잡게 된 것을 보면 격세지감마저 느낀다.

개발 문화도 많이 달라졌다. 당시에는 소프트웨어 개발자가 먹고 살 수 있는 직업이 아니었다. 초기에는 개발 자체에 재미를 느끼는 마니아들이 주축을 이루었지만, 생계를 위해서 다른 직업을 가지고 있는 경우가 많았다. 지금은 개발자도 어엿한 직업으로 자리잡았지만, 열정 측면에서 보면 지금의 개발자들이 배워야 할 점이 많지 않을까 생각한다. 즉 요즘 같은 환경에서도 옛날 마니아 시절의 사람들만큼 열정을 가진 사람들만이 성공할 수 있다는 점을 명심해야 한다.

초창기에는 국내 개발자들은 많은 어려움을 겪어야 했다. 나도 자료를 구하는 데 무척 애를 먹었다. 당시 국내에는 IT 서적들이 별로 없었고, 원서를 구하기도 만만치 않았다. 인터넷도 없었고, 주위에 물어 볼 사람이 있는 것도 아니었다. 혼자서 모든 것을 분석하고, 풀리지 않은 문제들을 스스로 해결해야 하는 상황이었다.

1980년대 후반부터는 컴퓨터 통신을 활용하여 컴퓨서브(Compu Serve)와 같은 미국 통신망에 접속할 수 있었다. 아무리 연구를 해도 도저히 풀리지 않는 경우에만 모뎀을 통해서 국제 전화로 접속한 뒤 미국 사람들에게 이것저것 물어보고 자료도 받곤 했다. 그때 전화 요금이 1분에 1천 원, 즉 1시간에 6만 원 정도였는데, 당시 내 월급이 30만 원 정도였으니 엄청나게 비싼 금액이었다.

그 밖에는 모두 직접 분석하고, 컴퓨터의 하드웨어에 불박이로 장착되어 있는 롬바이오스(ROM-BIOS)나 운영체제도 어셈블리어 수준에서 분석할 수밖에 없었다. 컴퓨터 바이러스 분석도 마찬가지로 참고 자료 없이 직접 분석해서 동작 원리를 알아냈다. 그러다보니 처음 컴퓨터 바이러스 관련 서적을 집필할 때는 그동안 공부한 것과 경험을 토대로 해서 참고 서적 한 권도 없이 직접 만든 자료와 기억만을 가지고

완성할 수 있을 정도였다.

그에 비하면 요즘은 오히려 자료의 홍수 시대라고 할 만하다. 이제는 자료를 구하지 못해서 개발을 하지 못하는 경우는 거의 없으니 그런 점에서 지금의 개발 환경이 엄청나게 좋아진 것이다.

그러나 예전보다 나아지기는 했지만, 개발자로서 성장하는 데 불리한 환경은 여전히 남아 있다. 그것은 국내 개발 환경의 특성과 한계 때문인데, 첫째는 개발자 스스로 코더의 성향에 머무르고 마는 마인드 문제와, 둘째는 나이가 들어서 전문가로 남을 수 없게 만드는 사회 통념 및 회사 제도, 셋째는 국내 패키지 소프트웨어 산업의 부진을 들 수 있겠다.

첫째로 개발자들은 일반적으로 프로그래밍, 더 정확하게는 코딩(coding) 자체에 많은 재미와 보람을 느끼는 경향이 있다. 풀리지 않는 수학 문제를 오랜 고생 끝에 풀었을 때 희열감에 사로잡히고, 자신이 만든 프로그램이 잘 작동하는 것을 보면서 자신의 분신처럼 애정은 느낀다.

그러나 이러한 일은 개발자의 보람 가운데 아주 작은 부분일 뿐이다. 미국에서는 이런 수준의 사람들을 코더(coder)라고 부른다. 그리고 많은 프로그램 경험을 통해서 좀더 수준이 올라가면 세부적인 코딩 자체보다는 전체적인 아키텍처(architecture), 프로토콜(protocol) 등 설계에 해당하는 일들을 맡게 된다. 오랜 기간 이러한 일을 하면서 연륜이 쌓이면 비로서 아키텍트(architect)가 된다.

그런데 우리나라는 소프트웨어 산업의 역사가 짧아서이겠지만, 코딩하는 재미에 묻혀 있거나 거기까지가 개발자가 할 수 있는 유일한 이이라고 생각하는 사람들이 의외로 많다. 물론 훌륭한 프로그래머 또는 아키텍트가 되기 위해서는 코더 시절에 탄탄한 기초를 다지는 것 역시 필수지만, 어느 정도 실력이 쌓인 후에는 코더 단계를 뛰어 넘으려는 노력이 필요하다.

둘째로 우리나라에서는 전문가라고 할지라도 나이가 들면 관리직이 되어야 성공했다고 인정하는 사회 통념이나 회사 제도 때문에 외국에 비해서 개발자가 선택할 수 있는 미래가 제한적이며 그 생명도 짧은 편이다. 또한 최근 이공계 기피 현상이 심각한 사회 문제로 떠오르면서, 개발자의 위치 역시 다소 위축된 것처럼 보인다.

개발자 스스로 관리자로 변신하는 것을 두려워해 능력이 닿는 한 개발자로 남고 싶어하지만, 반면에 하루가 다르게 기술적 진보가 이루어지는 상황에서 후배와 동등한 위치에서 새로운 개념과 기술에 적응해야 하는 부담도 많이 느낀다.

개발자들이 코더 단계에서만 머문 채 그 한계를 벗어나지 못하는 셋째 이유는 국내 패키지 소프트웨어 회사 수가 너무 적고 그 규모도 영세하기 때문이다.

개발자가 선택할 수 있는 미래라는 면에서 SI 업체와 패키지 소프트웨어 업체는

상황이 조금 다르다. 패키지 소프트웨어 업체에서 개발자가 선택할 수 있는 길이 상대적으로 조금 더 다양한 편이며, 특히 세계적인 회사들과 경쟁하는 패키지 소프트웨어를 개발하기 위해서는 전문적인 아키텍트가 절실히 필요한 실정이다.

이럴 때 일정 규모 이상의 패키지 소프트웨어 회사가 많이 있다면 각 회사마다 다양한 노하우가 쌓이고 회사간의 제휴나 인력 이동을 통해서 자연스럽게 함께 성장해 나갈 수 있지만, 숫자가 얼마 안되고 영세하다보니 같이 커나갈 수 있는 여건이 형성되지 못하고 있다.

사정이 이러하니 개발자가 나아갈 길을 적절하게 조언해 줄 수 있는 전문가 그리고 시스템이 갖추어진 회사들이 많지 않은 것이다. 이는 우리나라 개발자들의 불행이자, 우리나라 전체의 불행이기도 하다.

이미 몇 차례 강조했던 것처럼, 우리나라가 글로벌 시장에서 앞서 나가려면 지식 정보 산업에서의 경쟁력이 필수적이다. 따라서 우리나라의 미래를 위해서라도 앞서 이야기한 문제점들을 극복하기 위해 다양한 이해 관계자들의 노력이 요구된다.

개발자 자신의 마인드 문제에 대해서는 당연히 당사자의 노력이 요구되며, 소프트웨어 회사 역시 나이가 들어서도 개발자가 계속 남아서 발전할 수 있도록 제도적인 보완책을 마련해야 한다. 또한 국가 차원에서도 소프트웨어 산업 육성과 전문 인력 양성을 위해서 노력을 기울여야 할 것이다.

단, 국가 차원에서 전문 인력 또는 아키텍트를 키우기 위해서는 ‘개발자들이 성장하기 위해서는 두 가지 단계를 거쳐야한다’는 사실을 먼저 이해할 필요가 있다.

처음 단계는 개발자들이 공부를 통해서 성장하는 단계이다. 이 단계에서는 대학이나 학원과 같은 교육 기관에서 교육을 받거나, 소질이 있는 사람은 스스로 독학해서 성장할 수 있다. 하지만 공부만으로는 아무리 천재라 할지라도 어느 정도 이상의 수준을 넘을 수는 없다.

다음 단계로 성자하기 위해서는 큰 규모의 프로젝트를 팀을 이루어서 진행해 보는 경험이 필요하다. 특히 SI 프로젝트보다는 패키지 소프트웨어 개발 프로젝트를 해보는 것이 한 단계 높은 수준으로 제대로 올라갈 수 있다.

SI 프로젝트가 특정한 고객의 요구를 특정한 시스템에 구현하는데 비하여, 패키지 소프트웨어는 불특정 다수 고객의 요구 사항을 수렴하고 이를 다양한 컴퓨터 환경에서 문제없이 동작하게 해야 하는데다가 세계적인 기업들과 동시에 경쟁해야 하기 때문에 프로젝트의 난이도 측면에서도 차원이 다르다. 실무 경험 없이 이론만으로는 절대로 아키텍트가 될 수 없는 법이다.

따라서 정부와 학계에서도 이 점에 대한 이해를 공유하고 인력 양성 계획을 수립해야 한다. 정부에서는 장기적이고 근본적인 마인드를 가지고 단기간의 처방보다는

이 사회가 요구하는 진정한 전문가를 키우고 국가 경쟁력을 강화하기 위한 제도적인 접근을 해야 한다. 학계에서도 업계의 협조와 피드백을 받아서 현대 사회에 맞는 인재를 양성하기 위해 많은 노력을 기울여야 한다.

전문가형 개발자 육성을 위해서는 교육 제도의 개선 노력이 반드시 필요한데, 특히 대학에서 가르치는 것과 현장에서 요구하는 것의 간격을 메우기 위한 노력이 가장 우선적으로 필요하다.

이 과정에서 인도가 좋은 벤치마크 모델이 될 수 있을 것이다. 인도 역시 불과 10여 년 전만 하더라도, 지금처럼 많은 우수한 개발자를 배출하는 소프트웨어 강국이 될 것이라고는 아무도 생각하지 못했다. 그러나 지금은 미국 마이크로소프트 사에서 인도인 개발자들이 나가버리면 회사가 쓰러질 정도라는 이야기가 나올 만큼 인도인의 비중이 높다.

제도적인 측면에서, 특히 교육 제도적인 측면에서 올바르게 접근했기 때문에 가능한 결과라고 생각한다. 우리나라도 인도를 벤치마킹해서 우리 실정에 맞는 제도를 도입하고 시행한다면 경쟁력 있는 개발자들을 많이 배출할 수 있지 않을까 한다.

또한 전문 인력 양성과 함께 산업을 육성하는 일도 동시에 진행해야 한다. 교육 따로 산업 따로 접근하는 것은 개발자의 두 단계 성장 과정에 비추어보더라도 맞지 않는다. 일자리도 없고 먹고 살기도 힘든데 양질의 인력이 나올 리가 만무하기 때문이다.