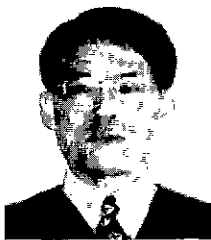


□ 특별기고 □

'리눅스 열풍'과 저작권과 한국 소프트웨어산업의 미래

전자신문사 서현진*



전세계를 강타하고 있는 최근의 '리눅스(Linux) 열풍'은 소프트웨어 혹은 저작권에 대한 일반적인 가치기준과 상식을 깨는 엄청난 사건이 아닐 수 없다. 연일 보도되는 많은 뉴스 가운데서도 특

히 관심을 끄는 것은 리눅스가, 90년대 이후 거의 경쟁상대가 없을 것 같아 보이던 마이크로소프트의 윈도우(Windows) 운용체계(OS)를 제압할 가장 강력한 경쟁상대로 떠오르고 있다는 내용이다.

실제로 마이크로소프트와 함께 90년대 컴퓨터업계를 양분해온 인텔은 한 리눅스 관련 벤처기업에 거액을 투자했고 오라클, 넷스케이프, 코렐, 인포믹스 등 대표적인 소프트웨어 회사들은 잇따라 리눅스 버전을 발표했다. 또한 선 마이크로시스템즈, IBM, 컴팩(디지털) 등 마이크로프로세서 개발사들 역시 적극적인 리눅스 지원 계획을 공표하고 나섰다. 이 같은 움직임은 앞으로도 계속 이어져 머지않아 리눅스가 유닉스나 윈도우NT를 제치고 엔터프라이즈 컴퓨팅 분야의 표준 OS로 자리잡을 것이란 분석도 나오고 있다.

불과 1년여전 그러니까, 97년말 까지만 해도 유닉스를 닮은 리눅스 OS가 정보기술 분야에서 이렇게 높은 관심을 불러올 것을 예측한 이는 별로 없었다. 아직 완전한 성공에 이르렀다고 평가하기는 이르긴 하지만, 소프트웨어가 한국의 미래전략산업이라고 외치는 우리 모두

는 이 '리눅스 열풍'을 통해서 여러 가지를 배워야 할 것 같다.

한 소프트웨어가 시장에서 업계표준으로 자리잡기까지 소요되는 기간에는 어떤 원칙이 있는 것은 아니다. 넷스케이프의 네비게이터(Navigator)처럼 발표된 지 1년도 못돼 세계 시장을 장악해버린 경우가 있는가 하면 윈도우처럼 10여년만(윈도 1.0은 80년대 초반에 발표했다)에 빛을 보게 되는 경우도 있다. 반대로 80년대를 풍미했던 로터스1-2-3(Lotus 1-2-3), 데이터베이스III(Dbase III), 워드퍼펙트(WordPerfect) 등은 순식간에 왕좌를 내주고 개발회사까지 다른 회사에 합병당하는 수모를 겪기도 한다.

물론 어떤 특정 분야에서 왕좌에 오를 수 있는 소프트웨어는 극소수일 수밖에 없다. 실제 소프트웨어 천국 미국에서는 연간 20여만 종의 새로운 프로그램들이 발표되지만 이 가운데 전문 월간지 등에 간단한 리뷰 기사라도 실려 상업적으로 최소한의 개발비 정도를 뽑아내는 경우는 5백여종 미만이며 그나마도 세계적으로 그 명성이 널리 알려지는 제품은 10여종도 안 된다고 한다.

새로운 소프트웨어가 성공하려면 두 가지의 특성 즉, 카리스마적 이미지(강력하고 획기적인 성능)와 남녀노소 누구에게나 또는 어떤 상황에서도 친근하게 다가갈 수 있는 편리성(수평적 혹은 수직 호환성)을 동시에 갖추어야 한다. 실제로 새로 개발되는 다수의 소프트웨어들은 처음에 카리스마적 이미지를 획득하는데 성공하지만 편리함에서 실패하고 있다.

아직은 미완이긴 하지만 리눅스 역시 사용자

*정회원·편집위원, E-mail: jsuh@etnews.co.kr

들에게 카리스마적 이미지와 편리성이 골고루 갖춰진 OS로서 분명하게 자리잡아가고 있는 듯 하다. 하지만 이런 것만으로 '리눅스 열풍'의 원동력을 설명해낼 수 있을까? 우리가 알고 있는 리눅스에 대한 것은 여러 부문에서 많은 것이 잘못 알려져 있다. 이 글의 핵심을 짚어 나가기 위해서는 우선 이 잘못 알려진 내용을 바로 잡을 필요가 있다.

리눅스는 우선 GUI, 컴파일러, 에디터, 문서형성기 등 여러 수많은 컴포넌트들과 각종 유틸리티들로 이뤄지는 OS의 구성에서 커널에 해당되는 부문을 칭하는 이름일 뿐이다. 실제 환경에서 OS가 커널만 가지고는 실행되지 못한다는 것은 두말할 필요가 없다.

'리눅스 열풍'에서 리눅스의 정확한 이름은 GNU/리눅스이다. 마찬가지로 GNU/리눅스의 개발자 또는 창시자도 일반적으로 알려져 있는 핀란드 출신 리누스 토발즈(Linus Torvalds)가 아니다. 그는 단지 커널로서 리눅스를 개발한 사람일 뿐이다. 따라서 리눅스 대신 다른 커널을 사용하는 GNU운영체제가 있을 수 있으며 실제로 유닉스 계통의 마하(Mach)커널을 이용한 GNU/마하가 개발중이기도 하다.

GNU/리눅스는 비영리 단체인 프리소프트웨어재단(FSF)이 주도한 GNU프로젝트에 의해서 탄생되었다(여기서 GNU는 GNU's Not Unix의 줄임 말이라고만 알려졌을 뿐 그 자체가 어떤 글자의 이니셜인지는 밝혀지지 않고 있다). FSF는 현 회장인 MIT교수 리차드 스톨먼(Richard Stallman)에 의해 1983년 창설했



그림 1 GNU 프로젝트 로고

다. 리차드 스톨먼은 GNU프로젝트를 성공적으로 수행한 공로로 스웨덴 왕립 공대로부터 명예박사 학위를 받았으며 GNU/리눅스를 구성하는 C컴파일러와 편집기인 GNU이맥스(Emacs) 등을 개발한 프로그래머이기도 하다.

GNU프로젝트는 FSF의 창설 이념에 의해 그 활동 방향이 잡혀졌다. 리차드 스톨먼은 80년대 초 대학생 시절 자신이 구입한(acquire) 소프트웨어에 대해서 왜 자유롭게 복제(copy)하며 소스코드를 개작(modifying)하거나 변형된 것을 포장(repackaging)해서 분배(redistribution)할 수 없는가에 대해 동료들과 진지하게 토론했다. 그 결과 스톨먼과 그의 동료들은 소프트웨어가 자유롭게 사용될 수 있어야 한다는 결론에 이르고 이런 이념을 실천해보이기 위해 FSF를 창설했던 것이었다.

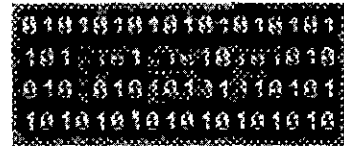


그림 2 FSF 로고

여기서 유의해야 할 대목은 바로 <free>라는 단어의 의미인데 이는 비용 측면에서 소프트웨어를 무제한 단순 복사해서 배포한다는, 이를테면 '공짜(for free)'의 뜻이 아니라 이용 측면에서 자유롭게 변형이나 기능의 개선을 가할 수 있다는 뜻이다. 그런 점에서 프리소프트웨어(free software)는 소스코드가 개방되지 않은채 단지 실행용으로 배포되는 프리웨어(Freeware)와는 근본적으로 다른 것이다.

FSF는 이 같은 <free>의 개념을 바탕으로 전세계 컴퓨터 사용자와 소프트웨어 개발자들이 소프트웨어를 자유롭게 이용할 수 있도록 하는 약정(stipulation)을 만들었다. 이 약정의 명칭이 바로 기존 저작권(copyright)의 개념과 대비된다는 뜻의 신조어인 '카피레프트(copyleft)'이다. GNU프로젝트는 이런 배경에서 1984년부터 시작됐다.

GNU프로젝트 돌입에 앞서 리차드 스톨먼은 GNU선언문(GNU Manifesto)을 발표했는데

그 골자 가운데 눈길을 끄는 대목은 GNU프로젝트에 의해 개발된 소프트웨어에 대해서 적용되는 프리(free)라는 단어의 정의였다. 모두 3가지로 됐는데, 이것을 요약하면 “첫째 GNU 소프트웨어를 복제할 수 있으며 복제한 것을 친구나 동료들에게 분배할 수 있는 자유를 가진다. 둘째, 모든 소스코드에 제한 없이 접근(full access)할 수 있으며 원하는 대로 변화(change)를 가할 수 있는 자유를 가진다. 셋째, 개선된(improved) 버전을 분배할 수 있는 자유를 가지며 그렇게 함으로써 소프트웨어를 공동으로 개발할 수 있는 공동체(community)를 구축할 수 있다”라는 것이었다. 공동체의 구축은 FSF의 창설로 구체화 되었다.

리처드 스톨먼과 FSF가 소프트웨어의 접근 방법과 방식에서 자유의 개념을 강조한 것은 80년대 이후 등장한 거의 모든 소프트웨어들이 저작권에 의해 보호되면서 소스코드에 대한 접근, 변형, 개선, 분배 등이 개발자들만의 고유 권한(proprietary)으로 제한되는 데 대한 반발이었다. 기본적으로 모든 소프트웨어는 70년대 이전의 정신으로 되돌아가 사용자들에 의해 개선되고 배포되어야 한다는 입장이었다.

GNU 프로젝트는 구체적으로 공동체 마인드를 갖고 있는 소프트웨어 개발자를 구성된 공동체가 기존 유닉스와 상향 호환성(upwardly compatible)을 가지는 완벽한 프리 운용체계를 개발하는 작업이었다. 세 운용체계의 개발 목표를 유닉스에 둔 것은 유닉스 역시 60~70년대 프리 정신에 의해 하나의 운용체계로서 완성된 이력을 갖고 있었기 때문이다. 따라서 두 OS는 일종의 같은 뿌리 의식을 공유하고 있다고 할 수 있으며 호환성의 유지는 궁극적으로 수많은 유닉스 자원들과 사용자들과 끌어 오기 위한 전략적 포석이라 할 수 있다.

1990년 GNU프로젝트의 결과가 처음으로 외부에 드러났다. 커널을 제외하고 사용자인터페이스(GUI), 텍스트편집기, C컴파일러·디버거 등 운용체계 구성을 위한 주요 컴포넌트의 개발이 FSF회원들에 완료된 것이다. 그리고 이어 91년, 헬싱키대학 재학생이던 리눅스 토발즈가 리눅스 커널 0.02버전을 공식으로 발표하면서, 비로소 커널과 각종 컴포넌트들을 결합

한 GNU/리눅스 배포판이 등장하기 시작했다.

현재 ‘리눅스 열풍’을 주도하는 것은 이들 배포판들로서 세계적으로 널리 알려진 것들로서는 「슬랙웨어(Slackware)」, 「드비안(Debian)」, 「레드햇(RedHat)」 등이 있다. 이들 제품은 모두 민간 회사들이 CD롬 또는 인터넷사이트(ftp) 등을 통해 무료(비상용) 또는 유료(상용)로 전세계에 제공되고 있다. 비상용 버전은 전문 프로그래머, 전문가, 교육기관 등에 인터넷을 통해 배포되며 모든 것이 사용자 책임, 즉 일체의 사후서비스가 제공되지 않는다. 상용은 기업이나 대규모 조직환경에 적합하도록 구성된 것으로서 공급자와 사용자간 유료 고객 서비스 체계를 전제로 한다.

여기서 바로 GNU선언문에 대한 의미가 잘 나타나 있다. GNU선언문에서는 저작권을 전제로 한 프로그램 판매만이 개발자가 생계를 유지할 수 있는 유일한 방법은 아니라고 지적하고 있다. 또한 저작권을 제한하여 복제를 막는 것만이 소프트웨어사업의 이윤을 창출한다는 생각 역시 고정관념이라는 것이다. 그런 점에서 앞서 언급한 유료 고객서비스는 새로운 이윤추구의 방법의 하나라고 할 수 있다.

GNU 선언문에서는 나아가 개발자는 자신의 창의적 아이디어 즉 저작권을 제한하는 것보다는 경쟁의 원리를 적용해서 아이디어를 발전 계승하는 것이 더 사회적이며 자신의 노력을 보상 받는 길이라고 역설하고 있다. 예컨대 소설의 경우 저작권이 성립되기 이전 고대에서는 어떤 작품이 다른 분야의 작가들에 의해 복제되고 인용돼 왔는데 결과적으로 이같은 과정은 작품을 후대에 지속적으로 전수시키는 데 거의 유일한 방법이자, 또 다른 작품을 탄생시키는 데 매우 유용한 결과를 가져왔다는 얘기다.

가령 인쇄산업이 대중화되지 못했던 우리나라 조선시대에 춘향전을 좋아하던 시어머니가 며느리에게 다음 세대로의 전수를 위해 종이가 헐거워진 책을 그대로 베껴(copy)내라는 숙제를 내렸던 것은 다반사의 일이었다. 이 과정에서 창의력이 풍부하고 강성적인 며느리가 있었다면 맘에 들지 않은 어느 대목에 가서는 자신의 생각대로 개작(modify)을 했을 가능성은 충분히 인정되는 바이다. 그래서 현재 내려오

는 춘향전은 여러 이종(variant)이 존재하게 되는 것이다.

19세기부터 서구사회를 중심으로 정착된 저작권제도도 원래 작가의식을 고취시키기 위해 고안되었으며 작가에게 이윤추구의 수단으로까지 활용되지는 못했다. 근대 이후 인쇄술의 발달과 함께 이 제도가 도입됐을 때 실질적으로 작가의 이윤추구 측면에서 저작권은 그다지 큰 의미를 갖고 있지는 않았다. 복제라는 것은 누구나 함부로 이용할 수 없었던 인쇄기를 이용해서만이 가능한 것이었기 때문이다. 저작권 제한 역시 그 자체가 사회적으로 더 큰 이득이 된다고 여겨졌기 때문이었지만 큰 영향력을 발휘한 것은 아니었다.

그러나 오늘날 소프트웨어의 경우는 저작권 개념이 탄생한 1백여년 전과는 완전히 다르다. 예를 들면 프로그램이 거창한 인쇄기나 복사기를 사용하지 않고도 이웃간에 간단하게 복제돼 배포될 수 있다는 점, 또한 소설처럼 단순히 읽거나 즐기기 위해 복제되는 것이 아니라 점, 나아가서는 복제 대상이 소스코드와 오브젝트코드로 구분된다는 점 등이 바로 그것이다. 이런 특성들을 무시한 채 1백여 전의 저작권 개념을 소프트웨어 프로그램에 그대로 적용한다는 것은 사회 전체에 정신적 물질적으로 해를 끼치는 것이며 법적 허용 여부를 떠나 사용자들의 컴퓨터 이용을 제한하는 결과를 가져올 수 밖에 없다는 주장이다.

이제 이 글의 결론을 맺어보자. 지나친 비약 일지는 모르겠지만 독자 여러분들은 이제까지 글을 읽어 오면서 필자가 의도하는 바를 어느 정도는 짐작했을 줄로 믿는다.

한국의 소프트웨어산업 발전을 위해 유가의 패키지들을 당장 무료화시키고 프로그램의 복제를 허용해야 한다는 논리는 절대 아니다. FSF나 GNU선언문의 취지 역시 처음부터 그런 의도를 가진 것은 아니었을 것이다. 하지만 소프트웨어를 미래전략산업으로 여기고 있는 정책당국이나 업계, 학계 입장에서 볼 때 FSF의 이념이나 GNU선언문과 같은 발상의 전환이 절실하게 필요한 때인 것만은 분명한 것 같다.

우선 정책적인 측면에서 보면 해답은 분명해진다, 사실 대다수 소프트웨어 프로그램들이 저작권에서 기반해서 독자화의 길을 걷고 있던 80년대 상황에서 리차드 스톨먼 같은 걸출한 인물이 나올 수 있었던 것은 선진국 대부분이 오래 전부터 자국 내 프로그램보호법에서 역공학(Reverse Engineering) 조항을 허용해왔기 때문이었다. 역공학이란 특정 프로그램의 소스코드를 분석해서 이를 토대로 새로운 프로그램을 개발해내는 기법. 부작용이 전혀 없었던 것은 아니지만 소프트웨어 발전에 지대한 공헌을 한 것도 사실이었다. 한국의 프로그램 보호법이 제정된 지난 87년 이후 여러 번의 개정 과정에서 역공학 조항의 삽입이 번번히 무산된 것과 비교하면 좋은 예가 아닐 수 없다.

또 하나의 사례를 들어보자. 지난해 부도 위기에 직면했던, 한국을 대표한다는 한 소프트웨어 패키지 업체 대표는 경영위기의 첫번째 원인이 프로그램의 무단 복제 때문이라고 변명을 한적이 있다. 문제는 많은 사람들이 이런 변명을 액면 그대로 받아들인다는 것인데 사실은 그렇지 않다. 경영위기에 봉착하기 2~3년 전 이 회사가 개발한 워드프로세서가 한국의 자존심으로 불리우며 업계 표준으로 부상하자, 업체는 이제품의 소스코드를 공개하여 보다 풍부한 애플리케이션들을 개발할 수 있도록 해달라고 요구한 적이 있었다. 하지만 이 회사는 저작권과 재산권을 들먹이며 “말도 안되는 소리”라고 일축해버린 적이 있다. 우연인지는 모르겠지만 GNU프로젝트에 의해 GNU/리눅스가 본격적으로 전세계 일반 사용자들에게 알려지기 시작한 것은 것은 바로 이즈음이다.

사례는 이밖에도 수없이 많다. 문제는 ‘리눅스 열풍’이 보여주듯 컴퓨터환경과 소프트웨어 개발환경이 엄청나게 변화하고 있다는 점이다. 우리 모두 ‘리눅스 열풍’이 주는 의미를 다시 한 번 되짚어 보아야 할 때이다. 그리고 늦었지만 발상의 전환이 필요한 때이다 아직도 소프트웨어를 미래 전략산업으로 여기고 있다면 말이다.