

위대한 프로그램 L^AT_EX

조성민 a.k.a smcho¹

위대함의 원천

제가 관심있는 것중의 하나는 좋은 것과 그렇지 않은 것의 차이가 무엇때문에 발생하는가 입니다. 왜 능력있는 사람과 그렇지 않은 사람, 초일류 회사로 발전해 나가는 회사와 그렇지 못한 회사, 더크게는 선진국과 후진국의 차이가 왜 발생하였을까 하는 것입니다. 1000년 전에는 번성했던 국가가 지금은 쇠잔해 있고 100년전에는 번성했던 회사가 지금은 없어져 버리고, 10년전에 뛰어났던 친구가 지금은 왜 무능한 삶을 살고 있을까 저는 궁금해 하고 있었습니다. 어째서 이러한 차이가 발생하였을까? 저만 궁금했던 것이 아니었는지 이에 대한 많은 책들이 나와 있어서 좋은 지식들을 많이 얻을 수 있습니다.

제가 알아낸 것중의 하나가 차이를 만드는 여러가지 중에 중요한 요소가 적당히 ‘좋은 것’을 추구하는가, 아니면 힘들지만 ‘위대한 것’의 추구하는가입니다. 그냥 적당히 좋은 것에 안주하는가, 좋은 것에서 더 좋은 것 궁극적으로는 위대한 것으로 나아가는가. 달리 말하면 그러한 위대함을 추구하는 행위를 좋고 바람직하게 가치로 여기는가의 개인적, 집단적 국가적 태도가 결국은 엄청난 차이를 발생시킨 다는 점입니다.

학계에서도 페이퍼를 몇개 생산했는지에 고심¹하는 그저 그런 좋은 모양을 추구하는 학자과 학계에 영향력을 미치고 싶어하고 궁극적으로는 위대한 업적을 추구하는 학자는 처음에는 별 차이가 없겠지만 결국에는 엄청난 차이로 발전하게 되는 것을 많이 봅니다. 세일즈 에서도 물건 몇개 팔아버리는 그저 그런 좋은 수익을 내려고 노력하는 사람과, 그 분야에 위대한 사람이 되려고 인적, 물적 시스템을 차근 차근히 구축하는 사람은 결과적으로 엄청난 차이가 날 것입니다. 더 숭고하고 위대한 가치를 추구하는 사람들이 그렇지 않은 사람 보다 더 많은 국가가 그렇지 못한 국가보다 결국 더 위대한 가치를 내는 것은 어떻게 보면 당연한 논리라고도 하겠습니다.

그러면 그 위대함의 원천은 무엇일까요? 저는 ‘근본(Fundamental)’에 대한 이해 및 근본을 바탕으로한 구축’이라고 생각해 봅니다. 페이퍼의 갯수가 아니라 페이퍼의 질(Quality)가 학계에서의 핵심이라는 근본적인 이해 및 공감미 없이는 페이퍼 갯수만을 양산하는 노력을 중지하기는 힘들 것이겠지요. 또 세일즈라는 것이 결국은 사람을 얻는 것이라는 근본적인 이해없이 장사꾼의 모습에서 벗어나기는 힘들 것입니다.

위대한 L^AT_EX

이런 의미에서 저는 L^AT_EX 이 위대한 프로그램이라고 생각합니다. L^AT_EX은² 컴퓨터의 신이라고 불리는 컴퓨터 과학자 이자 수학자인 도널드 크누스 박사께서 심혈을 기울여 연구하여 만들었고 그 만큼은 아니지만 컴퓨터 과학계에 수많은 업적을 남긴 레슬리 램포트 박사가 쓰기 편하

¹논문을 써보신 분은 하나의 아이디어로 상당히 많은 숫자의 페이퍼를 양산할 수 있다는 비밀아닌 비밀은 다 알고 계실 것입니다.

²앞으로는 특별한 차이가 없는한 L^AT_EX과 T_EX을 같은 의미로 사용하겠습니다.

게 발전시키고 또, 그만큼은 아니지만³ 상당히 뛰어난 CERN⁴의 과학자들이 유지하고 보수하고 있는 프로그램입니다. 현재까지 이정도 수준의 소프트웨어가 어디에 더 있을까요?

L^AT_EX은 훌륭한 사람이 만들고 유지해서 훌륭한 것이 아니라 근본을 가지고 있는 프로그램이기 때문에 훌륭한 것이라고 생각합니다. 다른 사람의 말을 빌리면 구텐베르크의 업적에 비길만한 엄청난 일입니다. T_EX의 개발자인 크누스 박사께서 T_EX을 개발하시기 전에 전세계의 문헌을 뒤져 이제까지 세계 각지의 조판 역사를 상당히 오랜 기간동안 연구하여 통달한 다음에 개발에 임했다고 하는 것은 범인으로서 놀란 입을 다물기 힘들게 하는 부분입니다.⁵ 우리가 뉴턴의 프린키피아를 원전으로 읽는 것은 아주 힘든 일입니다. 결국 범인이 뉴턴에 대해서 아는 것은 후세들이 그의 프린키피아를 아주 알기 쉽게 표현해 준 책들을 통해서입니다.⁶ 그러나 우리는 최소한 T_EX에 관해서는 도널드 크누스 박사의 업적 그대로를 즐기고 있습니다. 어떻게 생각하면 놀라운 일이 아닐까요?

이렇기 때문에 Bill Gates 도 도널드 크누스 라면 거의 찬사에 가까운 반응을 보이는지 모르겠습니다. Writing Solid Code 를 쓴 마이크로 소프트 워드 개발자인 스티브 머기어는 책의 서문에서 T_EX 책을 읽고서 도널드 크누스 교수의 자신감에 놀랐다는 이야기가 나옵니다.⁷ 저는 이 책을 보고 오히려 다시 놀랐습니다. 워드 개발자도 T_EX 알고리즘을 참고했겠구나 하고요. 한편 당연하다고 생각했습니다. 이보다 더 책의 편집에 관해서 설명한 위대한 책이 인류역사상 있겠습니까? 또 그 내용을 책으로 살실히 소개하는 경우가 있겠습니까? 또 자신의 작품을 다른 사람이 상업화 하는 것을 장려하는 경우가 또 얼마나 있겠습니까?

그러나 현실은 이러한 L^AT_EX이 처음에 보기에는 거의 알려져 있지 않고 사용되어지지도 않고 또 사용하기에도 힘든 것 같이 보입니다. 몇번 시도해 보다가 너무 어렵다고 관두는 일이 부지기수이고 아예 시도조차도 해보지 않는 것이 대부분입니다. 그러나 세상은 참 재미있어서, 입에 달고 얻기 쉬운 것에 중독되는 경향이 있습니다. 제가 경험하기로 하나는 결코 좋은 것을 그냥 주시지 않는 것 같습니다. 항상 어느 정도의 시련과 고통의 뒤에 숨겨 두시는 것 같습니다. 마약과 같이 엄청난 기쁨을 손쉽게 주는 것들의 말로를 생각해 보면 쉽게 얻는다는 것이 결국 인간에게 항상 좋은 결과만을 주지 못한다는 것을 생각하게 합니다.

컴퓨터도 같은 카테고리에 들어간다고 생각합니다. 처음 보기에 마우스로 몇번 클릭해서 뭔가 결과를 내는 것이 더 좋아 보이지만 그것은 진입장벽이 낮다는 것이지 무엇인가가 더 쉽다는 것을 보여주는 것은 아니라고 생각합니다. 어느 책에서 보니까 UNIX 의 여러 복잡하게 보이는 명령어도 체계적으로 몇주정도 잘 교육하였더니 60년 할머니도 grep 명령어가 없는 시스템에서 고통을 느끼더 라군요. 처음에 마우스만 클릭하던 사람은 십년이 지나면 마우스를 더 빨리 능숙하게 클릭하게 됩니다. 그러나 UNIX를 사용하는 사람은 각각의 기능을 통합하여 엄청난 생산성을 보여줄 수 있게 됩니다.⁸ 마치 L^AT_EX이 그러하듯이요. 워드를 사용하면 적당한 문서를 적당히 잘 만드는 것은 가능할지 모르지만 10년이 지나도 거기에서 거기가 아닌가요. 그러나 L^AT_EX은 어름 어름 배운 것이 10년이 쌓이면 상당히 수준 있는 문서를 손쉽게 만들 수 있는 수준이 됩니다.⁹ 그 깊이에 있어서 적당히 좋은 것과 위대한 것의 차이를 느끼는 부분이라고 생각합니다. 워드는 그 깊이를 빨리 느낄 수 있지만, L^AT_EX은 쓰면 쓸 수록 깊이를 느끼니 말입니다.

³도널드 크누스에 비해서 그렇다는 것이지만 모두 엄청나게 훌륭한 과학자들입니다.

⁴스위스에 위치한 세계적인 연구소, www 이 발명된 곳

⁵도널드 크누스 박사 당신 자체가 어렸을때 부터 아버지와 직접 손으로 조판을 했던 경험이 있다고 합니다. 이분이 자신의 책 The Art of Computer Programming 의 1판 품질에 불만을 품고 T_EX을 만들었다는 이야기는 아주 유명한 일화입니다.

⁶혹은 뉴턴의 이론을 실제로 적용한 것들을 통해서입니다.

⁷T_EXBook 서문에서 버그를 발견하면 상금을 주겠다는 이야기를 보고 한 말입니다.

⁸저는 UNIX 와 L^AT_EX이 인류 역사상 가장 위대한 소프트웨어라고 생각합니다.

⁹물론 개인에 따라서 많은 차이를 보일 것입니다.

김치찌게의 법칙

요즘 일본에 김치가 인기입니다. 전에는 냄새난다고 쳐다도 안보던 사람들이 이제는 다이어트 건강식이다하여 일본 어디를 가도 김치가 없는 곳이 없습니다. 누군가 그러시더라고요 이 김치를 먹는 일본인중 맛에 득도하여 김치찌게를 먹을 수 있는 사람이 전체의 5% 정도 된답니다. 어쩐지 일본에서 최고령 에베레스트 정복자가 나왔는데¹⁰ 이 사람 이야기를 일본 신문에서 본적이 있습니다. 이 할아버지의 건강 비결이 바로 김치찌게 였다고 할아버지가 이야기를 하더군요. 여하간 이 5%라는 수치를 저는 이렇게 해석했습니다. 특정한 어떠한 것을 사용하는 인구 중에 그 진정한 맛을 느끼는 사람이 5% 정도 된다. 이 법칙을 ‘김치찌게의 법칙’이라고 이름을 붙여 보겠습니다.

재미있게도 여러가지 수치가 김치찌게의 법칙을 보여주는것 같습니다. 컴퓨터 사용자중 Linux 사용자가 한 4% 정도된다고 합니다. 약간 논리에 억지가 있지만 컴퓨터를 제대로 해커 기질을 갖고 사용하는 사람, 즉 진정한 컴퓨터의 맛을 느끼는 사람이 Linux 사용자라고 할때¹¹ 어느 정도 수치가 맞지 않나 싶습니다. 우리나라는 워낙 T_EX사용 인구가 작아서 그렇지만 미국이나 일본, 독일의 경우 컴퓨터를 이용해서 글을 작성하는 사람의 비율도 김치찌게의 법칙이 적용되지 않을까 생각해 봅니다. 생각을 더 발전시켜 보면 음악을 듣는 사람중에 클래식을 듣는 사람도 비슷한 비율이지 않을까 싶습니다.

일 때문에 외국에 몇번 나가 보았습니다. 이 때 선진국¹²의 특징 중의 바로 이렇게 맛을 알아버린 사람들이 많다는 것입니다. 제가 보기에 쉽고 편한 것이 아니라 깊이가 있고 아름다움에 빠져서 그것을 추구하는 사람이 의외로 많았습니다. 비록 모든 사람이 그 위대함을 추구하지는 않는다고 하더라도 위대함을 추구하는 사람에 대해서 사회는 한없는 존경을 보내는 것을 저는 많이 목격하였습니다.

놀라움의 연속 L^AT_EX

L^AT_EX은 기능적인 놀라움 뿐만 아니라 지적, 사회적 놀라움의 연속이었다는 것을 기억합니다. 어느정도 수준의 교육받은 프로그래머들이라면 한번은 읽었을, 아니면 읽고 싶었을 아니면 읽어 보라는 소리를 들었을, 그도 아니면 대형 서점의 컴퓨터 코너에 꾸준히 자리를 잡고 있는 명저중의 명저 ‘The Art of Computer Programming’¹³이라는 책의 저자와 T_EX의 설계자가 동일하다는 것은 저에게 처음의 놀라움이었습니다. 마치 T_EX그 자체처럼 알면 알수록 대단한 사람이라는 평가를 넘어 도대체 이런 사람이 있을 수 있는가 하는 생각을 하게 하는 도널드 크누스 교수.

이 분이 자기의 천재성을 그 많은 시간을 들여서 완성하고, 자신이 보기에 버그가 없다고 공표하고 버그를 찾는 사람에게 상금을 주겠다고 책 서문에 썼을때 한번이라도 심각한 프로그램을 만들어 본 사람이라면 그것이 얼마나 엄청난 일인가를 느꼈을 것입니다. 도널드 크누스 교수는 T_EX을 만드는데 예상보다 상당히 많은 시간이 소요되어 ‘The Art of Computer Programming’ 완간이 늦어진 것에 대해서 서운해 하지 않았다고 합니다. 그만큼 애정이 가는 좋은 소프트웨어라는 의미이겠습니다.

또한 제가 명저라고 생각했던 많은 책들이 L^AT_EX을 이용해서 조판되었다는 것을 알았을때 또 한 새로운 기쁨을 느꼈습니다. 영미권에서 이공계의 유명한 교수들의 책은 많은 부분 그 교수들이 책을 조판합니다. 정확하게 말하면 L^AT_EX이 조판을 해주는 것이지요. 어렸을때 영어로 된 책

¹⁰제 기억으로는 70세에서 75세 사이였던것 같습니다.

¹¹이말에 트집을 잡고 싶어하시는 분은 약간 논리에 억지가 있다고 한 제말을 기억하시고 참아주시기 바랍니다.

¹²제가 주로 경험해본 미국,일본,독일의 경우로 한정하겠습니다.

¹³제가 알기로는 Bill Gates가 책표지에 추천을 한 유일한 도서가 아닌가 싶습니다.

들의 야릇한 폰트들이 실은 Knuth 의 METAFONT였다는 것은 신선한 놀라움이었습니다. 지금은 조금 달라졌지만 이전에 세계적인 논문들의 경우 거의 대부분 L^AT_EX의 스타일 파일을 제공하고 이것에 맞추어서 논문을 전달하라고 하는 요구사항을 봤을때 다시 놀랐습니다. 그럼에도 불구하고 국내 학회에서는 대부분 아래한글로 논문을 쓰는 것을 보고 또 놀랍니다.¹⁴

프로그래밍을 업으로 하면서 깨달은 것이 프로그래밍이라는 것이 사실은 컴퍼넌트들을 모아 서로 연결하여 만드는 것이라는 원리를 깨닫고 기뻐했던 적이 있습니다. 그런데 T_EXBook 에서 그가 T_EX을 만든 언어를 Web 이라고 한 것이 사실은 T_EX의 여러 primitive 를 서로 연결하여 T_EX을 만들었기 때문에 붙여졌다는 것을 읽고 다시한번 놀랐습니다. 첫번째는 그가 T_EX이라는 프로그램을 만들기 위해서 Web 언어를 만들었다는 것에 놀라고, 도널드 크누스 교수가 LR 파서의 창시자 인것을 기억하고 또한번 놀랐으며 역시 도널드 크누스 교수는 프로그래밍에 정통한 사람이라는 것을 느끼고 다시한번 놀랐습니다.

L^AT_EX을 사용하여 문서 및 책을 작성할때 많은 사람이 물어 봅니다. 그게 뭐냐, 불편해서 어떻게 쓰냐. 그러나 워드 프로세서를 가지고 한번이라도 제대로 된 책을 직접 만들어 본 경험이 있는 사람은 한번은 신기해 하면서 놀랍니다. 아니, 이런 프로그램이 있었냐? 그림을 집어넣어도 문제가 되지 않고 인덱스 제대로 만들어 주고, pdf 파일에 날개도 만들어 주고 이리 저리 조합해서 새로운 형태의 책을 만들 수 있다는 것이 가능한 것을 보고 좋아합니다. 즉, 한번이라도 책을 만들면서 Word 나 아래한글에 테인 사람이¹⁵ L^AT_EX의 기능을 보고 그 훌륭함을 알아차리는 것을 보고 한번 놀랍니다. 그러나 그럼에도 불구하고 그 사람이 L^AT_EX을 사용하기 보다는 다른 도구를 사용하는 것을 보면 저는 다시 놀랍니다.

제가 좋아하는 저자들이 많은 O'reilly 출판사는 책을 보기 편하면서 아름답게 만들어 줍니다. 여기에서도 XML 을 기반으로 한 DocBOOK 을 이용해서 책을 쓰고 이를 T_EX으로 변환하여 처리한다고 합니다.¹⁶ Open Source 의 창시자인 Richard Stallman 및 그의 그룹도 infoT_EX이라고 하는 독자 포맷을 가지고 문서를 만듭니다. 즉, GNU 에서 나오는 모든 ps 나 pdf 문서는 T_EX을 이용해서 만들어진 것입니다. 또한 DoxyGen 이라고 하는 소프트웨어를 문서화 하는 유명한 툴에 기본 출력포맷중의 하나가 L^AT_EX입니다. T_EX이 이렇게 여러분야에서 실제로 아름다운 책을 만들고 있고 알게 모르게 사람들의 손에 전달된 다는 것에 놀랍니다.

나를 구해준 L^AT_EX

대학교 시절에 처음으로 책¹⁷을 만들기 시작하면서 책과 저와의 인연이 시작되었습니다. 아무것도 모르던 시절, 그냥 만들었던 책이었지만 책을 만드는데 내용에 고민하였던 것 만큼이나 워드 프로세서¹⁸때문에 발생한 문제를 해결하느라고 고민하는 시간이 많았다는 비이성적인 상황을 누구나 다 그렇게 하는 것으로 넘길정도로 무식했던 것 같습니다.

논문을 쓰게 되면서 L^AT_EX을 처음 알았습니다. 제가 있었던 학교는 많은 사람이 L^AT_EX을 사용했었고 논문을 제출하는 학회도 대부분 sty 파일을 규정하여 줄만큼 L^AT_EX을 주로 사용하는(사용해야 하는) 분위기가 강했습니다. 물론 워드를 쓰는 사람도 McIntosh 도 많이 사용했지만 말입니다. IEEE 논문중에 Consumer Electronics 같은 Transaction은 비교적 투고하기 쉬우면서도 실제의 논문의 질도 그렇게 높지 못합니다. 게다가 논문의 출력 품질도 온 세상의 모든 워드프로세서의 집합이라고 보일만큼 잡다하고 통일성이 없습니다. 즉, 싸구려 같이 보입니다. 그러나 가장

¹⁴국내 학회가 해외 학회에 비해서 수준이 낮다고 꼬집으려고 하는 것이 아니라, 국내에서는 T_EX을 이용해서 논문을 제출하지 않아도 되는 곳이 많다는 것을 말씀드리고 싶었던 것입니다.

¹⁵말도안되는 이유로 고생한 사람이

¹⁶정확한 이름은 잘 모르겠습니다.

¹⁷중급자를 위한 터보-C 완성, 달리만들 제작

¹⁸아래한글

채택되기도 힘들면서 권위가 있는 Transactions on Communication 은 내용도 뛰어날 뿐만 아니라 하나 같은 포맷을 사용하고 수식도 미려하고 예뻐했습니다. 친구가 이야기하기를 이곳에 두고 하기 위해서는 L^AT_EX으로 작업해야 한다고 하더군요.¹⁹ 이때 부터 L^AT_EX은 고급이지만 쓰기 힘들고 하는 선입견이 생겼던 것 같습니다.

저도 세간의 사람들과 크게 다르지 않아서 L^AT_EX을 처음 사용할때에는 이해를 하지 못했습니다. 특히 아래한글을 이용해서 꽤 두꺼운 책도 만들어본 경험이 있어서 책은 그냥 만들면 되는 것이라는 인식을 가지고 있었던 것 같습니다. 그러나 해외에 논문을 투고할때 아무래도 L^AT_EX을 이용하지 않으면 문서의 품질이 너무 떨어져서 어쩔 수 없이 L^AT_EX을 사용할 수 밖에 없었던 것 같습니다. 영어 논문이야 L^AT_EX으로 만든다고 하지만 제가 L^AT_EX으로 처음 작업했을 당시²⁰에는 한글이 그렇게 좋지 않고 또 PK 폰트를 사용했기에 pdf 로 만들어서 보면 아주 조악하게 보여서 실제로 사용하기가 쉽지 않았습니다. 또 윈도우즈에서 사용할 수도 없었습니다. 이렇게 해서 조금씩 L^AT_EX과는 멀어져 가게 되었던 것 같습니다.

이후 창업하여 엔지니어링을 직업으로 만큼 많은 양의 문서를 그것도 여러가지 언어로 만들어야 할 필요성이 많이 생겼습니다. 특히 매뉴얼의 경우 몇가지 언어로 만들어야 하나까 어떻게 하던 해결책을 찾아야 했습니다. 처음 시도한 방법은 컴퓨터를 몇개 사다 놓고 각각의 컴퓨터에 영어, 한글, 독일어, 일본어 OS 를 깔고 각각의 언어에 해당하는 MS-Word 및 pdf를 사용했습니다. 나름대로 처음에는 잘 되었는데, 문서가 복잡해 지고 외부에 번역을 시키는 일이 생김에 따라서 다양한 문제가 발생하게 되었습니다.

가장 이해가 안가는 문제는 영어에서 작성한 워드가 한글에서 못읽는 일이 발생하는 것입니다. 왜 그런지 어떨때 그런지 아무도 모릅니다. 또 한글 OS 에서 영어만으로 이루어진 문서를 만들어서 이를 영문 OS 에서 읽으면 못읽는 일이 발생합니다. 문제는 공란을 넣을때 이것이 영어가 아니라 한글로 들어간다는 것인데 이것도 왜 그런지 아무도 모릅니다. 이 문제를 해결하기 위해서는 문서의 모든 공란을 체크해 보아야 합니다. 윈도우 헬프 파일을 만들때 일단 html 으로 변환을 해야 하는데 여기에도 문제가 생깁니다. 왜 그런지는 아무도 모릅니다. 하나씩 다 읽어서 점검해야 합니다.

어느날 처음 책을 만들었을때의 공포가 되살아 났고, 그 공포는 사실로 드러났습니다. 저희 시스템을 제어하는 라이브러리 루틴을 워드로 만들었는데 매일 조금씩 분량을 늘려 나갔습니다. 물론 그림도 조금 있었구요. 어느날인가 20K 짜리 문서가 1M가 되더니 다음에 10M 가 되는 일이 발생했습니다. 한일은 그림 하나 더 넣은 것이었습니다. 로딩하는데 5분, 그나마 반타작은 로딩하다가 죽어 버리는 일이 반복되었습니다. 문서의 내용이 그 문서를 작성하는 행위보다 중요하다는 듣지 않는(들을 수 없는) 마이크로소프트에에 항의를 하면서 안타까워 하며 고통스러워 했지만 이미 늦었습니다. 거의 1주일을 그문제로 초죽음이 될때까지 고생을 했던 것 같습니다.²¹

그제서야 조금의 문제없이 문서를 만들었던 L^AT_EX이 생각이 났습니다. 그래서 그 다음에 약 80장정도 되는 레포트를 만들때 L^AT_EX을 도입하기로 감히 모험을 했습니다. 그림이 조금 많아서 잘못하다가가는 또 지뢰밭을 걸겠구나 하는 공포가 모험을 하게 만들었습니다. L^AT_EX을 처음 까는 것 부터 모르는 것이 너무 많아서 고생을 했지만 KTUG 분들이 너무나도 자세히 알려주셔서 실제로 거의 문제가 되지 않았습니다. 모르는 것을 구체적으로 기록하여 게시판에 올리면 거의 90% 정도는 해결을 해주셨기 때문입니다. 덕분에 오랫동안 문서를 작성하는 일에만 신경을 쓰고 그것이 어떻게 파일화 되는 가에 신경을 쓰지 않아도 되는 당연하지만 꼭 그렇지 만은 않은 일을 할 수 있었습니다.

물론 제가 했던 환경에서 문서 작업을 하시는 분들이 많지는 않을 것입니다. 즉, MS-Word 가 제공하는 지뢰밭을 경험하지 않아도 되는 대부분의 도메인에서의 문서작업 이 L^AT_EX의 문법을

¹⁹몇년전 이야기 입니다. 지금은 어떻게 바뀌었는지 잘 모르겠습니다.

²⁰90년대 초

²¹이 문제도 새로운 워드 버전에서 해결되었을 것이라고 믿지만 확인하고 싶지는 않습니다.

공부하는 고통을 감당할 필요가 없게 만든다고 생각합니다. 그러기에 MS-Word가 잘팔리는 이유고 많은 (대부분의) 사람이 워드를 사용하는 이유일 것입니다. 저도 동일했구요. 그러나 실제로 책을 만드는 행위를 할때 L^AT_EX은 월등히 뛰어난 효용성을 갖는 것은 제가 경험한 사실입니다. 그러나 이것 이외에도 L^AT_EX은 더 많은 장점이 있습니다. 문서에 커멘트를 달 수도 있고요, CVS 혹은 SubVersion 을 이용해서 버전 관리도 가능하고요, include 기능을 이용해서 문서를 여러 형태로 조합하는 것도 가능합니다. Web 과 연동할 수 도 있구요. 동시에 많은 사람이 문서를 만드는 것도 가능합니다. 물론 워드도 같은 기능을 할 수 있다고 주장하고 설득하지만 원래 그런 기능을 내재하고 있는 것과 그것이 가능하도록 조작한 것과는 많은 차이가 있다고 생각합니다. 최근에 저는 도은이 아버님의 도움으로 일본 노래의 가사를 한글로 대역하여 예쁘게 출력할 수 있었습니다. MS-Word에서는 결코 시도도 해보지 않았을 일을 말입니다.

KTUG

이제까지 저는 ‘한국인은 개개인은 똑똑한데 같이 뭉쳐서는 잘 일을 못한다’라고 배워 왔습니다. 그런데 외국에 나가면 평가가 조금 다릅니다. ‘한국인은 고난을 많이 당해서 그런지는 모르겠는데 잘 뭉친다.’ 물론 2002년 월드컵에서의 강한 인상이 그러한 인상을 주었는지는 모르겠습니다만 여하간 저는 한참 동안 어느것이 우리의 옳은 모습인지 헷갈렸습니다. 그러던 것이 KTUG 의 여러 분들을 보면서 생각이 정리가 되었습니다. KTUG 은 ‘진정으로’ 도와주는 집단입니다. L^AT_EX이 좋아서 모였기에 가능하다고 생각합니다. 2002년 월드컵도 좋아하는 일을 함께 했기에 가능했듯이 KTUG 도 여러가지 일을 하시는 분들이 문서를 아름답고 품위있게 만들어 주는 도구인 L^AT_EX의 위대함을, 아름다움을, 편리함을 공감하고 느끼면서 한마음으로 도와주시는 곳임을 느낍니다. 이면우 교수가 말했듯이 ‘신바람’이 나서 시너지를 내는 한국인의 전형을 보여주는 곳이 바로 KTUG 이라고 생각합니다.²²

편의성이라는 측면에서 KTUG의 여러 분들은 L^AT_EX의 발전에 있어서 정말로 큰 일을 해주셨다고 생각합니다. 완벽한 한글 지원, 유니코드 지원, 한글 pdf, 한글 책갈피 이전에는 상상도 못했던 일이 지금 가능합니다. 제가 KTUG 이 없었을때 어름 어름 했던 일들은 고통의 연속이었고 그 결과도 그렇게 좋지 못했습니다. 그러나 지금은 KTUG Collection 한장이면 10분 만에 완벽한 한글 문서를 만들 수 있습니다. 실제로 제가 여직원 한명에게 셋팅하고 교육하여 제 일을 도와주도록 하는데 걸린 시간은 2시간 남짓입니다. 생산성의 향상은 이전에 비하면 놀랍습니다.

L^AT_EX의 미래

요즘은 미래를 예측하는 것이 유행입니다. 사람들이 모두 관심을 가지고 있고 또 못맞추더라도 본전이고 혹 맞추면 칭찬을 듣기 때문일 것입니다. 저도 L^AT_EX에 대해서 다음과 같은 예측을 해봅니다.

100년 후에도 사람들이 L^AT_EX을 사용할 까요? 저는 거의 100% 그렇다고 생각합니다. 그 때 즈음해서는 L^AT_EX5 정도가 나오지 않을까요? 100년후에 매크로 소프트라는 회사가 메가 울트라 워드 프로세서를 선보여서 시장 점유율이 90%가 되더라도 L^AT_EX은 버그 없는 안정된 소프트웨어로 굳건히 사용될 것이라고 생각합니다. 물론 제가 지금 만드는 이 문서도 그때에 전혀 문제없이 동일한 출력 결과를 내어 줄 것입니다. 물론 그 문서의 결과를 보는 포맷은 수퍼 기가 4D pdfx 가 되겠지만요.

²² 물론 저만해도 짜증 나는 일도 가끔은 있지만, 훨씬 더 짜증 날 일을 잘 이해해 주시면서 도와 주시는 분들이 대부분이기에 오히려 감사하면서 살고 있습니다.

100년 후에도 MS-Word 를 사용할 까요? 그것은 모르겠지만 당시에 사람들이 손쉽게 문서를 만들 수 있다고 생각되어지는 무슨 도구를 계속 사용할 것입니다. 지금 제가 만든 문서를 그 때에도 읽을 수 있을 지는 잘 모르겠습니다. 아마 변환기 몇개를 돌리면 가능하겠지요.

T_EX을 사용하는 인구가 늘어날까요? 아마도 아닐 것입니다. 아마 ‘김치찌개의 법칙’ 정도의 숫자가 꾸준히 다른 워드 프로세서의 능력없음을 지탄하면서 사용할 것입니다. 그 숫자는 크게 늘지도 않겠지만 결코 크게 줄지도 않을 것입니다. 위대함을 볼 수 있는 사람은 소수이지만 없어지지 않는 테니까요. 하드웨어의 성능이 지속적으로 발전해서 L^AT_EX에 Wysiwyg 의 기능을 실시간으로 입히는 것이 가능해지면²³ 조금 더 사용자가 많아 질 수도 있을 것입니다. 국내에서 T_EX을 사용하는 인구가 늘어날까요? 저는 아니라고 봅니다. 요즘의 세태는 적당히 좋은 것을 추구하는 정도가 아니라 입에 달고 쓰기 쉽고 버리기 쉬운 것을 좇는 것 같이 보이기 때문입니다.²⁴ 대학에서도 논문을 쓸때 워드로 예쁘게 그리려고 노력할 것이고요, 책은 계속해서 아래한글로 만들거나 조금 돈을 더 들여서 Quark Express 를 사용하겠지요. 그림이 덕지 덕지 붙은 정신없는 책들을 만들때 말입니다. L^AT_EX은 별로 배우고 싶어하지도 않고 배우려고 하지도 않을 것 같습니다.²⁵ 만일 미국에서 L^AT_EX을 조판엔진으로 사용하는 워드 프로세서가 히트를 해서 시장 점유율이 1등 이 되면 그때에는 다 그것을 사용할 것입니다.

저는 앞으로 L^AT_EX을 사용할까요? 당연합니다. L^AT_EX의 위대함을 알아버린 지금, 열악한 시스템을 강요받는 것은 고통이니까요. 자기 애인이 알고 보니까 세상에서 가장 미인인데 제정신이 아닌 이상 다른 여자를 훑쳐 보겠습니까?²⁶ Ken Thompson은 Multics²⁷ 때문에 버릇이 나빠져 이전 상태로 갈 수 없었기에 UNIX를 만들었다고 합니다.²⁸ 저도 Ken Thompson 의 말에 공감합니다. L^AT_EX때문에 너무 버릇이 나빠져서 L^AT_EX의 방식이 아니고 책을 만드는 것은 지금은 별로 고려하고 싶지 않습니다. L^AT_EX을 이용해서 손쉽게 책을 만들 수 있는데 왜 다시 워드나 아래한글로 돌아가겠습니까? 게다가 공동작업도 가능하고, 버전 관리도 가능하고, 프로그래밍과 결합해서 자동화 할 수도 있고, 문서 품질도 훨씬 예쁘고 ... 어떻게 L^AT_EX을 사용하지 않을 수 있을까요? 나중에 MS-Word를 사용하도록 강요 받으면 어떨까요? 그건 그때 생각하겠습니다. 지금 L^AT_EX에 관해서 알아가기도 바쁘니 까요.

²³L_yX 라는 리눅스 소프트웨어가 이런 비슷한 일을 하는 것으로 알고 있습니다.

²⁴학교에 계신 분들과 이야기 해보면 진짜 심각한 것 같습니다.

²⁵너무 비판적인가요? 하지만 한국이 상당히 보수적이고 빨리 변화하는 것 같이 보이지만 잘 변화하지 않는 것도 사실인것 같습니다.

²⁶가끔 바람을 피울수는 있겠지만 곧 돌아올 수 밖에 없을 것입니다.

²⁷UNIX를 만들기 이전에 했던 프로젝트, 너무 프로젝트가 커서 실패하였습니다.

²⁸mult → uni, ics → x 프로그래머의 말장난입니다.