

실패정보의 활용

(주)한국데이터뱅크/고문 김 증 모

몇 일 전, “실패경험도 중요한 자산임으로 이를 활용하기 위한 실패사례활용 연구팀의 발족”이란 요지의 외신 인용보도를 본 일이 있다. 이와 비교될지 모르지만, 우리는 “타산지석”이나 “반면교사” 등으로 비유되는 교훈적인 말을 수없이 들어 왔다. 그러나 이것들을 얼마나 의미 있게 받아들였는가 또는 얼마나 실행에 옮겨 보았는가에 대해 새삼 돌이켜 보게된다.

약 십년전쯤, 해외의 소프트웨어를 수주, 개발하여 납품하는 입장에서 있었던 때의 일이다. 그쪽 시스템개발 표준에 따라 작업을 하고, 성과물뿐 아니라 작업의 경과기록도 함께 제출하도록 되어 있었다. 당시 지적사항중 한 예를 소개한다. 작업의 경과기록 중에는 품질확인기록으로서 “오류관리표”라는 것이 있는데, 점검하여 찾아낸 오류에 대하여 그 종류, 원인, 영향 등을 기록하는 것이다. 이렇게 찾아내어 기록하는 오류의 수에는 통계적인 기준이 설정되어 있었는데, 기준에 너무 못 미치는 오류수가 보고되고 있었다.

이런 현상에 대한 서로의 입장은 이렇다. 공개적으로 할 수 있는 말은 아니지만, 프로그래머로서는 자기가 작성한

프로그램에서 많은 오류를 찾아냈다는 것을 알리고 싶지 않다는 생각이었다. 그래서 찾아낸 오류에 대하여 수정은 해야하지만 기록으로서 모두 보고하고 싶지는 않았던 것이다.

품질관리 면에서는 프로그래밍과정에서 찾을 수 있는 오류는 모두 찾아 제거하는 것이 필수적인 작업이며, 오류가 별로 없이 완성했다는 것 자체가 오히려 이상하다. 제거한 오류가 적었다는 것은 오히려 찾을 수 있는 오류가 아직 남아있을 가능성이 있다는 이야기도 될 수 있다는 것. 물론 테스트케이스 설정의 문제일 수도 있다. 어쨌든 난 이도 등에 따라 차이는 있겠지만 기준치 전후의 오류발생과 제거를 정상으로 보는 것이었다. 또 이 기록으로서 오류의 발생원인분석 등을 행하여 개발작업의 개선에 이용하는 체제였다.

그러나 쉽게 동화되는 사정은 아니었다. 성공사례는 쉽게 자랑하고 싶어도 실패사례는 알려지는 것을 꺼리는 마음과 통하는 심정이라고나 할까? 어쨌든 실제의 현상에 비하여 기록으로 남기는 것이 적다는 것은, 실패의 경험도 자산으로 이용하려고 하는 관점에서 다시 생각해 보아야 하는 과제가 아닐까 한다.

또, 몇 년 전부터 어느 외국의 전문잡지에 고정 컬럼으로서, 정보시스템에서의 여러 가지 실패사례를 연재하고 있는 것을 보아왔다. 제목은 “움직이지 않는 컴퓨터”로 시작했다가 “위기에서의 탈출”로 바뀌었다. 기사는 사고개요, 원인규명과 대책 등으로 나뉘고, 대상은 금융, 통신, 교통 등 공공시스템과 일반기업도 포함되고 있으며, 발생원

인도 하드웨어, 소프트웨어와 운용상의 미비 등 다양했다. 물론 책임자의 인터뷰까지 포함되어 있었다. 이렇게 성공 사례의 발표가 아니면서도 공론화의 대상으로 나서는 것은, 아마도 실패사례를 소중한 자산으로서 가치를 부여하려는 마음일 것 같다. 우리는 언제 우리의 이런 기사를 볼 수 있을까? 생각해 본다.

얼마 전, MP3의 메모리카드에 수록된 내용을 지우고 다른 것을 수록하려고 하다가, 본래의 용량 32MB에서 30B 정도로 축소되어 있는 것을 발견했다. A/S에 문의하였더니 새 버전의 설정용 소프트웨어를 인터넷에 올려놓았으니 내려 받아 사용하라는 것이었다. 구 버전에 오류가 있었다는 의미였다. 시판중인 제품임으로 오류의 존재를 시인하고 사과하는 문구정도는 넣어 두었을법한데 없었다. 가능하다면 아마도 감추고 싶은 마음이었을 것 같다. 이 경우뿐만 아니라, 특히 소프트웨어의 잠재오류는 해당조건이 주어지지 않으면 발동하지 않으므로, 수년간 별 탈없이 잘 돌아가던 것도 장담할 수 없다는 특징을 잊지 말아야 할 것이다.

소프트웨어 테스트에서 새겨두어야 할만한 것 한가지. 어느 소프트웨어 오류에 대한 연구에서, 어떤 OS이용자들이 발견한 오류의 47%가 그 시스템 중 4%의 모듈 중에서 발견되었다는 보고가 있었다. 이런 현상으로서 어느 저명한 학자(G. J. Myers)는 테스트의 원칙 13가지를 나열한중에 9번째에 다음과 같은 요지를 쓰고 있다. “프로그램의 어떤 부분에서 오류가 아직 존재하고 있을 확률은 이미 그

부분에서 발견된 오류의 수에 비례한다.” 즉 소프트웨어 오류를 완전히 제거할 수 없다면, 상대적으로 오류가 많았던 부분에 테스트를 집중함으로써 테스트의 효과를 올릴 수 있다는 것이다.

그러면, 서두의 “실패지식 활용연구회”에 대하여 알아보자. 실패에서는 반드시 배울 것이 있으므로 그 과정을 검증하여 얻은 지식, 즉 실패지식을 공유하여 기술의 신뢰성 향상에 활용한다. 또 실패를 미연에 방지함으로써 사회적 안전성을 제고한다. 이런 목적으로 실패정보를 DB화하여 활용하기 위한 연구를 한다는 발족취지이다. 약 1년 전에 대학교수, 대기업의 임원 등 20명에 가까운 위원으로 발족, 그 동안 7회에 걸친 모임에서 실패사례의 발표와 활용방안의 토의를 해온 것 같다.

여기서의 사례보고와 토론의 한 토막. 어떤 회사에서는 사내에서 실패사례를 공유하려해도 각기 체면 때문에 좀처럼 제시하지는 않으면서도, 타인의 사례는 알고 싶어한다는 상황이어서 이를 해소하는 방안이 필요했다는 것. 또한 실패경험의 활용을 유도하기 위해서 원인규명과 책임추궁을 분리한 결과 가벼운 실패사례에서는 효과가 있었다는 것.

소프트웨어로부터 오류를 처음부터 완전히 제거하는 것은 불가능함으로 그 영향을 어떻게 최소화시키는가가 중요하다. 이를 극복하여 가는 과정에서 문제점을 사전에 또는 트러블이 발생하기 전에 발견한 경우 보상을 주기도 했다는 것.

실패의 조사에서는 과학적인 사실로서 인과관계의 확인과 사전예방을 위한 두 가지 측면으로 인식되어야 한다는 것 등이다.

우리의 정보기술분야에서도 이와 같은 종류의 과제가 산적해 있는 것이 아닐까! 체면에 연연할 때가 아닌 듯하다. 문제나 사고가 발생하기 전에 체계적으로 이에 대응하는 방안의 강구에 지혜를 모아야 할 것이다. 정보시스템 관련교육, 자격증제도 등 이 분야의 첫 단추부터 다시 끼우는 용단이 필요한 시점이 아닌가한다.