

소프트웨어 리엔지니어링

노 중 호

((주)CHNO 컨설팅 대표이사)

이 시대에 존재하는 모든 기업이나 정부기관 또는 개인들은 다 같이 상상을 초월한 상황에 직면해 있다고 하겠다. 격동하는 변화의 물결에 휩싸이면서 변화의 속도를 따라잡지 못하는 위험요수를 안고 있으며, 반면에 기회요소로서 누구나 제로 베이스선상에 서있다는 점이다.

여기서 우리가 생존하는 길은 위험요소에 매달리는 것보다 기회요소를 활용하는데서 찾는 것이 더 현명할 것이라고 생각된다. 기회요소를 찾는 것은 「잊어버리는 방법」을 배우는데서부터 시작된다. 지금까지의 결과는 과거이다. 그것이 사랑스럽고 애착이 가고 잊을 수 없는 것이라도 그것에 연연하다 보면 미래를 놓치게 된다. 특히 기술은 흘려보내야 한다. 그렇지 않으면 썩게 마련이다. 기술은 과거가 존재하나 기술자는 과거를 파괴해야 한다. “부인과 아들, 딸을 제외하고 바꿀 수 있는 것은 모두 바꿔라”가 리엔지니어링

이다. 사고방식, 습성, 관행, 절차, 제도, 조직, 구조 등 모든 것이 변화의 대상이다.

소프트웨어 리엔지니어링의 대상이 무엇인가에서부터 이슈화해 보자. 컴퓨터가 이해할 수 있는 언어로 체계화된 프로그램제품을 바꾸는 것이 리엔지니어링인가? 그것은 결과로 나타나는 것뿐이니 리엔지니어링의 대상이 아니다. 소프트웨어를 설계하고 개발하는 「엔지니어」들이 리엔지니어링의 대상이다. 특히 IT엔지니어들은 자신의 사고방식, 자신의 습성, 자신에게 익숙한 기술의 틀을 깨는데 보수적 성향을 갖고 있다. 자신의 틀에 들어오지 않는 것에 대해서는 공격적으로 방어하려 드는 경우가 많다. 새로운 구상이 딱혀들리가 만무하다.

소프트웨어를 다루는 사람들의 사고방식, 습성 등을 바꾸는 것이 소프트웨어 리엔지니어링이다.

기술의 혁명이 기존의 모든 체계를 뿌리 채 흔들어 놓으므로 소프트웨어도 리엔지니어링의 대상이 된 것이다. 그러나 소프트웨어 리엔지니어링은 단순히 소프트웨어영역에 국한해서 다룰 수 있는 과제가 아니다. 즉, 비즈니스영역, 정보생산영역, IT인프라구조영역, 시스템관리영역 등과 연계해서 다룰 문제이다.

기술의 트렌드는 객체지향이므로 소프트웨어도 객체지향

으로 리엔지니어링하여야 한다고 생각하는 사람이 많다. 따라서 객체지향분석(OOA), 객체지향디자인(OOD), 객체지향개발(OOP)등의 이론이 속출하고 있다. 그러나 대부분 어플리케이션 개발생산성을 다루고 있을 뿐이다. 어플리케이션 프로그래머, 시스템 디자이너, 시스템 분석가에 의해 정보 시스템이 개발되고 운영되어야 한다는 것을 전제로 하고 있다.

이러한 발상부터 리엔지니어링이 되어야 한다.