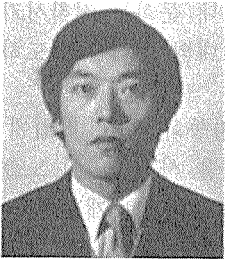


DACOM에서의 소프트웨어 技術開發 現況



林 春 鳳

韓國 데이터通信(株) S/W技術研究部長

새로 개발되는 시스템은 사용자 요구사항 및 장애 시스템 운영 환경을 고려하여 작성된 개발목표에 시스템이 접근하고 있는가를 확인하는 절대평가와, 운영되는 시스템의 경우는 보수작업 결과, 품질의 정도가 얼마나 상승되었는가가 확인되는 상대평가가 실시 되도록 한다.

지구촌은 정보산업 시대에 진입하고 있다.

정보산업사회에서는 정보를 유통시키는 소프트웨어 제품이 핵심 상품이 될 것이고 이 핵심 상품의 품질과 생산성은 소프트웨어 기술에 의해 좌우되게 될 것이다. 따라서 소프트웨어 기술은 정보산업 사회의 가장 중요한 기술이 될 것이 틀림없다.

소프트웨어 기술 발전과정은 수공업 시대, 반자동화 시대, 자동화 시대로 구분하고 수공업 시대를 다시 비구조적 수공업 시대, 구조적 수공업 시대로 구분한다.

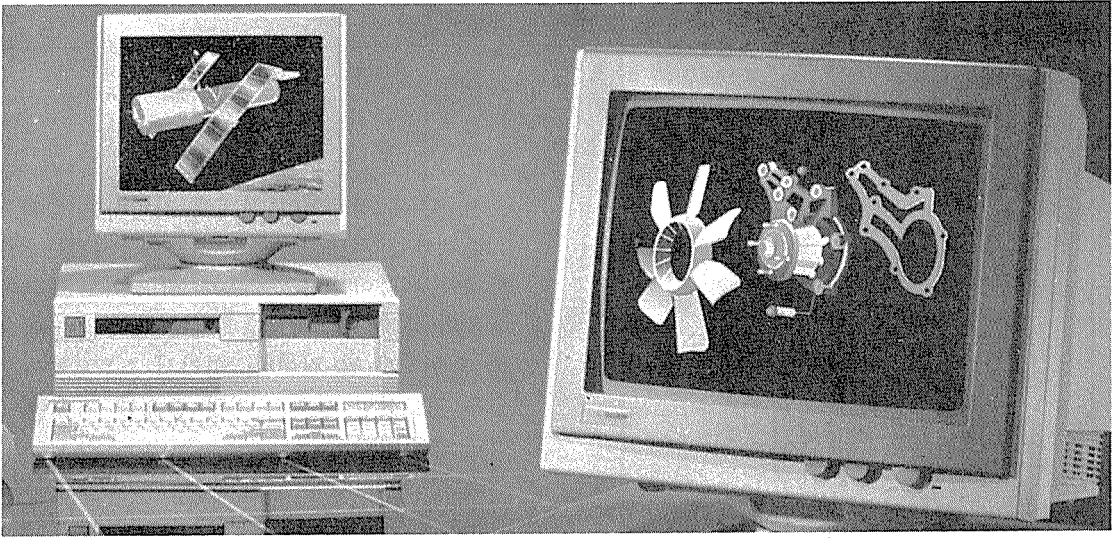
비구조적 수공업 시대는 소프트웨어를 예술이라고 칭할 정도로 하드웨어 효율, 개인의 독창성을 강조하여 개발 당사자만이 소프트웨어를 이해할 수 있고, 심지어 시간이 경과하면 개발당사자조차도 이해할 수 없는 경우가 있게 되는 시대이다.

구조적 수공업 시대는 소프트웨어 개발에 여러 단계(분석, 설계, 구현, 설치, 시험운용)를 두고 구조적 기법을 강조하여 개발 당사자는 물론 관련인 모두가 이해할 수 있도록 소프트웨어가 개발되고 개발표준이 만들어지고 운영 되는 시대이다.

반자동화 시대는 개발 방법론이 본격적으로 도입되어 개발지원도구가 만들어지고 활용되어 소프트웨어 품질과 생산성이 크게 향상되는 시대이다.

자동화 시대는 소프트웨어 요구사항의 조기 확정을 돕고 개발지원 도구 활용에 의하여 소프트웨어 코드가 자동 생성되는 시대이다. 요구사항도 원형제작기법(Prototyping)을 활용하여 반자동적으로 생성된다. 주로 사용되는 언어도 종래의 점차적(Procedural) 언어에서 비점차적(Non-Procedural) 언어로 바뀌게 된다.

DACOM에서는 설립초기부터 소프트웨어 기술개발사업을 꾸준히 추진해온 결과 부분적으



DACOM은 소프트웨어 품질검사 영역에도 AI 기법의 도입을 신중히 고려하고 있다

로 상당한 성과를 보고 있다. 이에 본고에서는 DACOM의 소프트웨어 개발환경을 소개하여 소프트웨어 개발 환경 구축에 조금이라도 도움이 되었으면 한다.

소프트웨어 개발 방법론으로써, PROMPT II를 기본으로 하여 개발단계를 구분하고, 단계별 활동목록이 작성되며, 활동별로 산출물이 지정되어 개발 과정 진행상황이 가시화되고 있으며, 개발 활동을 지원하기 위하여 구조적 기법인 CORE, LSDM, Yourdon 기법, SADT 등이 도입, 이용되고 상기방법론과 구조적 기법 적용을 지원하는 자동화 도구가 다수 도입도 되고 개발도 되어 선택적으로 활용되고 있다. 또한 시스템 개발 표준을 DACOM 상황에 맞게 만들어 유지하고 있으며 정기적인 교육을 통한 보급과 활용성을 높이는 방안으로 품질검사 의 견서에 구체적인 권고안을 제시하고 있다.

이런 방향에서 DACOM에서는 개발지원도구를 선정하여 도입하고 필요한 도구를 제작하는 전담조직을 두고, 소프트웨어 개발에 활용함으로써, 소프트웨어 품질과 생산성 향상에 노력하고 있다. 그림 1은 현재 DACOM의 S/W 공학환경을 소프트웨어 개발방법론, 구조적 기법과 개발 지원 도구관점에서 본 것이다. 또한 이러한 환경을 기반으로 소프트웨어 품질보증을 위하여 소프트웨어 시험 및 품질검사를 전

담하는 조직을 운영하고 소프트웨어 기술 전파를 위한 소프트웨어 공학 관련 지침서 작성 작업이 활발히 진행되고 있기도 하다.

이로 미루어 DACOM의 소프트웨어 기술은 현재 구조적 수공업시대 문화를 정착시키고 부분적으로 반자동화 시대의 진입을 시도하고 있는 중이다.

DACOM에서는 반자동화 시대 진입에 앞서 수작업 환경의 도구들을 사용하면서 노출된 문제들을 분석하고 있다. 그리고 개발계획 수립 시에는 도구개발 가능분야를 제시한 후 실사용 개발부서의 의견을 듣고, 각 분야별 우선 순위를 결정하여 일년 단위로 개발을 추진하고 있다.

'89년도 중점 목표를 중심으로 S/W 기술 개발 노력을 소개한다.

첫째, 소프트웨어 개발 방법의 정립이다.

시스템 개발은 사용자, 관리자 및 개발자에 의해 이루어진다. 개발기간 동안 사용자는 자신의 요구대로 시스템이 개발되고 있는지 알고 싶어 하며, 관리자는 기대된 기간과 비용내에서 개발이 진척되고 있는지 알고자 한다. 그리고 개발자는 그들간의 공용화된 언어로 의사소통이 이루어지기를 원한다.

이러한 삼자의 요구를 충족시키기 위한 체계적 관리와 지원 기법이 시스템 개발 방법론이다. 시스템 개발 방법론을 사용하여 시스템을 개발

개발단계	방법론	구조적 기법					개발지원도구					
개시	통합 방법론	CORE										
분석		SSA	LSDM				자료사전처리기	Peregrine IV	응용시스템생성기	비용추정도구	프로젝트 관리 지원 도구	
설계		SSD										
구현				JSP	YSP							
설치												
시험운용												

- 주 1) 통합 방법론 : DACOM이 다양한 규모와 특징을 가진 사내 프로젝트에 융통성있게 모든 수명주기에서 적용되도록 개발한 시스템 방법론.
- 주 2) CORE : 영국 Systems Designers사에서 개발한 요구사항 분석기법. CORE는 Controlled Requirements Expressions의 약자.
- 주 3) SSA/SSD : 미국 Yourdon사의 구조적 시스템 분석/설계기법(구조적 Systems Analysis/Design).
- 주 4) LSDM : 영국 정부와 LBMS사가 공동으로 개발한 시스템 분석/설계 방법으로써 LBMS 구조적 Development Methods의 약자.
- 주 5) JSP : Jackson's 구조적 Programming의 약자.
- 주 6) YSP : Yourdon 구조적 Programming의 약자.
- 주 7) Peregrine IV : DACOM이 미국 Peregrine Systems사에서 도입한 4세대 언어.
- 주 8) 비용추정도구, 자료사전처리기, 응용시스템생성기, 프로젝트관리지원도구는 모두 DACOM에서 개발한 소프트웨어 개발 지원도구임.

그림 1. 소프트웨어 수명주기 단계별 기법 및 도구 적용

하면, 사용자는 자신의 요구사항을 직접 확인할 수 있고, 관리자는 개발의 진도 및 비용이 기대치대로 시행되고 있는지 알 수 있으며 개발자는 방법론을 공유화된 의사소통 도구로 활용하여 상호 의사 교환을 용이하게 할 수 있다.

이에 DACOM에서는 PROMFT II, III의 개념을 도입하여 기도입된 Yourdon기법, CORE, LSDM 등의 기법과 Peregrine IV, Infomix 등의 4세대 언어와 DBMS, 자체적으로 개발한 자료사전 처리기, 응용 시스템 생성기 등의 자동화 도구를 상호 연계하여 통합방법론을 만들었다. 통합방법론은 종래의 개발 주기에 기초하고 있으나, 원형 개발 기법을 사용하여 문서를 통합하여 표준화하는 동시에 문서

분량을 효율적으로 조절하도록 하였다.

활용을 위한 사내 전수 교육이 아직은 미진한 상태에 있다. 따라서 '88년도에 자동화된 지원도구를 개발 완료하여 통합방법론의 관리 활동 계획, 실시 및 점검을 용이하게 하고 있다.

둘째, 소프트웨어 개발지원 도구의 통합화이다.

소프트웨어 개발지원 도구가 별개의 체제로 독립되어 있어 개발자는 각기 다른 사용법을 익혀야 하는 불편이 있고, 지금 사용하는 도구에서 다른 도구를 사용하려면 새로운 체제로 전환되어야 한다. 따라서 도구 사용자는 오히려 도구 사용이 시스템 개발 보다 불편한 것이 될 수도 있다. 따라서 도구 사용이 분명 소프트웨

어 품질과 생산성 향상에 이바지하기 위해서는 도구의 통합이 필요하다고 보고 이를 추진하는 중에 있다.

셋째는 소프트웨어 재사용(Reuse)을 위한 도전이다.

새로운 시스템을 개발할 때는 항상 처음 개시하는 것처럼 시작하지만, 사실상 하는 일은 이미 앞서서 한 일을 전혀 새로운 일처럼 하고 있는데 지나지 않는 것을 종종 느낄 수 있다.

예컨대, 입출력 루틴, 보고서 작성 루틴, 공통계산 루틴 등은 대부분 반복이고 또한 일정한 형태임을 알 수 있다. 앞서 산출된 업적을 이용하여 무엇인가 활용할 것이 있는 상태에서

일을 진행하면, 시스템을 개발할 때, 이용할 것이 하나도 없는 상태에서 보다는 생산성이 향상될 것이며, 그것도 자주 사용된 것을 사용한 다면 사용된 횟수만큼 품질이 높은 것이 될 것이므로 소프트웨어 품질도 따라서 개선될 것으로 본다. 하지만 재사용은 어렵다.

기개발된 소프트웨어 부품만 모은다고 해서 되는 것이 아니라 이를 활용할 수 있도록 체계적으로 저장하고 이를 사용하는 방법이 구축되어야 한다. 지금까지는 부품 라이브러리를 중심으로 DBMS 체제를 중요시했으나 아직은 부족한 감이 있다.

따라서 DACOM에서는 그림 2와 같은 체제

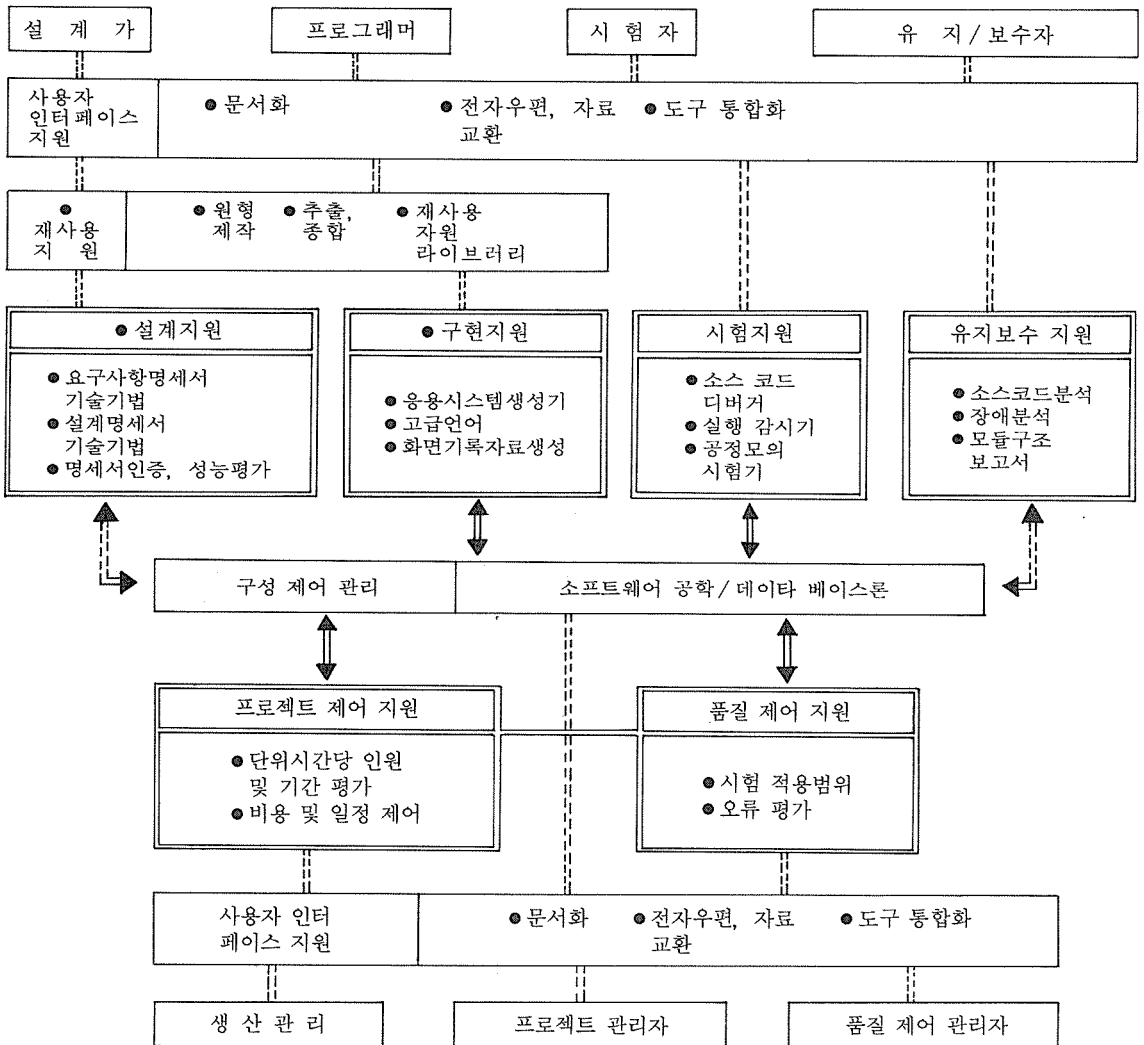


그림 2. 재사용 체제

로 소프트웨어 재사용에 도전하고 있다.

네째는 소프트웨어 품질보증이다.

하드웨어의 가격이 낮아지고 소프트웨어 가격이 상대적으로 높아지면서 소프트웨어의 중요성은 높아졌다. 하지만 좋은 소프트웨어의 기준이 명확하지 않은 상황에서는 소프트웨어의 가치를 알 수 없으며 이에 따라 그 중요성도 퇴화되는 감이 있다.

이에 DACOM에서는 아래와 같은 체제로 소프트웨어를 평가하는 체제를 갖추고 있다.

새로 개발되는 시스템은 사용자 요구사항 및

장래 시스템 운영 환경을 고려하여 작성된 개발목표에 시스템이 접근하고 있는가를 확인하는 절대 평가와, 운영되는 시스템의 경우는 보수 작업 결과, 품질의 정도가 얼마나 상승되었는가가 확인되는 상대 평가가 실시되도록 한다. 또한, 품질목표에 따른 시스템 성능 평가가 가능하도록 품질목표는 표1과 같이 정하고 품질기준에 따라 점검항목, 점검항목에 따라 점검방법, 점검방법에 따라 품질수치가 있다고 보고 이에 대한 노력을 계속 경주하고 있다.

표1 품질목표와 품질기준의 상호관계

<범례>

- * : 현재 적용 가능한 품질기준
- : 해당 품질목표를 충족하는 품질기준
- 빈칸 : 관련없음

품질목표 품질기준	효율성	융통성	무결성	상 호 운용성	유 지 보수성	이식성	신뢰성	응 답 속도성	재 사용성	시 험 용이성	사 용 용이성
사용감사성			●								
사용통제성			●								
*정 확 성							●				
통신공통성				●							
*의사교환성											●
*완 전 성							●				
*일 관 성					●		●				
자료공통성				●							
*오류허용성							●				●
실행효율성	●										
확 장 성		●									
*도구설비성										●	
기계장치독립성						●			●		
*모 들 성		●		●	●	●			●	●	
운 용 성											●
*응답시간적합성								●			
원시코드독해성		●			●	●			●	●	
기억장치효율성	●										
*구조적단순성					●		●			●	
*시험적합성							●				
처리음적합성								●			
*추적가능성							●				

다섯째는 소프트웨어에 AI 기법의 활용이다. 개발되어야 할 시스템이 점점 복잡해지고 거대해짐에 따라 미리 정해진 단계에 따라 시스템을 구성하는 데에는 많은 노력과 시간이 요구되고, 무수한 자원의 투입에도 불구하고 시스템 성공 가능성을 확신할 수 없게 되었다.

그러므로, 해당 시스템에 대하여 가능한 풀이의 공간으로부터 적절한 시행 착오적 방법으로 해결 가능성이 희박한 부분을 점진적으로 제거함으로써 기대하는 시스템에 적은 자원으로 접근하도록 하는 것을 AI 활용연구의 목표로 하여 이에 대한 노력을 경주하고 있다. 예컨대 DACOM에서는 망(network) 진단시스템 개발에 AI 기법을 도입한 바 있다.

한편, DACOM은 소프트웨어 품질검사 영역에도 AI 기법의 도입을 신중히 고려하고 있다.

마지막으로 언급해야 할 것은 연구원들이 새로운 상품 개발에 우선한 나머지, 제품 사용자의 요구에 맞게 고쳐 주는데는 매우 인색한 것이 우리의 일반적인 현실이라는 점이다. 이를 감안하여 사용자 인터페이스 측면을 강화하여 친밀성을 높이고 지원 방법론 및 도구 자체도 구조적 설계 이념에 따름으로써 유지 보수성을 높이고 있다.

그리고 무엇보다도 인간적인 유대강화를 꾀하고 정기적인 도구 사용현황을 조사하여 사용자 측면에서 문제점을 파악하고 개선점을 찾는 데 노력하고 있다.

휴 게 실

유태인의 지혜가 담긴 「탈무드」에 다음과 같은 비유가 있습니다.

「항구에 두 척의 배가 떠 있었습니다.

한 척은 출항중에 있고 한 척은 항구로 돌아오고 있었습니다. 사람들은 모두 떠나는 배에 환호를 보내면서 들어오는 배는 거들떠 보지도 않았습니다.」

이것을 보고 한 현자(賢者)가 말했습니다.

“바다로 나가는 배를 보고 그렇게 너무 축하하지 마시오. 앞으로 그 배가 몸서리쳐지는 폭풍을 만날지, 아니면 무서운 위험을 만

날지 그 누구도 장담할 수 없지 않습니까? 차라리 짐으로 돌아갈 손님들을 태우고 무사히 부두에 닿는 배를 축하하는게 나을겁니다.”

이 세상의 모든 일도 이와 마찬가지로 아닐까요?

힘찬 출발의 감격도 소중하지만 알찬 성취의 보람은 더욱 값진 법입니다.

용(龍)의 머리에서 뱀의 꼬리가 나오지 않도록 우리 모두 마음의 고삐를 다시 한번 점검할 때입니다.