

□ 기술해설 □

정보공학과 CASE

대우정보시스템(주) 이 재 관*

● 목	차 ●
1. 서 론	2.3 정보공학방법론을 지원하는 CASE 틀
2. 본 론	2.4 고객지향적 정보공학방법론과 CASE
2.1 고객지향적 정보공학방법론이란	툴로의 전환
무엇인가?	3. 결 론
2.2 정보공학 변천과정	

1. 서 론

소프트웨어 위기를 극복하기 위해 우리는 소프트웨어 생산성 향상 노력을 끊임없이 경주해 왔다. 특히 최근에는 소프트웨어 생산성 향상과 품질보증을 위해서 컴퓨터를 이용한 소프트웨어 개발방법론을 정립하고 자동화하여 소프트웨어가 또다른 소프트웨어를 대량으로 만들어내는 소프트웨어 대량생산체제가 구축되었다. 그러나 대량 생산된 소프트웨어는 여전히 사용자나 기업 경영자의 마음에 흡족하지 못하고 경영전략을 수행하는데 별로 도움이 안되는 가치가 낮은 제품으로 전락하였다. 결국 CASE의 무용론이 대두되며 많은 비용을 부담했던 고객(기업 경영자)으로부터 불만이 제기되고 우리는 그것에 대한 보상을 해 주어야 하는 상황에 처하고 말았다.

CASE 틀과 새로운 개발방법론을 적극적으로 활용해도 우리의 위기는 극복될 수 없는 것인가? 소프트웨어를 많이 만들어 제공하면 해결할 수 있다고 믿었던 우리의 전제가 잘못 설정되었다는 말인가? 그렇다. 고객이 원하는 것은 대량 생산된 품질이 좋은 소프트웨어가 전부가 아니다. 고객은 경영전략을 수행할 수 있는 정보전략과 정보자재를 제공받기 위한 욕구로 가득차 있었다.

알고 싶을 때 무엇인가를 알아야 하는 욕구. 우리는 그 욕구를 만족시켜 주지 못했다. 단지 많은 소프트웨어를 만들어 변화무쌍한 고객의 욕구를 물량으로 채워 줄 수 있을거라고 기대하고 있었다.

우리는 요즘 세상의 화제가 되고 있는 고객지향의 경영전략에 대해 다시 한번 생각해 보아야 한다. 정보전략도 경영전략과 같이 고객지향적으로 변해야 우리의 고객은 만족하고 감동할 것이다. 우리는 변해야 한다. 고객이 무엇인지 또는 누구인지를 다시 정의하고 고객을 위한 활동을 찾아서 즉시 수행해야 한다. 그리고 고객이 만족할 수 있도록 끊임없이 인식하고 노력해야 한다.

우리는 고객이 원하는 것이 대량 생산된 소프트웨어가 아니라 정보전략과 정보 그 자체라는 사실을 깨닫게 된다. 우리는 지금까지 생각해 왔던 고정관념을 깨고 소프트웨어 생산성에서 고객의 정보전략과 정보관리를 위한 새로운 패러다임으로 전환하여 고객지향적 방법론과 툴로서 그 가치를 재평가 받아야 할 것이다.

필자는 이 리포트를 통해 정보공학방법론과 CASE 틀을 고객지향적 관점에서 재조명하기 위해 방법론이 수용하고 있는 범위와 표준에 대해 제시하고 특히 방법론의 변천과정을 기술하여 본질을 인식하고 국내에서 많이 사용되고 있는

*정회원

방법론에 대해 알아보고 고객지향적 방법론으로서 변신할 수 있는 전환점을 모색할 것이다. 또한 정보공학방법론에 기초하고 지원하기 위해 만들어진 CASE 툴에 대하여 정리해 보려고 한다. 우리가 고객지향적 CASE 툴을 선정하려고 할때 가져야 하는 판단기준과 국내에 소개된 툴을 비교해서 이해하고 고객을 위한 활동을 정의할 수 있도록 한다.

정보저장소(Repository, Encyclopedia)를 통해 방법론과 툴의 관계를 재설정해 보고 그것들을 결합하여 고객을 만족시키는 활동을 잘 수행할 수 있는지를 확인해 볼 것이다. 그리고 각 기업들이 특별히 자체의 방법론을 설정하고 CASE 툴과 결합하여 사용하려할때 고려하여야할 사항도 생각해 보고자 한다.

정보전략은 경영전략과 마찬가지로 고객의 욕구를 만족시키기 위해 무엇을 할것인가를 결정하여 수행하는 방법을 설정, 구축하는 것으로 정보환경의 지배를 받는다. 그래서 정보전략은 고객지향의 환경 즉 기업문화에 영향을 받게 된다. 서양의 합리적인 일 문화에서 서양인에 의해 만들어진 정보공학방법론과 CASE 툴은 고객의 정보전략과 정보관리에 이용되기 위하여 우리는 어떻게 적응해야 하고 기업문화를 바꾸어야 하는가 혹은 우리에게 맞게 한국적 정보공학방법론이 되도록 변형시켜서 사용해야 하는가에 대해 결정을 내려야 한다. 이 모든 결론은 후에 더 많은 고민을 통해 내리도록하며 특히 정보전략계획을 수립하고 비즈니스영역분석을 실시하고 비즈니스시스템을 설계할 조직을 어떻게 구성하고 구축가동된 고객의 정보시스템과 정보관리는 누가 수행해야 하는지에 대해서 제시할 것이다.

우리는 정보공학방법론과 CASE 툴이 고객의 정보전략과 정보를 관리할 수 있는 새로운 패러다임이라고 확신을 해야 한다. 단지 기존에 가지고 있던 모든 관념을 버리고 고객을 재정의하며 그 고객을 위하여 진정 무엇을 해야하는지를 설정하여 수행할 때 고객은 만족할 것이며 우리는 또한 고객의 만족을 통해 정보산업의 신개척자임을 확인할 것이다.

2. 본 론

2.1 고객지향적 정보공학방법론이란 무엇인가?

정보공학은 정보전략계획부터 시스템 설치까지 컴퓨터 시스템 어플리케이션의 전 라이프사이클을 위한 구조화된 방법이라고 정의내릴 수 있다. 정보전략계획의 앞단계는 비즈니스전략(경영전략, 사업전략)을 정의하는 비즈니스전략계획 수립과정이다. 이 비즈니스전략계획은 다변화하는 경영환경속에서 생존하기 위해 고객을 재정의하고 그 고객을 만족시키기 위해 수행하여야할 활동을 정의하는 것이다. 이 단계는 정보공학의 범위에서 벗어나며, 엄격히 말해서 기업의 폭넓은 임무를 수행하기 위해 사업계획을 수립하는 경영층의 일로만 알아왔다. 그래서 경영전략계획은 경영전략을 수행하기 위해 요구되는 정보전략에 대해서는 정의하지 않는다. 그러면 우리는 질문을 가지게 된다. 정보공학의 범위는 어디까지인가? 정보전략과 비즈니스전략계획은 어떤 관계를 가지고 있는가?

2.1.1 정보공학의 범위

정보공학은 비즈니스전략으로부터 정보전략을 수립하기 위해 기반이 되는 정보를 얻게 된다. 특별히 비즈니스를 수행하는 기업의 고객에 대한 정의와 그 고객을 위해 기업이 수행해야 하는 활동을 인식하게 된다. 우리는 이때 비즈니스전략과 정보전략의 관계가 유기적으로 밀접하게 연결되어 종속성(Dependency)을 가진다는 것을 깨닫게 된다. 그래서 정보전략계획은 비즈니스전략계획과 동시에 수행되거나 비즈니스전략계획을 수립한 후에 실시하는 것이 바람직하다.

국내의 정보공학환경은 JMA(James Martin Association)의 IEM(Information Engineering Methodology)과 Earnst & Young의 Navigator가 중심적이며 나름대로의 지원하는 CASE 툴과 연계하여 활동하고 있다. 이후 기술되는 정보공학방법론은 IEM과 Navigator를 비교해서 그 차이점을 정리하는 것으로 풀어나갈 것이다.

전 시스템 라이프 사이클은 여러 단계로 구분된다. 정보시스템전략 연구, 정보기술전략 연구, 논리적인 설계, 물리적인 설계, 개발/구축

그리고 이행/설치, 정보공학은 이 모든 단계를 지원한다. 정보공학 이전의 방법론에서는 대체로 정보전략계획단계가 생략되었거나 경영전략과 완전히 분리하여 별도로 수행하고 반영하는 형식을 취하고 있다. 또한 정보공학은 현재 많이 발전하여 개발환경과 목표환경이 다양해졌지만 중앙집중화된 데이터처리환경에서 일괄과 On-line 처리를 목표로 하고 있다. 중앙집중화된 환경 개념과 기술은 아직 일반적인 것으로 정보공학 방법론도 일반적인 개념과 기술을 추구하고 있으며 다양한 환경 변화에 따라 설계기술의 발전을 압력받고 있다. 보통 정보공학의 범위를 논할 때 정보전략계획부터 이행/설치까지의 기술에만 국한시키지만 프로젝트관리, 품질보증 그리고 소프트웨어 패키지과 같은 통제관리기술도 포함시켜야 한다. IEM의 경우는 통합, 제공하지 않고 있으며 JMA에서는 PACE라는 별도 분리된 프로젝트관리 방법을 가지고 전략계획과 산출물에 대한 품질보증을 위해 부가적인 지침을 제공한다. 그러나 품질측정을 위한 기준이 설정되어 있지 않아 아쉬움이 있다. Navigator는 단계(Phase)와 과업(Task)의 구조안에서 프로젝트관리와 소프트웨어 패키징을 통합, 제공하고 제한과 검사활동으로 품질관리를 제공한다. 그러나 산출물에 대한 품질수준을 측정하는 방법은 따로 없다. 그래서 정보공학은 “측정이 곤란하다”라는 논쟁을 계속해 오고 있다.

2.1.2 정보공학 표준들

구조화된 방법은 구조적, 기술적, 도큐먼트적인 관점의 세가지 표준으로 나누어진다. 정보공학 방법론은 조직적이라는 표준이 추가된다.

구조적 표준은 Stage/Phase의 순서에 대한 정의, 적용해야 할 기술, 만들어지는 산출물에 관련되는 것으로 분석가/설계자의 실제적인 어플리케이션이라기보다 작업을 위한 과업중심적인 개념이다. 그림 1.1에서 보여주듯이 IEM은 7단계로 나누어진다. Navigator는 계획, 분석, 설계, 구축과 설치로 4가지의 순차적인 Phase로 구성된다.

IEM의 첫단계는 정보전략계획(ISP)단계이다. ISP는 기업의 비즈니스계획을 지원하는 미래의

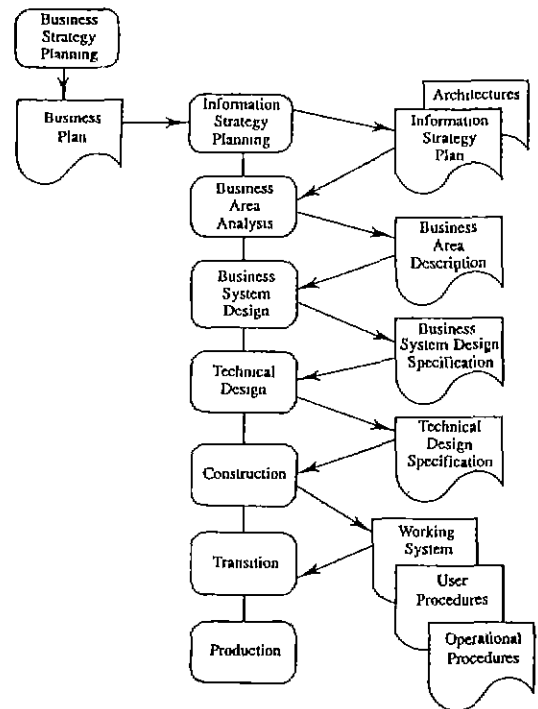


그림 1.1 IEM의 Stage와 산출물

컴퓨터 어플리케이션의 설계와 개발을 위한 계획/구조를 설정한다. 기업의 데이터와 활동의 높은 레벨의 모델을 포함하고 있는 정보구조, 그 구조를 지원하기 위한 실현가능성을 가지고 있는 주요한 시스템의 다른 형태를 보여주는 비즈니스시스템구조와 시스템구조를 지원하기 위해 하드웨어, 소프트웨어 그리고 통신환경을 보여주는 기술적 구조로 구성된다. 정보구조는 비즈니스영역분석(BAA)단계의 기초정보가 되고 비즈니스시스템구조는 비즈니스시스템설계(BSD)단계의 기초정보가 되며 기술구조는 기술적 설계단계의 기초정보가 된다.

ISP단계후 우리는 비즈니스영역분석(BAA)단계를 수행하는데 ISP단계에서 정의한 비즈니스영역을 중심으로 데이터와 기능요구를 더 상세하게 분석하고 매트릭스를 사용해서 두 객체형태의 상호작용성을 분석하는 활동도 포함한다. 그것은 비즈니스영역에서 요구되는 “무엇(what)”을 정의하고 현시스템은 비즈니스영역모델을 확인하는 수단으로 사용하곤 한다. 이 단계의 후반부에서, 비즈니스영역을 논리적으로

관계된 비즈니스 기능의 단위로 클러스터화하여 소요자원과 우선순위를 결정하게 된다. 이것을 기초로 하여 비즈니스설계를 위한 개발프로젝트를 결성하고 설계를 시작하게 된다. 비즈니스영역분석단계는 매우 중요한 단계로 데이터와 활동분석을 위해 별도 팀을 분리, 구성하여 실시하는 것이 바람직하다. 기업의 장기적인 데이터 관리를 잘 수행할 수 있도록 데이터관리팀과 데이터베이스관리팀을 구성하여 책임과 권한을 가지고 진행하게 한다.

비즈니스시스템설계(BSD)단계는 세가지 주요 과업을 가지는데, BAA 단계에서 정의한 비즈니스요구를 어떻게 수행할 것인지를 정의하고 컴퓨터시스템과 비즈니스를 수행하는 사용자간의 환경을 하드웨어/소프트웨어 환경에 맞게 정의하는 사람/기계간 인터페이스설계와 운영성을 감안한 처리방식의 결정, 즉 효율적인 처리를 위해 절차를 묶는 결정을 하는 것이다. 특별히 하드웨어/소프트웨어의 성능을 확보하기 위한 활동은 다음 단계인 기술적인 설계에서 이루어진다.

기술설계(TD)단계는 비즈니스시스템을 위해 BSD단계의 일반적인 설계를 근거로 하여 재정련하고 선정된 하드웨어/소프트웨어 환경을 위해 데이터스토리지와 응답시간 성능을 고려한 구조화된 물리적 설계이다.

Navigator에서는 비즈니스시스템설계단계와 구축단계에 그 기능이 나누어져 있다. 개념적인 스키마를 외부적인 스키마를 통해 내부적인 스키마로 변환하는 중요한 단계라고 생각되며 특별히 시스템의 성능에 대하여 점검하고 데이터베이스 구축단계를 위해 이 시점에 설치하는 것도 바람직하다.

구축단계는 전단계에서 정의된 비즈니스시스템을 컴퓨터시스템에 설치하는 것으로 주로 개발/테스트용(향후 가동계의 백업용으로 전환됨) 하드웨어에서 코드를 생성한다. 데이터베이스 설치하는 이 단계에서도 가능하지만 코드 생성후 곧바로 단위(Unit)테스트를 하기 위해 기술설계 단계에서 미리 수행하기도 한다. 구축단계에서 중요한 것은 하드웨어에 종속적인 사항에 대하여 확인하고 수행성을 테스트하는 부분도 포함되어

있다는 것이다. 테스트에 대한 전략은 별도로 설정하여 팀을 구성하고 실시해야 한다.

이행단계는 기존에 있던 시스템을 전단계까지의 작업으로 만들어진 새로운 시스템으로의 전환을 하는 것이다. 실 데이터에 대한 변환이 이루어지므로 치밀한 계획이 필요하고 만일의 사태에 대비한 "Fall-back" 시나리오를 수립하는 것도 감안해야 한다. 시스템 테스트시점에 리허설을 몇번 수행해 보는 것이 비즈니스의 공백을 줄일 수 있는 방법이다.

마지막 단계는 가동이다. 보통 상세한 작업항목이 나열된것은 없지만 비즈니스 기능을 잘 수행하는지를 확인하고 운영조직의 체계적 변환을 점검한다. Encyclopedia 관리상의 관점에서는 유지보수와 시스템보호체계로의 전환을 의미하는 것이다.

Navigator는 단지 네가지 Phase를 가진다. 계획 Phase는 IEM의 정보전략계획 Stage와 비슷하나 약간의 차이가 있다. 이 Phase에서는 기업 목표와 전략적 비즈니스계획을 지원하기 위해서 비즈니스 기능, 정보요구, 조직을 표현한 기업 비즈니스모델을 구축한다. IS원칙, 현정보시스템 평가와 기업모델은 미래의 시스템 구조와 어플리케이션 프로젝트계획의 기초로서 사용된다.

분석 Phase는 계획단계의 구조의 데이터와 어플리케이션 부분에 대해 집중적인 분석을 한다. 이 단계는 계획단계안에서 정의한 비즈니스영역을 중심으로 하여 제안된 비즈니스해결방안의 개념적인 시스템을 설계한다. 개념적인 설계는 모든 어플리케이션 설계와 개발 소프트웨어에 있어서 논리적이고 일반적이다. 분석 Phase의 주요입력정보는 EIA이고 주요 출력정보는 비즈니스영역정보 모델(BAIM)안에서 상세화된 어플리케이션이 개념적 시스템설계이다. EIA의 보다 상세한 부분집합이고 기술사항과 조직적 요구는 단지 도큐먼트가 된다.

설계 Phase에서는 어플리케이션을 위해 논리적이고 개념적인 설계를 물리적인 설계로 전환한다. 이 단계는 IEM의 비즈니스시스템설계와 데이터 변환, 테스트와 IS 하부구조 설계 등이 더 포함되어 있다.

구축과 설치 Phase는 설계단계에서 만들어진

물리적인 정의로부터 가동할 수 있는 어플리케이션을 생성하고 테스트하는 것이다. 그리고 하드웨어, 소프트웨어와 통신장치도 이때 설치된다. 어플리케이션의 유지보수 및 성능향상계획을 수립하고 이 단계를 마친다.

Navigator는 루트맵(Route Map)의 기능을 소개해 왔다. 이 루트맵은 계획단계를 가지고 있지 않지만 빠르고 제한된 개발시간을 효율적으로 수행키 위해 작은 프로젝트 형태로 구성할 수 있도록 제시한다. 맵은 사용자에게 그 방법을 잘 사용할 수 있도록 별도의 부분으로 구성하여 제공하고 있다.

IEM과 Navigator는 구조적으로 상당히 비슷한 모습을 하고 있으나 실제 수행내용은 차이점을 보이고 있다. 어느 것을 선택하느냐하는 질문은 CASE 툴과의 관계에서 명확해질때까지 조금 참기로 하자.

기술적 표준은 그림 1.2에서 보여주듯이 기본적인 기술과 그것들의 상호관계를 나타낸 것이다.

이 기술은 특별한 단계만을 위해 정의되는 것이 아니고 여러 단계에 적용되고 상호연관성을 가지고 관계를 맺고 있다. 고객에 대한 정의와 활동을 가지고 수행된 기업의 경영전략을 출발

점으로 하여 시스템과 데이터베이스를 만들때까지의 작업은 또 다른 다이어그램 기술을 통해 수행된다. 주요한 예로, 실제관계데이터모델링, 프로세스 분해와 프로세스 종속성 분석기술 등이 있다. 특히 액션다이어그램기술은 논리적인 분석을 통해 프로세스의 내용을 정리할 때 사용되며 절차를 구성할 때도 사용된다. 프로그램언어와 비슷하다고 느끼겠지만 표준화된 언어로 사용자와 의사전달할 수 있다는 점을 감안하면 무조건 코드로만 취급해 버릴 수는 없다.

정보공학에서 사용되는 논리적인 설계기술은 여러가지가 있지만 그중에서 특징있는 것만 알아보고 비교해보기로 한다.

• 실제관계모델링

이 기술은 기업의 비즈니스 목표와 그것을 수행하기 위해 요구되는 목표들을 지원하기 위해 실제관계도(ERD)를 가지고 논리적인 데이터모델을 만드는 것이다. 데이터 객체는 실제와 서브타입으로 나누고 그 실제안에 데이터의 속성을 정의한다. IEM은 BAA단계에서 ERD가 완성이 되고 ISP단계에서는 주제영역모델(Subject Area Model)을 만든다. 그러나 ISP단계에서 실체가 확연히 드러난 것에 대해서는 나타내 주고 있다. 하여간 데이터 중심적인 정보공학에서는 이 기술이 중요하고 여기서 만들어진 데이터 모델은 향후 모든 단계에서 품질을 점검하는 기준으로 제시되어진다. Navigator는 계획과 분석 Phase에서 수행된다.

• 프로세스 분해

기업안에서 비즈니스 프로세스는 추상적인 수준에서 정의된다. IEM에서는 ISP와 BAA 단계에서 행해진다. ISP 단계에서는 비즈니스영역정도의 기능수준에서 점차 분해하여 제일 높은 프로세스수준에서 멈춘다. BAA 단계에서는 ISP단계에 이어 프로세스를 단독적으로 수행이 가능한 원소적(Elementary) 프로세스수준까지 분해한다. 요즘 유행하고 있는 비즈니스 리엔지니어링의 프로세스와의 관계에 대해 의문이 생길것이다. 비즈니스 리엔지니어링 개념에서의 프로세스는 IEM의 기능(Function)과 일치하며 IEM의 프로세스와는 수준이 다르다는 사실을 인정해야 한다.

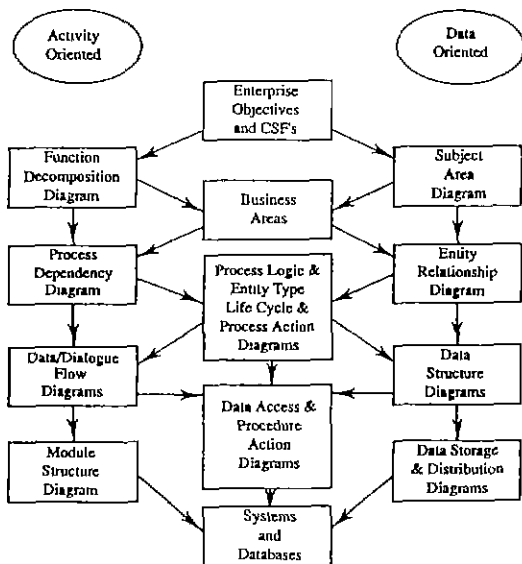


그림 1.2 정보공학 기술

- 프로세스 종속성

이 기술은 프로세스가 수행되는 순서를 정의하는 것으로 IEM의 ISP와 BAA 단계에서 적용된다. Navigator에서는 적용되지 않는다. 프로세스 분해도의 프로세스 수준과는 일치하지 않으며 단지 실체에 작용하는 프로세스에만 관련된다. 즉 실체에 있는 정보를 조회하여 확인하는 프로세스는 이때 기술되지 않으며 예를 들어, “고객 등록” 프로세스 수행후 “주문접수”하는 프로세스가 수행된다. 아래에서 기술될 데이터흐름도와 비슷한 형태로 나타내지므로 혼동을 많이 하고 용도에 대한 논란도 계속되고 있다.

- 데이터흐름도

이 기술은 프로세스 분해도안에서 지원되고 있는 것과 부가적인 정보를 보여 줄 수 있도록 하고 있다. 이 부가적인 정보는 기업주변 즉, 고객을 중심으로 하여 주변에 있는 데이터의 흐름을 보여 준다. 그 데이터는 단지 한 프로세스를 거쳐서 상태가 변화되는 흐름을 의미한다. IEM의 경우는 BSD 단계에서 프로세스분해와 종속성, 데이터와의 상호작용성을 검증하기 위해 사용되며 Navigator는 계획단계에서 그려진다. 데이터흐름도는 프로세스 분석용으로 처음 나타났으며 프로세스 종속도와 혼동을 일으키고 있다. 실제로 프로세스 종속도는 데이터흐름도에서 데이터스토어와 그곳에서 나오고 들어가는 데이터흐름을 제거한 모양과 같기 때문이다. 구조적 방법을 지속적으로 사용왔던 분석가들은 데이터흐름도의 중요성을 강조하지만 필자는 데이터와 프로세스의 상호작용성을 검증하는 용도로 활용하는 것이 바람직하다고 생각한다.

- 액션다이아그램(프로세스와 프로시저)

위의 기술들은 분석한 프로세스 로직을 정의하고 정리하는데 기본적인 기술이다. 동작명령과 문법을 가지고 문장화하여 표현한다. 특별히 인간의 언어에 가깝고 간단한 기호를 가지고 표현하기 때문에 읽고 이해하기가 쉬우며 이전 방법론에서 많이 나타났던 문법적 에러를 발생하지 않도록 해준다. 그러나 프로그램을 많이 짰 분석가는 코딩 정도로 가볍게 생각하기도 하며, 지원 툴마다 설계단계에서의 프로시저 구성방법이 다르기 때문에 혼란을 겪기도 한다.

- 다이얼로그 흐름도/온라인 대화 설계

이 기술은 프로시저나 프로시저 스텝을 위해 트랜잭션 화면이 나타내지는 순서를 제어방법을 정의하는 것이다. Navigator의 설계 Phase와 IEM의 BSD Stage에서 이루어지며 차이는 메뉴를 통해서 “Front-end”사용자를 위한 화면 트랜잭션의 Trigger방식에 있다.

- 클러스터분석

이 기술은 IEM ISP, BAA Stage에서 적용된다. 매트릭스를 통해 동일한 실체를 다루는 프로세스의 그룹화/클러스터화를 하는 것이 이 기술이며 별도의 알고리즘을 기준으로 하여 수행된다. 대표적으로 ISP와 BAA Stage에서 수행하여 비즈니스영역과 비즈니스시스템을 정의하는데 사용된다. IEM이 CASE 툴의 지원을 받아야 하는 부분으로 Review를 통해 관계값을 정의하고 그룹화/클러스터화를 통해 조정하여 반영시킨다.

기술적 표준을 마치기에는 아직 남아 있는 사항이 많이 있지만 전체적인 상호연관성을 가지고 접근하면 몇가지 제외하고는 기능과 의미가 분명해지리라고 본다. 지금까지 우리는 IEM과 Navigator에 대한 비교해 보았는데 두 방법론이 정보공학에서 어떤 위치에 있는지에 대해 재조명해 볼 필요가 있다.

2.2 정보공학 변천과정

구조화된 설계방법의 변천과정은 그림 1.3에서 보여주듯이 여섯개의 단위 흐름이 있고 특히 두 흐름은 BIS와 James Martin계로 양대 주류를 이루고 있다.

BIS는 영국에 근거를 둔 소프트웨어 하우스와 컨설팅업체로 1970년대에 MODUS라는 지금의 구조화된 방법론을 개발했다. BIS에는 Roger Latham and Burchett Management System (LBMS)를 만들고 MODUS를 확장해서 LSDM이라 불리는 방법을 개발했다. 또한 LSDM은 영국정부의 표준설계방법을 만드는 Central Computing and Telecommunication Agency (CCTA)에 의해 선정되었다. 그 방법은 Structured System Analysis and Design Method (SSADM)이라 불렀다. SSADM은 정보시스템이

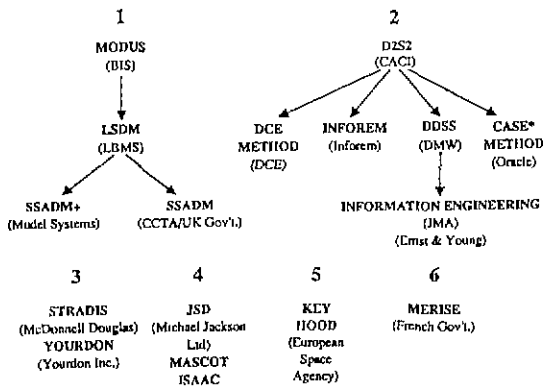


그림 1.3 구조적 설계방법의 변천과정

나 어플리케이션 시스템개발방법이 아니다. 그것은 프로그램 정의나 주요성공요인상의 기술을 포함하지 않고 있다.

또한 주요한 하나의 흐름은 James Martin 흐름이다. 시작점은 James Martin과 Clive Finkelstein에 의해 “정보공학”이라고 명명된 발행물이었다. 이것은 1981년 Savant Research Report로 발간되었다. 이것과 비슷한 정보공학은 Development of Data Sharing System(D2S2)라는 방법을 CACI에서 개발하고 있었다. CACI는 미국의 정보시스템 컨설턴트 회사로 영국에 사무소를 가지고 있었고 ORACLE 데이터베이스 제품에 대한 판권을 가지고 있었다, ORACLE팀은 CACI를 떠난 후 ORACLE 회사에서 D2S2를 사용해서 CASE*METHOD를 개발했다. CACI의 일원이었던 Tony Carter는 Doll Martin Worldwide에서 DDSS방법을 개발했다. 그리고 Doll Martin Worldwide는 James Martin Associates로 합병되어 DDSS 방법론을 “Information Engineering Method”라고 불리우던 JMA 방법과 결합했다. IEM은 IEF(Information Engineering Facility)라는 Texas Instruments로부터의 툴을 가지고 통합된다. 그리고 동시에 James Martin은 DDI(=Database Inc=Knowledgware)와 함께 활동하고 있었으며 또한 비슷한 방법론 철학을 연구하고 있었다.

같은 시기에 James Martin은 경영 컨설턴트인 Arthur Young의 국제적인 기업과 함께 손잡고 소프트웨어 개발회사인 Knowledgware에서 또

하나의 정보공학에 근거한 방법을 만들었다. 그 방법론은 1990년에 확장되어 Navigator System Series라 불렸고 후에는 Navigator라 불렸다. Knowledgware CASE툴인 IEW/ADW를 만들었고 정보공학 Dictionary/Encyclopedia의 지원이 표준화될 것이라고 믿고 있었다.

국내에 개발방법론 및 CASE 툴이 보급되기 시작한 것은 IBM의 AD/Cycle이라는 Framework이 발표된 1990년말부터 시작되었으며 본격적인 프로젝트에의 적용은 IEF에 의해서 시장을 형성하게 되었다. 필자가 참여하게 된 시기도 비슷하여 회상해 본다면 IBM의 AD/Cycle에 근거한 Repository 전략이 변경되면서부터 혼란을 겪었던 것 같다. 그러면 국내에 도입된 IEF와 ADW를 중심으로하여 CASE 툴에 대해서 알아보고 선정시 고려해 보아야할 관점을 생각해 보기로 한다.

2.3 정보공학방법론을 지원하는 CASE 툴

위에서 서술한 것처럼 정보공학방법론을 근거한 대표적인 CASE 툴은 Texas Instrument의 IEF와 Knowledgware의 ADW가 있다. 이 두 툴은 태어난 시기가 비슷하고 James Martin에 의해 만들어졌고 Encyclopedia에 기반을 두며 나름대로 지원하는 방법론을 가지고 있다. 그래서 CASE 툴은 자신이 지원하는 특별한 방법론을 가지게 되며, IEF는 IEM을 ADW는 Navigator를 지원하는 통합된 툴이다. CASE 툴을 보통의 스프레드 시트나 워드프로세서와 같은 그래픽 툴 취급을 해버린다면 그 가치를 상실해 버리는 결과를 가져올 것이다. 물론 CASE 툴이 요술방망이와 같은 취급을 하는 것도 어리석은 생각이라고 아니할 수 없다.

2.3.1 CASE 툴의 특성

CASE 툴은 몇가지 주요한 특성을 가진다. 첫째로는 반드시 방법론을 지원하고 있다는 것이다. 자신이 지원하는 방법론이 없이는 결코 존재할 수 없으며 단지 프로그램을 생산하는 공정 일부분을 도와주는 것뿐이다. IEF는 IEM을 지원하고 ADW는 Navigator를 지원하고 있다. 그

래서 정보공학방법론과 CASE 툴이 완전히 일치하지 않는다면 서로 절충하여 사용할 필요가 있다. 대체로 시스템적 제한사항을 극복하기 위해 툴 중심으로 해결하는 경향이 있다.

둘째로는 정보저장소(Repository, Encyclopedia)를 가지고 있다는 것이다. 정보공학방법론과 CASE 툴을 통해 계획, 분석, 설계된 모든 객체들을 체계적인 관리형태의 데이터베이스로 구축하여 관리해 나간다. 이것은 특별히 방법론에 근거하여 진행상황을 관리하고 보다 효율적으로 프로젝트를 운영할 수 있도록 한다. 품질보증을 위한 활동도 정보저장소를 이용하여 분석하면 손쉽게 결과를 얻을 수 있다. 그러나 체계적인 관리를 위해서는 메타모델(방법론에 근거)을 이해하고 있는 데이터베이스 기술자를 지정하여 운영과 보안관리전략도 수립하여야 한다.

셋째로는 매트릭스 분석기능을 갖추고 있어야 한다. 정보공학방법론이 CASE 툴을 전제하고 있는 이유이기도 하다. 매트릭스를 통한 분석은 Review를 하여 관계값을 정의하고 이것이 특별한 클러스터화 알고리즘을 통해서 클러스터화될 때 효용성이 있는 것이다. IEF와 ADW는 매트릭스 기능을 잘 지원해 주고 있으나 IEF는 클러스터링이 뛰어나고 ADW는 관계값의 정의를 다양하게 할 수 있다. 그리고 매트릭스는 종류가 다양해서 용도와 의미를 잘 파악하여 선택 후 수행해야 한다. 비즈니스 업무영역과 비즈니스 시스템 정의시에는 반드시 실체/기능, 실체/실체, 기능/기능 매트릭스의 친화성 분석을 통해 결정하는 것이 바람직하다.

넷째로는 액션다이아그램을 기호로 지원한다는 것이다. 이것은 비즈니스분석을 통해 업무를 정리하여 로직을 정의하는 도구로서 사용자 언어에 가까운 명령어와 기호로 구성되어 있다. IEF에서는 프로세스 액션도와 프로시저 액션도가 있으며 ADW는 미니스펙과 모듈 액션도가 있다. 두 툴의 기능을 비교해 보면 의미와 역할은 같지만 재사용성 측면과 명령어가 사용자에게 얼마나 친숙하나라는 측면에서는 IEF가 더 뛰어나다고 볼 수 있다.

이상으로 몇가지 특성을 알아 보았지만 우리가 잘 활용할 수 있기 위해서 어떤 툴을 선택해야

할지 신중히 검토해야 한다. 그래서 우리가 CASE 툴을 선정할때 고려해야 하는 관점을 생각해 보기로 한다.

2.3.2 CASE 툴의 선정시 관점

우리는 정보공학방법론과 CASE 툴을 사용하는 목적이 소프트웨어 생산성에서 고객의 정보 전략과 정보를 체계화하고 관리하기 위함으로 패러다임이 전환되고 있다고 위에서 서술했다. 이 패러다임의 전환에 따라 CASE 툴 선정시 우리의 관점도 코드의 자동생성에서 정보전략과 정보를 체계적으로 수립하고 관리해 나갈수 있는 Encyclopedia로 전환되어야 한다. 아직 완벽하게 구축된 Encyclopedia를 가진 CASE 툴은 없지만, PACBASE(국내에는 도입되어 있지 않음)와 IEF가 강점을 가지고 있다. 그러나 그 Encyclopedia가 강점이 있다지만 우리는 무엇으로 그것을 평가하는가? 우리는 먼저 정보저장소(Repository, Encyclopedia)의 논리적 메타모델을 확인해 보아야 한다. 그리고 각 Stage간의 모델관리전략을 확인하고 작업운영방식(Checkin/Checkout, Subset 등)이 얼마나 효율적으로 이루어져 있는지를 검토해 보아야 한다. 가장 중요한 기능은 버전통제, 변경관리와 형상관리를 꼽을 수 있는데 아직은 버전통제와 영향도 분석정도의 수준이다. 그러나 향후에는 기능이 충분히 확장되어 프로젝트관리와 품질보증 활동도 포함될 것으로 기대된다.

정보공학방법론은 데이터지향적인 방법론이다. 그러므로 CASE 툴 선정시 데이터를 분석하는 기능부분이 다른 기능보다 뛰어난지 확인해야 한다. 통합 CASE 툴은 아니지만 데이터를 분석하고 데이터베이스를 구축하는 단계까지의 기능을 강화한 BACHMAN과 같은 툴도 있다. 그래서 위의 두가지 관점을 충분히 만족시켰다면 전체적인 정보공학방법론과 CASE 툴이 조화를 잘 이룬다고 판단할 수 있다. 특히 방법론과 CASE 툴이 서로 다른 회사에서 제공된다면 더 상세히 검토해 볼 필요가 있다. 그리고 선정된 후 새로운 개발문화의 변화에 대처할 수 있도록 많은 교육과 적극적인 수용이 요구된다.

2.4 고객지향적 정보공학방법론과 CASE 툴로의 전환

요즘 몇년간 유행어가 된 “비즈니스 리엔지니어링”과 함께 경영의 혁신적 변신을 하는 활동이 전개되고 있다. 경영전략은 이제 새로운 고객- 기존 고객의 새로운 정의, 을 정의하고 그 고객을 위한 활동으로 변신하고 있다. 그 활동은 오직 고객만을 위한 기업내에서의 유일한 활동이며 최대의 가치를 창조하기 위해 서로 조화를 이루며 노력해야 한다. 우리는 이제 경영전략을 수행하는 경영자와 그와함께 활동하는 사용자를 고객으로 재정의하며 방법론과 CASE 툴을 고객을 위해 잘 활용하도록 한다.

2.4.1 개발조직문화의 전환

지금까지 우리가 해온 일은 고작 프로그램 짜는 일, 즉 정보시스템을 만드는 일만 해 왔고 개발완료된 시스템의 사용에 대한 책임은 자금을 투자한 경영자와 사용자에게 돌아갔다. 이 고객들은 버그가 있어도 말을 못했고 그들이 요구하는 답이 나오지 않아도 불만을 제기하지 못했다. 그러나 이제는 “정보시스템 개발”이라는 말에서 “정보전략과 정보관리”라는 의미로 전환을 요구한다.

경영환경은 생산성을 논하다가 품질을 논하는 시기가 있었고 이제는 고객지향의 벤치마킹, 비즈니스 리엔지니어링, 해방경영의 시대가 도래했다. 고객의 정보전략과 정보를 관리해 주는 우리의 환경도 고객을 위한 활동중심의 조직문화도 해방되어야 한다. 아직도 프로그래머가 고객의 정보전략계획을 수립하고 데이터를 정의하여 데이터베이스를 설치하고 코딩/테스트하여 시스템을 제공하고 있는가? 소프트웨어 생산성을 화제도 삼던 시대는 지났다. 품질을 강조하는 시대도 지났다. 품질은 당연한 것이다. 이제 고객지향적 활동으로 해방되어 수행한다면 전에 고민하던 것이 자연스럽게 해결이 되고 정보공학방법론과 CASE 툴도 그 역할을 다할 것으로 기대할 수 있다.

2.4.2 원칙의 준수

정보공학방법론과 CASE 툴을 잘 이용하기 위해서는 방법론에서 제시하는 지침과 표준을 잘 준수해야 한다. 서양의 합리주의 사고에서 나온 정보공학방법론은 지원하는 CASE 툴이 아무리 훌륭해도 수없이 많은 Review를 통해서 공동된 의견을 도출하기 때문에 귀찮고 지루한 작업이다. 특히 프로그래밍 경험을 많이 한 사람일수록 정통적인 원칙보다는 지름길을 많이 찾고 응급조치적인 작업수행을 하게 된다. 고객을 위한 활동에는 지름길이 없다. 영원한 고객을 만들고 만족시키기 위해서는 장기적인 정보관리체계를 구축하고 꾸준히 노력하는 수밖에 없다. 이때 정보공학방법론과 CASE 툴은 우리의 활동을 정의하는데 도움을 주고 수행해 나가는데 힘이 될것이다.

우리는 정보공학방법론의 데이터지향적이란 전제사항을 깊이 생각해 볼 필요가 있다. 고객이 우리에게 요구하는 욕구는 자신의 정보전략과 정보에 대한 관리이다. 즉 데이터관리는 정보전략을 데이터베이스관리는 정보를 관리한다고 보면 적절할 것 같다. 이 데이터관리자와 데이터베이스관리자는 기능중심과 조직중심이라고 생각할 수 있겠지만 그것은 고객의 정보전략과 정보를 관리하는 활동으로 프로세스지향적이라고 볼 수 있으며, 모든 조직이 깨어진 후 기업의 정보에 관심을 가지고 지속적으로 관리하려고 노력하는 사람들을 모아서 독립적으로 책임을 가지고 수행할 수 있도록 해야 한다. 이때 데이터관리 활동의 하나로 정보전략을 관리하는 정보저장소에 대한 관리도 강조된다. 정보저장소 관리역할을 수행하려는 사람은 반드시 정보공학방법론을 충분히 이해하고 있어야 하며 물리적인 구조와 DBMS 기술도 습득하고 있어야 한다.

국내에서 정보공학방법론을 사용해본 기업들을 조사해 보면, 방법론과 CASE 툴을 정보시스템을 만드는 도구로써 혹은 프로그램을 자동생성하는 도구로 생각하고 출발하여 대체로 효과를 얻지 못했다고 한다. 정보공학방법론과 CASE 툴을 이용해 활동을 성공적으로 수행하기 위해서는 컴퓨터 시스템적 사고를 버리고 고객의 정보전략과 비즈니스적 관점만을 생각하며 수행해 나가야 한다. 그리고 가내수공업적 조직을 없애

고 고객의 정보전략과 정보를 관리하기 위한 프로세스중심적 환경을 만들어 고객을 위한 활동을 하고 있는 직원들에게 믿고 맡겨야 한다.

3. 결 론

정보공학방법론과 CASE 툴은 동반자적 관계를 가지고 지속적으로 발전해 가고 있다. 그러나 모든 방법론과 CASE 툴이 그러하듯이 강점과 약점을 가지고 있어 사용자의 융통성을 가진 활용이 강조되며 자체적인 방법론을 설정하여 독자적인 문화를 구축하려한다면, CASE 툴의 제한사항을 극복하고 정보공학방법론의 근본적인 원칙을 준수하는 것이 바람직하다. 그리고 정보 저장소의 메타모델을 확인하여 철저히 관리할 수 있도록 전략을 수립하고 운영해야 한다. 프로젝트관리와 품질보증에 대한 방법은 별도로 구축하여 활동을 효율적으로 할 수 있도록 한다.

우리의 고객만족을 위한 활동은 지속적으로 이루어지며 특별히 최초의 프로젝트였다면 그 평가는 유지보수시점에 이루어질 것이다. 소프트웨어 향상을 위한 패러다임에서는 유지보수를 하는 활동이 요구되지만 고객지향적인 패러다임에서는 경영전략의 변화에 대한 신속한 수용과 기존에 운영되고 있던 정보전략과 정보와의 재사용과 통합에 대한 문제를 해결해야 한다. 그러나 타입종으로 전환하여 고객을 완전히 재정의하려고 할 때는 최초의 프로젝트와 같이 큰 변화를 수용해야 할 것이다.

정보기술과 경영환경은 정신을 못차릴 정도로 급속히 변화하여 가고 있다. 클라이언트/서버 컴퓨팅, 오픈시스템, GUI 등 정보기술의 발전은 경영환경을 자극하여 경영과 정보전략에 큰 영향을 미칠 수 있고, 비즈니스 리엔지니어링, 벤치마킹, 해방경영과 같은 경영환경변화는 또한 고객지향적 정보전략과 정보관리체제를 변화시킨다.

우리는 요즘 객체지향 패러다임에 대하여 전문가들로부터 많은 이야기를 듣고 있다. 그러나 새로운 패러다임으로의 전환을 위해서는 위기를 느끼고 새로운 목적가치를 부여해야만 한다. 우리가 가질 수 있는 가치기준은 고객이란 사실을

알았다. 고객들의 경영환경을 느껴보자. 고객의 경영환경은 해방경영 패러다임으로 전환되고 있으며 우리의 정보환경도 정보공학방법론을 거쳐 객체지향 패러다임으로 전이를 꾀하고 있다. 그 근거는 우리 모두 고객을 중심으로 한 활동과 기존 조직으로부터 해방되고 데이터를 객체의 개념으로 활동영역을 넓혀 간다는 사실이다.

필자는 이 리포트를 통해 정보산업에 종사하는 우리들의 깊은 반성을 촉구한다. 우리는 정보기술의 맹신자가 아니며, 우리 또한 고객의 진실된 욕구를 만족시켜야 하는 사명을 가지고 있다는 것을 깨닫자. 우리도 진정 고객이 무엇인지? 고객이 누구인지? 정의하고, 고객을 위한 활동이 무엇인지를 찾아 성실히 수행해 나가야 한다. 우리는 조직으로부터 해방되어 고객지향적 정보공학방법론과 CASE 툴을 통해 고객의 경영전략을 지원하는 정보전략을 수립하고 정보가치를 높일 수 있는 관리활동을 해나가야 할 것이다.

참고문헌

- [1] John S. Hares, "Information Engineering For The Advanced Practitioner", John Wiley & Sons, 1992.
- [2] James Martin, "Information Engineering" I,II, III, Prentice Hall, 1990.
- [3] Tom Peters 저, 노부호외 5인역, "해방경영", 한국경제신문사, 1994.
- [4] Von Halle, B. & Wasim E Rajput, "Back to Business Rule Basics", Database Programming and Design, 15-16, October, 1994.
- [5] Von Halle, B. & Wasim E Rajput, "Whole lot of shaking Going on", Database Programming and Design, 13-15, August 1994.

이 재 관



1979~1986 성균관대학교 물리학과 학사
 1987~1991 포항종합제철(주) 평양제철소 전산시스템부 근무
 1992~1993 포항종합제철(주) 본사 전략정보시스템 구축 프로젝트 참여
 1994~현재 대우정보시스템(주) 정보기술부 근무

관심분야: CASE, 데이터모델링, 데이터베이스