

비즈니스 프로세스 서술도구 비교: 은행 자산건전성 평가업무 중심으로 안현섭^a, 김용재^b, 함유근^b

^a KAIST, 경영공학 박사과정, coolahnphd@gmail.com

^b 건국대학교 경영대학, 서울광진구 화양동1, 02-450-3608 {yjaekim@gmail.com,ykhahm@konkuk.ac.kr}

논문요약

정보시스템의 발전과 함께 현실업무가 더욱 복잡해지면서, 정보시스템 개발 초기단계에서 사용자의 요구사항이 누락되거나 중복되는 경우가 자주 발생하며, 이에 따라 차후에 이를 수정하기 위한 노력이 증대되고 전체개발비용과 기간이 증가하게 된다. 이 문제를 해결하기 위해서 업무 프로세스를 표현하기 위한 다양한 방법론이 제안되어왔지만, 각 방법론의 우열을 어떻게 가릴지에 대한 체계적인 항목별 연구는 이제까지 없었다. 이 연구에서는 사용자가 직관적으로 이해할 수 있고, 분석가가 업무 프로세스를 충분히 표현할 수 있으며 동시에 시스템구축 비용을 최소화할 수 있는 방법론이 가장 이상적이라는 점을 고려해서, 누락도, 중복도, 명확도, 표현력 등의 방법론 평가기준을 제안하며, 이 지표를 은행의 자산건전성 평가업무라는 표준업무에 적용해서 각 방법론의 순위내지는 문제점들을 살펴본다. 프로세스 중심 기법인 자료흐름도와 유즈케이스 다이어그램과 비교할 때, 데이터 중심 기법인 개체관계도가 상대적으로 낮은 함축성과 높은 복잡성에도 불구하고 낮은 누락도와 명확한 표현력을 가지고 있으며, 업무기술서와 병행될 때 프로세스 함축과 누락에 대한 부담을 크게 감소시킬 수 있을 것으로 나타난다.

Abstract

In developing information systems, omissions and redundancies of user requirements are too frequent to increase development cost as they need to be remedied at later stages. In order to cope with this problem, various tools describing business processes have been proposed in the past and yet fewer attempts have been made to systematically and categorically evaluate performance of individual tools describing business processes. In this paper, we offer a general framework for appraising competing tools for describing business processes, including ERD, DFD, and Use Case Diagram in terms of ease of use, completeness, chance of having omissions, and . Based on a exemplary business From the measures proposed in this paper for comparing descriptive tools, ERD turns out to stand out over other often-used methodologies. Despite its problem with poor readability, ERD proves to offer relatively higher expressive power and ease of use. Being

supplemented by well-formed job descriptions, ERD would remain a better choice in developing business information systems.

주제어: 정보시스템, 요구사항 분석, 개체관계도, 자료흐름도, 유즈케이스 다이어그램, 업무기술서

1. 연구 소개

1.1 연구 배경

정보시스템 개발에 선행되는 사용자의 의견수렴 과정에서 나타날 수 있는 업무 프로세스의 묘사가 부정확하게 되면, 정보가 누락하게 되고 이로 인해서 의사소통이 추가적으로 필요하게 되면서, 개발기간이 과다하게 늘어나게 된다(Berente and Lyytinen 2005; Glass 1997). 사용자가 자신이 수행하는 업무를 시스템 분석가에게 설명하는 과정에서 빠짐없이 나열하지 못할 경우, 시스템 구축을 위한 인터뷰는 여러 차례 수행해야 하며, 이 과정에서 업무에 필수적인 데이터와 프로세스에 대한 정보가 계속 누락되거나 혹은 중복되거나 아니면 요구사항 자체가 왜곡된 채로 전달되기도 한다(Emam and Madhavji 1995). 분석가는 의견수렴 과정에서 발생하는 부작용을 제거하기 위해서 여러 차례 실무자와의 인터뷰를 통해서 프로세스에 대한 명확한 정보를 수집할 수 밖에 없는데, 이 과정이 비효율적으로 반복되면 정보시스템 개발 프로젝트의 개발비용과 기간은 원래의 목표를 크게 상회하게 된다(Pressman 2004).

이에 따라, 정보시스템 구축의 다섯 단계인 계획-분석-설계-구현-시험 (유치수 외 2명 1998) 중 가장 중요한 분석 단계에서 사용자 요구사항을 명세(明細)화 하기 위한 다양한 방법론들이 제시되고 활용되어 왔다. 예를 들어, Yourdon (1989)은 기업 업무를 프로세스 모델링 중심으로한 보고서 하는 구조적 방법론을, Valacich et. al (2005)은 구조적 방법론의 단점을 보완하기 위한 프로토타입 기법을, Brinkkemper (1996)는 데이터 모델링을 중심으로 정보공학 방법론을, 그리고 Booch (1986)는 데이터와 프로세스를 함께 모델링하는 객체지향(object-oriented) 방법론 등을 제안해왔다. 한편 이들 방법론을 큰 축으로 삼아, 업무의 인과관계와 시간순서를 보여주는 다이어그램을

사용하여 설계자와 현업의 실무자들의 의사소통을 돕고자 하는 시도가 있었다(Schriber 1969). 그러나 사용자의 요구사항을 명확히 정의하려는 설계단계에서의 오류가 적지 않은 현실에서 미루어 볼 때, 현재 활용되고 있는 IDEF (Godwin *et. al* 1989), IRL (Huckvale and Ould 1995), 개체관계도(Chen 1976), 자료흐름도(Kettinger *et. al* 1997)와 같은 방법론들이 사용자의 요구사항을 명확히 기술하고 반영하는데 효율적이지 못할 뿐 아니라, 방법론 자체가 가지고 있는 한계점들로 인해 업무 프로세스를 정확히 묘사하지 못하고 있다.

따라서 사용자 요구형성 과정을 최소화하면서—즉 인터뷰 횟수를 줄이면서— 동시에 전사적 관점에서 프로세스를 더 명확하게 기술할 수 있도록 기존의 방법론들을 보완하려는 노력이 시급하다고 할 수 있다. 만약 정보시스템 개발 프로젝트에서 개별 부서와 인원 단위로 프로세스 분석 작업이 선행된다면, 데이터와 프로세스에 대한 통합적 시각의 부재로 데이터와 프로세스의 중복과 불일치가 필연적으로 발생하게 된다(문송천 2004). 사용자와 분석가의 의사소통에서 발생하는 이런 문제를 최소화하기 위해서, 시스템 분석가에게는 기업 업무 전체에 대한 종합적 관점을 수용할 수 있고 사용자의 요구를 효율적으로 형성할 수 있는 도구가 필요하며, 실무자에게는 분석가가 제시한 결과를 놓고 자신들이 요구한 사항들이 빠짐없이 기술(記述)되었는지를 명확하게 검증할 수 있는 방안이 제시되어야 한다. 결국, 분석가와 사용자 모두를 위해서 효율적인 의사소통 도구로서의 프로세스 묘사 기법에 대해서 심층적으로 탐색해 볼 필요가 있고, 기존에 사용되는 방법론들을 보완할 수 있는 방안들이 제시된다면 정보시스템 분석 설계 단계에서 발생하는 부담을 크게 줄일 수 있을 것이다.

1.2 연구 동기 및 목적

정보시스템 분석 단계에서 사용자 요구사항을 수렴하는데 있어서 발생하는 비효율은 대개 의사소통 반복에 따른 분석시간의 과다소요에 기인한다. 따라서 분석업무의 반복을 감소시키며,

동시에 사용자 요구사항을 최소한의 시간 내에 얻을 수 있는 방법이 모색되어야 한다. 사용자의 행위와 관계된 데이터의 묘사가 정확히 묘사되지 못하면, 분석가는 반복적으로 인터뷰를 해야 하며 개념적 설계단계로 넘어가지 못하게 된다. 만약 분석 결과물이 구현하고자 하는 시스템의 데이터 개체와 개별 프로세스의 소단위(小段位)를 빠짐없이 나타내 준다면, 분석업무의 반복을 줄일 수 있고 시스템 분석에 들어가는 소요 기간과 비용을 절감시키는 효과를 가져 올 수 있다. 하지만, 요구사항이 자주 변경되고 이에 따라 시스템 분석 결과가 수정되어야 한다면, 데이터 설계 시 개체 개수의 변경 및 묘사 프로세스 리스트의 증감을 가져오고, 이 변경사항을 유지하기 위해서 사용자와 분석가 간의 의사소통 업무 반복이 발생하게 된다. 예를 들어, <표 1>은 A기업 정보시스템 구축 프로젝트에서 관찰된 분석업무 과다소요로 인한 업무지연의 심각성을 놓고, (Benediktsson *et. al* 2003)이 제시한 COCOMO모형으로 추정해 본 결과이다.

표에서 각 프로젝트는 개발업무의 복잡도에 따라 32KDSI(The Number of Thousand of Delivered Source Instructions)와 128KDSI로 구분되어 있다(Cherf 1992). 만약 요구사항이 명확하게 표현되고 한번에 수렴될 수 있다면, 각각 14.7개월과 27.1개월만에 프로젝트가 완수될 수 있을 것이다. 그러나 분석을 1회 반복 실시할 경우 프로젝트 개발 기간은 18.1개월과 34.8개월로 각각 23%, 28%의 기간이 추가적으로 늘어나며, 3회의 분석업무가 요구될 경우 24.9개월과 50.2개월로 각각 69%, 85%가 추가로 증가하게 된다. 여기서 우리는 2회 분석에 비한 3회 분석의 추가증가비율은 46%와 57%로, 최소한 A기업의 경우, 추가로 늘어난 기간은 비선형의 형태를 보이는 것을 알 수 있다. 따라서 분석업무의 반복을 피하고 나아가 프로젝트 비용을 통제하기 위해서, 사용자의 요구사항을 효율적으로 관리할 필요가 있다.

사용자 요구사항을 명확하게 하기 위해서 업무 프로세스를 너무 자세하게 묘사하면, 분석가와 실무자 모두가 처리해야 할 정보량과 복잡성이 증가하게 되기 때문에, 현실을 간단명료하게 그러나 정확하게 묘사할 수 있는 기법이 요구된다. 기존에

<표 1> 정보시스템 분석 기간의 증가 추이 사례 (단위: 월)

활동	분석-설계 1회 수행		분석-설계 2회 반복		분석-설계 3회 반복	
	32KDSI	128KDSI	32KDSI	128KDSI	32KDSI	128KDSI
계획/분석	1.2	3.1	2.4	6.2	4.8	12.4
설계	2.2	4.6	4.4	9.2	8.8	18.4
코딩/단위 테스트	7.7	12.2	7.7	12.2	7.7	12.2
시스템 테스트	3.6	7.2	3.6	7.2	3.6	7.2
총 소요기간	14.7	27.1	18.1	34.8	24.9	50.2

제시된 정보시스템 개발 방법론들이 다이어그램이라는 단순한 표현 방식을 다양하게 변형해서 기업 업무 프로세스를 표현하는데 사용해진 이유는 복잡한 업무일수록 그림을 통한 간단한 묘사가 더욱 효과적이었기 때문일 것이다. 그러나, 사용자 요구를 과도하게 단순화시킨다면 전달하고자 하는 정보가 손실되는 정보누락이 발생한다. 현업 실무자의 암묵적 지식을 형식지로 표현할 경우, 문장 형태로 표현할 경우에는 상세한 묘사가 가능하지만(Hull et. al 2005), 다이어그램이나 플로우차트(flow chart)와 같은 축적인 형태로 변환하는 과정에서 기술되어야 할 내용이 생략될 수 있다. 만약 정보누락이 지속적으로 발생한다면, 추가적인 시간과 자원이 이를 수정하기 위해서 들어갈 수 밖에 없다. 예를 들어, <표 2>는 K은행의 정기예탁 상품 업무에서 어떻게 정보가 누락되는지를 보여 준다. 비록 단순하고 이해가 쉬운 표현일지라도 <표 2>의 다이어그램에서는 현업 업무 프로세스에 중요한 정보가 누락되어 있다. 은행의

묘사할 경우 업무에 관한 정보누락이 발생하게 되며, 명확한 업무 파악을 위해 추가적인 의견수렴 과정을 필요로 하게 되고, 결국 업무기간이 과다하게 늘어나게 된다. 따라서 정보의 누락을 허용하지 않고 동시에 중복되는 작업을 피하면서, 사용자의 업무를 간단명료하게 묘사할 수 있는 기법에 대한 심층적 평가가 필요하다. 우리는 이 논문에서 기존 기업 프로세스 묘사 기법들을 평가하기 위한 기준으로 용이성, 묘사 관점상 특징, 방법론적 온전성, 그리고 적용편이성(便易性)의 4개의 객관화된 지표를 제시하여, 기존에 제시되었던 기법들의 장점과 문제점을 논의한 다음, 이를 보완하기 위한 방법을 제시해 본다. 나아가 그림을 활용하여 기업 업무를 묘사할 때, 어떤 기법들이 프로세스 묘사 기법을 사용하는 사용자의 학습과정을 최소화 하면서 동시에 명확한 표현을 가능하게 할 것인지를 탐색해 본다.

<표 2> 프로세스 단순화 과정에서 발생하는 정보 누락의 예

구분	도출 결과	비고
프로세스 묘사 내용	“K은행의 고객은 은행에 매월 약정금액을 보험과 예금의 결합상품인 ‘KA’ 상품계좌에 적립한다. 은행은 20년 동안 6.5%의 복리이자를 고객에게 매월 지급하고 매 분기 결산일 마다 이자소득의 10%를 세금으로 차감한다”	문장 형태의 정보를 문장으로 기술
다이어그램으로 단순화시킨 프로세스		도형요소들을 활용한 직관적 기술
표현된 정보	1. 고객은 은행에 약정금액을 입금한다 2. 은행은 고객에게 이자를 지급한다 3. 고객과 은행 사이에 세금차감이 발생한다	3개의 문장으로 프로세스를 표현
손실된 정보	1. 업무의 순서 (입금-이자지급-세금차감) 2. 이자율 (6.5%) 3. 세금차감 방식 (은행이 고객계좌에서 직접 공제) 4. 과세율(10%) 5. 상품명(방카슈랑스, KA) 6. 이자 산출 형태(복리/단리/기타) 7. 이자 지급 시기 (3,6,9,12월의 25일)	프로세스를 표현한 문장에서 다이어그램으로 변환시킬 때 손실된 정보

예탁업무는 고객과 은행 사이에서 계좌를 중심으로 발생하는 입금, 이자 지급 및 세금 차감 프로세스가 통합된 형태이지만, 다이어그램에서는 화살표로 나타내지 않은 고객과 은행 사이에 세금차감이 실제로 어떤 방법으로 이뤄지는지 파악할 수 없으며, 이자율과 과세율 그리고 이자 산출 형태, 지급 시기 등과 관련된 최소 7 개의 정보가 손실되어 있음을 알 수 있다.

결국 적절한 기준 없이 업무 프로세스를 단순화시켜

이후 본 논문의 구성은 다음과 같다. 2장에서는 대표적인 업무 프로세스 묘사 기법을 비교 검토하면서 각 기법이 지닌 특징과 제한점을 살펴보면, 우수한 기법을 가려내기 위한 기준을 제시해 본다. 3장에서는 효율적인 프로세스 묘사 방법론을 가려내기 위한 기준을 간략히 검토한 후, 표준 업무에 기반하여 2장에서 선정된 3가지 대표 프로세스 묘사 기법인 자료흐름도(DFD), 개체관계도(ERD), 유즈케이스 다이어그램(UCD)을 실제로 적용해 본다. 4장에서는 도출된 결과물의

차이점을 상세히 분석해 보고 방법론간의 특징과 우수성을 정량화된 척도로 비교 분석한 다음, 5장에서는 비교결과를 기초로 효율적인 묘사 기법의 요건을 재확인하고 방법론 자체가 가진 한계점을 보완하기 위한 방안을 탐색해 본다.

2. 관련 연구

추상적인 기업 업무 관행을 명시적인 도형요소로 표현하기 위해서 일반적으로 사용되는 다이어그램이나 플로우차트는 정보의 일부를 생략하는 과정에서 중요한 정보를 누락시키기 쉽다. 두 표현 방법 모두 단순한 도형과 도형에 부여된 의미를 사용해서, 기업 업무에서 발견되는 요소와 흐름을 표현하지만, 도형이나 기호에 특별한 의미를 부여하는 방식에서 그리고 어떤 방식으로 기술하는지에 대해서는 방법론마다 차이가 나며, 지나친 단순화 때문에 묘사하고자 하는 프로세스에 대한 주요정보가 누락되거나 왜곡되어 전달되기가 쉽다. 따라서 어떤 관점에서 추상성을 줄이면서 동시에 명확성을 제고시킬 수 있느냐가 프로세스 묘사 기법상의 우열을 가려내는 기준이 된다. 우선 프로세스 묘사 기법은 사용자 관점에서 자신의 업무를 얼마나 쉽게 표현해 낼 수 있는지, 그리고 분석가의 이해 과정을 거치고 정보시스템 개발 단계로 전환되었을 때, 최소한의 정제화 과정만을 가지고도 실제적인 형상화가 가능한지로 분류할 수 있다. 우리는 이번 장에서 용이성과 적용편이성이라는 두 가지 측면을 가지고 대표적인 분석기법 6가지가 가지는 한계점을 비교해 본다.

어떤 특정 기법의 적용편이성이 높더라도, 업무 프로세스를 묘사하는 과정에서 사용자가 이해하기 어려운 규칙을 사용하거나 중요한 정보가 다량 누락된다면, 용이성 측면에서 우수한 기법으로 평가 받기는 어렵다. 따라서 효율적 프로세스 묘사기법 판별은 사용자 측면에서의 용이성과 분석가 측면에서의 적용편이성의 두 관점이 균형을 이루는 것이 바람직하다고 하겠다. 아무리 사용자 관점에서 명확한 묘사가 이뤄졌다 하여도, 시스템 분석가가 실제 업무에 활용하기 위한 데이터베이스 시스템으로 전환시키는데 상대적으로 많은 노력이 소요된다면, 전체 프로젝트에 들어가는 비용과 시간이 비례하여 증가할 수 밖에 없다. 반대로 적은 노력을 가지고 사용자의 요구사항을 시스템에 구현할 수 있다 하더라도, 기본적으로 사용자가 이해하고 받아들이기 어려운 의사소통 도구로 요구형성이 이뤄진다면, 분석단계에서 많은 시간이 소모되는 것을 피할 수 없게 된다.

우리는 이번 장에서 실제로 활용되고 있는 상용 데이터베이스 시스템이나 프로그래밍 언어를 통해 특정 방법론의 문제점을 조명해 보는 대신, 개념적

요구사항 분석을 토대로 각 방법론의 원리와 특성과 관련된 한계점을 도출하고자 한다. 따라서 기술적인 구현 방법에 대한 부분은 논외로 하며, 현업의 실무자 측면에서의 효율적인 요구형성에 초점을 둔다. 분석 기법 자체가 단순하여 적은 구성요소만을 가지고 효과적인 의사소통이 가능하며, 복잡한 업무에서도 간단 명료한 결과물을 도출해 낼 수 있다면, 사용자와 분석가 모두에게 적합한 업무 프로세스 묘사 도구로 평가할 수 있다. 따라서, 용이성 측면에서 프로세스 묘사 기법들을 간략히 분류·제시하고 각 방법론이 가지는 한계점들을 아래와 같이 정리·비교한다.

2.1 IDEF

IDEF 시리즈의 초기 시도인 IDEF0 기법은 업무 표현상의 높은 모호성을 보인다. 단 두 가지 표현요소로 그려지는 업무 모델링 결과물에서 사각형 상자로 표현되는 활동의 의미는 단순한 활동의 의미 외에도 운용, 변환과 같은 3개의 유사한 동적 개념으로 다르게 해석할 수 있다 (Mayer et. al 1995). 그래픽 요소인 화살표는 위치에 따라서 4가지(통제, 입력, 출력, 메커니즘) 다른 의미를 갖는데, 사각형으로 표현된 도형 요소와 화살표로 나타낸 의미 조합이 12가지로 표현될 수 있기 때문에, 이를 문장 형태로 다시 기술할 경우 정확한 의미를 전달하는 과정에서 정보가 왜곡되어 표현될 가능성이 존재하며, 명확한 의미 표현이 어려울 수 있다.

2.2 System Dynamics

Forrester (1961)에 의해서 제안된 System Dynamics 기법은 4가지 표현 도구를 활용하기 때문에 프로세스 표현에서의 정확도가 높지 않다. 기본적으로, 이 기법에서는 개체들이 다른 개체들과 어떻게 상호작용하는지 이해하는데 초점을 두고 있기 때문에 정적인 상태를 묘사하는데 그친다는 한계를 보일 수 밖에 없다 (Giaglis 2001). System Dynamics 기법에서는 Flow, Level, Rate, Parameter 라는 네 가지 표현도구를 통해서 업무를 표현할 수 있지만, 실제로 업무에 참여하는 개체들의 행위를 묘사하는 것은 기본적인 자원 흐름을 가리키는 Flow에서 그치고 있으며 이 묘사 범위도 지극히 제한적이다. 따라서 프로세스 묘사에 있어 고도의 상세한 접근을 시도할 경우 복잡성이 높아지기 때문에, 오늘날 이미 복잡해진 현업 환경에서 활용하는 데는 지극히 제한적일 수 밖에 없다 (Wolstenholme et. al 1993).

2.3 Role Activity Diagram

Role Activity Diagramming 기법도 업무를 묘사함에 있어서 표현도구의 과다로 사용자의 입장에서 활용

용이성이 낮다. 이 기법에서는 기본적으로 6개의 그래픽 요소를 활용하는데 프로세스, 행위, 그리고 그들간의 상호작용 내에서 개인이나 그룹의 역할을 모델링 하는데 초점을 두고 있다(Huckvale and Ould 1995). 다른 프로세스 묘사 방식과 다르게 프로세스 모델에 있어서 역할과 이에 대응되는 개념인 액티비티(activity)라는 개념을 적용한다. 그러나 대부분의 프로세스 묘사 기법들이 업무를 묘사하는데 있어 최소한의 도형 도구를 사용하는 것과는 대조적으로, 두 배 가까운 요소들을 별도로 사용함으로써 표현이 복잡하게 되어 사용자가 쉽게 이해할 수 없게 될 가능성이 높다. 또한 다른 기법이 사각형이나 타원과 같이 도식화 하기 쉬운 도형을 표현도구로 사용하는 데에 비해서, Role Activity Diagramming에서는 표현도구 간의 구성이 복잡하여 실제 업무에 구현했을 경우 가독성이 심각하게 떨어지게 된다.

2.4 자료흐름도 (Data Flow Diagram: DFD)

자료흐름도는 흐름도에 나오는 프로세스 레벨의 깊이만큼 반복적 분석이 선행되어야 한다는 한계점을 지니고 있다. 이 기법을 활용할 때는 설계자의 직관에 따라, 여러 하위 레벨로 이어지는 업무를 상세하게 분석하는 것을 목표로 자료흐름을 상세하게 기술한다. 분석 단계마다 레벨(level)이 형성되는데, 이를 통해 구조적 외부 개체, 내부의 처리 방법, 그리고 데이터 저장과 관련된 요소 등을 4개의 그림 도구를 활용하여 비즈니스 프로세스 내에서 묘사하고자 한다. 결국, 정보시스템 내부에서 데이터가 업무 흐름 내부에서 어떻게 생산되고 어디로 이동하는지 시스템 내/외 영역을 구분하여 이해할 수 있는 관점을 제시한다. 무엇보다, 기존의 기법과는 다르게, 이 방법은 처음으로 기업 내 업무의 액티비티나 컨트롤 대신에 데이터에 초점을 두고 있다는 점을 주목할 필요가 있다(Yourdon 1989). 그러나 데이터 흐름에만 초점을 둔 DFD는 불완전한 요구사항을 상세히 기술하지는 못하며, 일반적으로 Level 3까지 수행되는 업무 분석에서 최소 3회의 반복과정을 거쳐야 하기 때문에, 결국 사용자와 분석가의 업무를 지연시킬 가능성을 높이게 된다.

2.5 개체관계도 (Entity Relationship Diagram: ERD)

개체관계도는, 앞서 지적했듯이, 현업 사용자와 분석가 사이에서 비 효율적 의사소통 반복을 수반한다는 한계를 지니고 있다. 이 기법은 DFD와 함께 가장 폭넓게 이용되어 온 데이터 모델링 기법으로, 시스템에 저장된 데이터들간의 네트워크라고 정의할 수 있을 만큼 개체간의 관계에 대해서 높은 표현력을 가지고 있으며, 시스템 안에 저장된 데이터를 모델링하고 연관된 개체들끼리 서로 독립적으로 운영할 수 있게 하는 관계를

설정해주는 데에 초점을 두고 있다(Chen 1976). 그러나 비즈니스 프로세스를 묘사하는 데 있어서, 개체관계도는 자료흐름도와 마찬가지로 불분명한 요구사항이 제시되거나 추가적 요구사항이 발생했을 때 분석과 검증 작업을 반복해야 하기 때문에, 프로젝트 완성까지 추정된 비용과 기간을 초과하게 될 가능성이 높다고 볼 수 있다.

2.6 유즈케이스 다이어그램 (Use Case Diagram: UCD)

객체지향 분석의 대표 기법인 UML 중 하나인 유즈케이스 다이어그램 기법은 사용자의 시각에서 목표 시스템을 바라볼 수 있게 해준다는 장점이 있긴 하지만, 이 방법을 현실에서 쉽게 적용하기에 몇 가지 어려움이 있다. 우선 분석을 통해 도출된 유즈케이스 다이어그램은 도출된 이후에도 관계형 데이터베이스 시스템으로 전환하는 작업을 추가적으로 거쳐야 한다. 또한 행위 중심으로 표현된 추상적 개념들을 철저히 사용자 지향적으로 구성하였기 때문에, 분석상의 다양성을 보여주고 복잡한 시스템을 단순화시켜 묘사할 수 있지만, 현존하는 대부분의 관계형 데이터베이스 시스템으로 데이터 구조를 전환시키기 위해서는 또 다시 개체관계도와 같은 기법을 별도로 사용해야 한다는 부담이 있다.

2.7 방법간의 특성 비교

정확하고 신속하게 업무 프로세스를 기술하려면, 현업 실무자들이 묘사 기법들을 쉽게 접근하고 이해할 수 있어야 한다. 우리는 사전 지식이 전무하거나 거의 없는 현업 실무자들이 쉽게 접근 여부를 나타낼 수 있는 ‘활용 용이성’이라는 지표를 도입한다. 만약 표현 요소의 개수가 적고 간단하다면, 현업의 실무자가 배우고 사용하기가 쉽고 도출된 결과물을 직관적으로 이해하는데 부담을 덜 느끼게 된다. 반대로 사용하는 기호가 많을수록 분석업무에 정통하지 못한 사용자들은 분석-설계 업무를 위한 분석가와의 지식 공유행위가 부담스럽게 느껴지며, 정확한 요구사항이 기술되었는지를 확인하기가 힘들게 느끼게 된다.

이들 6개의 기법들은 기업 업무를 묘사하는데 있어서 데이터 중심적 관점, 프로세스 중심적 관점, 그리고 추가적으로 분석 단위의 데이터와 프로세스가 융합되어 나타나는 역할(Role) 관점에 의해서 분류될 수 있다. 역할 관점은 업무에 연관된 분석 단위가 전체 프로세스 내에서 어떤 부분을 차지하는지 하향적 시각으로 접근하기 때문에, 종합적인 틀을 제시하는데 바람직하다. 그러나 업무가 다양해질수록 실제 업무 흐름을 반영하기에는 미흡한 측면이 있고 복잡성이 증가하기 때문에, 역할 관점 분석은 업무 프로세스를

<표 3> 프로세스 묘사기법 비교 결과

묘사 기법	표현 요소 개수	묘사관점	방법론적 온전성	적용편이성
IDEF0	2	프로세스	Low	Low
Role Activity Diagram	6	역할	Mid	Low
System Dynamics	4	프로세스	Mid	Mid
개체관계도	3	데이터	High	High
자료흐름도	4	프로세스	High	Mid
유즈케이스 다이어그램	4	프로세스	Mid	High

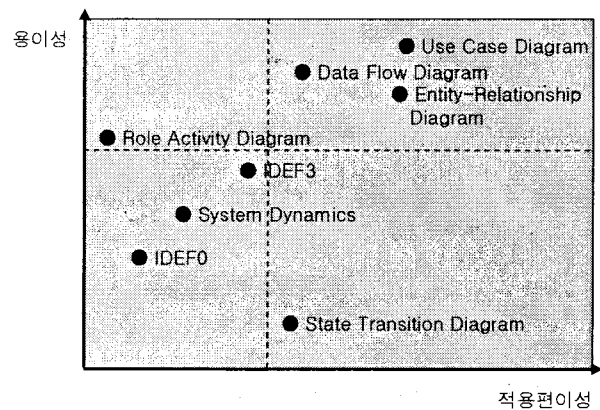
세세히 묘사하기에는 상대적으로 부적당하다는 사실이 노정된다. 예를 들어, 개체관계도와 같은 데이터 중심 기법은 실제 업무에 관계된 데이터 명세를 1:1로 파악할 수 있다는 장점에도 불구하고, 프로세스 묘사에 대한 특별한 가이드 라인이 없이는 자칫 행위들의 경우의 수를 모두 표현하기 어려운 경우가 생길 수 있다. 또한 자료흐름도로 대표되는 프로세스 중심 관점은 참여자들이 어떤 흐름으로 업무에 참여하는지를 일목요연하고 세세하게 보여줄 수 있다는 장점이 있긴 하지만, 이를 기준으로 필요한 데이터를 도출해야 한다는 추가적 부담이 있다.

고려하지 않았기 때문에 업무 프로세스를 그대로 기술할 경우 프로그래머와 분석가들로 하여금 추가적인 변환 작업이 따라야 했다. 예를 들어, 개체관계도로 표현된 결과물의 경우 별다른 수정 없이 그대로 개체와 관계를 프로그래밍의 테이블로 전환이 가능하기에, 그리고 유즈케이스 다이어그램은 객체지향 프로그래밍 개념과 일치하기 때문에, 유연성 있게 구현작업으로 전환시킬 수 있기에 적용편이성이 뛰어나다고 할 수 있다. 이제까지 검토해 본 프로세스 묘사 기법들의 특성들은 <표 3>과 같이 요약될 수 있다.

묘사 기법의 방법론적 ‘온전성’은 업무를 표현하는 기법 한 가지만으로 기업의 데이터와 프로세스, 그리고 업무의 분기나 시작과 종료를 얼마나 명확히 표현할 수 있는지를 나타내는 지표이다. 온전성이 높은 방법론일수록 간단명료하게 정확한 묘사를 할 수 있다. 예를 들어, 온전성이 떨어지는 IDEF0 기법의 경우 별도의 보완을 위한 다이어그램을 필요로 하며, Role Activity Diagram는 역할 위주 업무를 표현하고 있기 때문에 상세한 업무 분기나 의사결정 과정 등은 표현해 내지 못하고 이에 따라 명세서를 별도로 첨가해야 한다.

한편 <표 3>에 나온 6개 방법론이 용이성 및 적용편이성의 2차원 측면에서 각각 어디에 위치하는지를 <그림 1>이 보여준다. 다음 장에서는 용이성과 적용편이성에서 다른 기법들보다 우위에 있는 것으로 나타난 자료흐름도, 개체관계도, 그리고 유즈케이스 다이어그램을 표준업무를 놓고 비교해 본다.

마지막으로 프로세스 묘사 과정을 통해 도출된 결과물을 ‘적용편이성’ 척도로 비교해 볼 수 있다. 적용편이성이란 최종 도출결과를 통해 상용 데이터베이스 시스템으로 전환시키거나 실제 프로그래밍 언어를 통해서 시스템 구현시킬 때 얼마나 적은 노력이 들어가는지를 나타내는 지표이다. 현재 지배적으로 활용되고 있는 관계형 데이터베이스 시스템이나 객체지향 프로그래밍 시스템이 출현하기 전에도 여러 가지 프로세스 묘사 기법이 나타났는데, 이 기법들은 프로그래밍 환경이나 데이터베이스 시스템 구조를 전혀



<그림 1> 용이성과 적용 편이성으로 비교한 프로세스 묘사 방법론의 특성

3. 표준 업무 기반 적용 사례

효과적인 업무 시스템분석을 위해서, 분석가의 입장에서는 업무 프로세스의 요구사항을 누락시키지 않기 위해 보다 자세한 묘사가 필요하며, 사용자의 입장에서는 묘사가 간단명료해야 결과물을 쉽게

이해할 수 있고 결과물이 실제의 업무 프로세스를 반영하는지를 확인할 수 있다. 만약 이 두 목표가 적절히 배합되지 못한다면, 사용자와 분석가 모두는 반복적인 작업을 비효율적으로 수행해야 한다. 따라서 정보시스템 분석단계에서 필요한 사용자 요구형성 과정에서 기술된 내용이 사용자가 이해하기 쉬울수록 좋으며, 사용자의 이해 가능성을

증대시키기 위해서는 되도록 간단한 프로세스 묘사가 선행되어야 한다. 그리고 기술된 내용의 정확도를 높이기 위해서, 구현하고자 하는 프로세스에 대한 정보누락이 없어야 하며 중복되는 정보가 최소한이어야 한다. 나아가 업무 표현에서 추가·변경되는 내용이 있을 경우에 이에 따라 추가적으로 발생하는 업무 부담이 적을수록 분석가에게는 좋다.

이번 장에서는 표준업무로 A은행의 자산 건전성 평가업무를 중심으로 묘사 상의 정확도와 명료성에 관하여 각 방법론들이 가지는 차이점들을 원활하게 비교해 본다. 은행의 자산 건전성 평가업무는 대부분의 금융기관에서 이뤄지는 여신 심사, 분류 회수에 관련된 업무이며, 인터뷰, 설문조사, 방문조사 등의 방법을 통해 파악된 업무는 정해진 양식에 맞추어 대개 직무단위 별로 기술된다. 금융기관의 업무는 부서단위로 계층적 특성을 갖는 경우가 많고, 자산 건전성 평가업무는 대부분의 금융기관에서 이행되는 표준적인 단위 업무이며, 따라서 다양한 분기 별 데이터가 존재하고 평가업무가 충분히 복잡하기 때문에, 논문이 검토하는 기법들을 비교하기에 좋은 테스트베드가 된다. 또한 기업 업무를 구성하는 여러 형태의 행위가 이미 명시적으로 주어져 있기 때문에, 업무 프로세스를 기술하는 데 있어서 주관성이 개입할 여지가 상대적으로 적다고 볼 수 있기 때문에, 표준적인 업무로 보아도 무방하다.

3.1 연구를 위한 가정

업무 프로세스 묘사에 대한 객관적이고 명확한 비교를 위하여 A은행의 표준업무를 선정하였지만, 명확한 분석 결과 도출 및 비교를 분명하게 만드는 몇 가지 가정들을 도입한다. 동일한 업무에 대해 비교 가능한 척도를 적용하기 위한 가정들은 아래와 같다.

가정1 (설계될 요구사항): 업무 프로세스 표현 기법을 통해 도출된 결과물로 형성된 요구 자료의 정확성과 완전성에는 차이가 있을 수 밖에 없으나, 객관적으로 비교 가능한 형태로 프로그램으로 변환하는데 사용하는 요구사항은 동일하다고 가정한다.

가정2 (분석 능력의 동일성): 도출되는 모든 결과물은 동일한 표현능력을 가진 현업의 실무자와 분석가에 의해 이뤄진다.

가정3 (묘사 범위): 표현 범위를 명확히 하기 위해 업무 프로세스 묘사는 자산 건전성 단위업무에 직접적으로 연관된 업무로 제한하였고 다른 단위 업무(회계, 인사, 재무 등)와 관련 부분은 생략한다.

가정4 (개체관계도의 의 표현): 개체관계도를 작성하여 표현할 때는 Optionality와 Cardinality를

나타내기 위한 Crow's foot를 활용하지만, 공간의 부족으로 인한 표현상의 어려움을 줄이고 비교상의 편의를 도모하기 위해 모두 (Chen 1976)의 방식을 사용하여 나타낸다. 그리고 단순한 표기를 위해 도출된 개체의 Non-Key 속성도 별도로 표현하지 않는다.

가정5 (업무기술서): 업무기술서는 문장형태의 표현도구로 현업에서 나타나는 프로세스와 데이터를 빠짐없이 표현할 수 있는 방법이다.

가정6 (유즈케이스 다이어그램의 표현): 액터와 유즈케이스 간의 관계는 표현의 단순화를 위하여 실선으로 표현하는 관계만을 사용하며 확장관계나 연관관계는 별도로 나타내지 않는다.

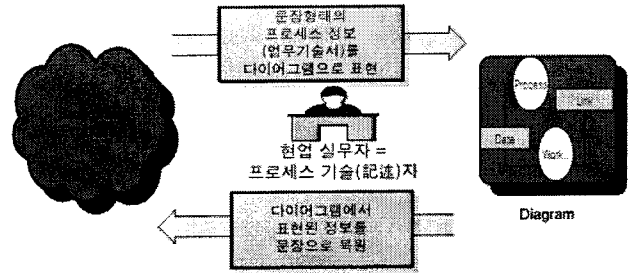
3.2 평가 기준 및 지표

사용자 요구사항을 정리한 결과물이 다양한 형태의 업무형태를 자세히 기록하지 못하다면 표현력이 약한 방법론으로 평가 받기 쉽다. 그러므로 각 기법에 의해 도출된 결과물이 프로세스 관점에서 얼마나 다양한 형태의 업무 형태를 표현해 낼 수 있는지에 대한 정도는 중요한 비교 기준이 된다. 즉, 데이터 관점에서 기업 업무를 묘사함에 있어서 입출력, 생성 및 참조가 잘 나타나고 있는지를 살펴 볼 필요가 있으며, 동적인 업무 프로세스에 대한 표현력의 관점에서 프로세스의 분기와 다양성뿐만 아니라 활용되는 데이터를 통해서도 나타나는 복잡성을 제대로 표현할 수 있어야 한다. 아울러 프로세스의 양이 절대적으로 많더라도 프로세스에 대한 정보가 부족하게 기술된다면, 업무의 복잡성을 묘사하는데 충분하지 못하며, 표현되는 데이터를 양적 측면 위주로 기록하면 업무의 동적인 측면을 소홀히 한 채 정적인 측면에 집중하게 된다는 문제점도 항상 고려되어야 한다.

자료흐름도에서는 데이터 입출력에 관계된 화살표의 개수가 각 데이터에 관련된 업무행위와 연관이 되어있으며, 개체관계도에서는 개체들 사이에 맺어지는 관계 화살표가 기업 업무 행위로, 유즈케이스 다이어그램의 경우는 타원형의 모양으로 표현되는 각각의 유즈케이스가 업무 참여자가 보이는 행위라고 할 수 있다. 도출된 결과물을 놓고 업무에 대한 묘사가 정확한지를 보기 위해서, 표현된 데이터와 프로세스가 어느 정도까지 상세히 기술되었는지를 조사하고, 단순 수치 비교가 아닌 정성적 비교가 가능하도록 단위 행위마다 업무 파악이 심층적으로 진행되었는지를 조사할 수 있다. 우리는 먼저 표준 업무로 선정된 A은행의 자산 건전성 평가업무에 대해 상세히 기술한 다음, 계속해서 상대적으로 우수한 것으로 분류된 자료흐름도, 개체관계도, 유즈케이스 다이어그램의 세 기법들이 표준업무를 놓고 도출하는 결과물을 놓고 비교·분석한다.

3.3 업무 기술서

현업 실무자에게 명시되지 않고 암묵적으로 존재하는 단위 업무의 절차와 방식을 문장 형태로 상세히 서술하는 과정이 없다면 프로세스 묘사 결과물에 대한 정확성 평가는 불가능하다. 업무기술서는 이러한 필요성을 만족시켜주는 문장 형태의 업무 처리 규정이다. 업무기술서는 업무 분석 이전에 요구형성 과정이 더 중요하다고 보는 요구 형성 위주의 방법론으로, 어떤 의뢰자의 요구라도 '업무 처리 규정' 형태로 나타낼 수 있어야 하고, 불필요하거나 중복되는 요구들을 최소화 하며, 일관성이 없이 표현되거나 상충되는 요구들이 있다면 이들을 일치시키는 작업을 수행하여, 일관된 업무처리규정이 나오도록 돕는다. 데이터간의 관계를 가능하면 선명하게 업무 처리 규정을 작성하면 결과물에서 데이터의 연관성을 유지할 수 있기 때문에, 데이터간의 관계 표시 여부는 업무 프로세스 묘사 정확도를 가능하는 가장 중요한 자료가 된다. 지면 상의 이유로 표준업무에서 도출된 업무기술서는 이 논문에서 생략되었지만, <그림 2>는 업무 기술서가 하는 이러한 역할을 도식으로 보여준다.



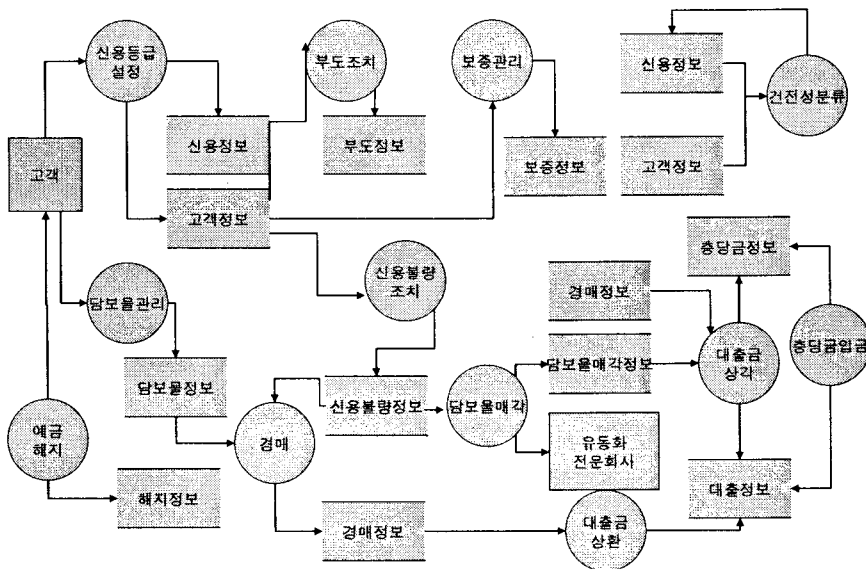
<그림 2> 문장형태로 서술한 업무기술서의 역할

개체와 개체 사이의 관계로 보며, 이를 업무 기술서의 각 규정 항목과 1:1 상응해서 나타나도록 개체들과 개체 사이의 관계를 도식화하면, <그림 3>과 같은 개체관계도가 완성된다. 마찬가지로 데이터의 흐름을 데이터의 흐름을 위주로 구성된 자료흐름도, 그리고 각 액터의 행위 중심으로 재구성한 유즈케이스 다이어그램을 업무기술서에 의해서 만들 수 있다. 이제 동일한 업무기술서에 의해서 얻어진 세 개의 기법을 놓고 정량적인 비교가 가능하다.

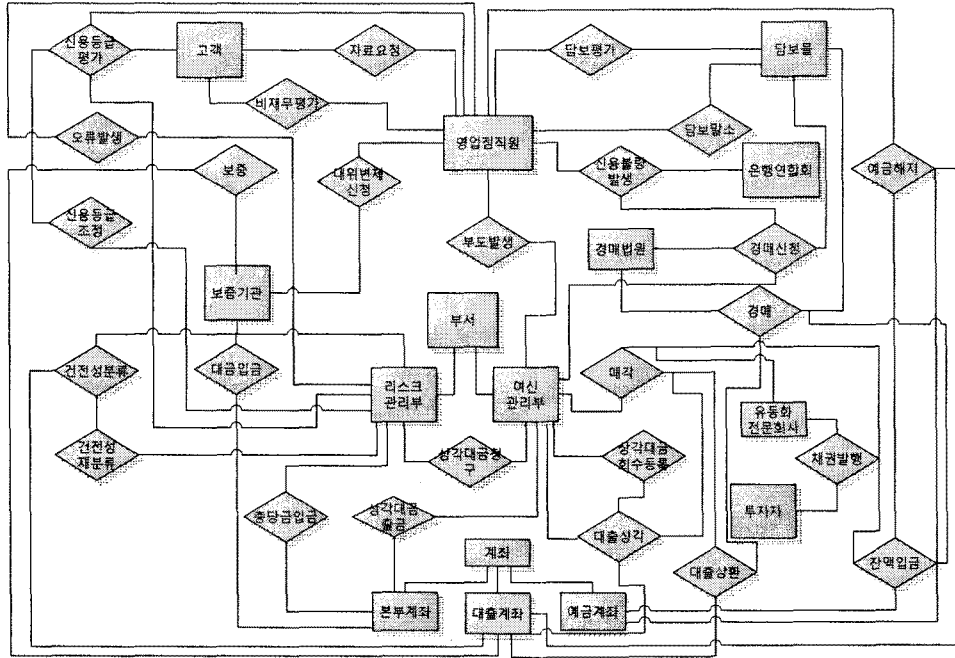
하지만, 명확한 작성 규칙 없이 작성된 업무기술서는 자칫하면 업무를 중복 표현할 수 있기 때문에, 중복 표현을 최소화 하기 위해 문송천(2004)이 제안한 NS 방법론과 같이 <주어+동사+목적어> 형태의 단문형식으로 기술되어야 한다. 업무 분석이 파악된 다음 업무기술서를 바탕으로, 개체와 관계를 식별하기 위해서 여기서는 '어떤 주체는 어떤 개체에 대해 어떤 행위를 한다' 라는 3형식 문장을 기준으로 '어떤 개체에 대해 해당되는 명사부분을 주가 되는 개체로 보고, '어떤 행위를 한다' 라는 동사부분을

3.4 자료흐름도로 표현한 자산 건전성 평가업무

자료흐름도는, 모든 기업 업무를 데이터의 생성, 삭제, 입출력, 참조와 같은 행위와 관련되어 있다는 시각으로 접근해서, 프로그램이나 시스템에서의 데이터의 논리적 흐름을 도형으로 표현하는 업무 표현 도구라고 할 수 있다. 시스템 안과 밖을 구분하기 위해서 소스(source)와 싱크(sink)를 직사각형이나 정사각형으로 표시하며, 데이터의 흐름을 보여주기 위해서 입력 데이터에 가해지는 변형과정을 원이나 둥근 직사각형으로 표시하고, 데이터의 흐름을 화살표로 나타냄으로 일목요연한 결과물을 보일 수 있다. <그림 3>은 A은행의 건전성 평가업무를 표현한 자료흐름도이다. <그림 3>의



<그림 3> 자료흐름도로 도출한 표준 업무



<그림 4> 개체관계도로 작성한 표준 업무

자료흐름도에서는 모두 12개의 데이터와 관련된 행위가 묘사되었고 자료의 참조나 원천이 되는 개체는 16개가 발견되었다. 또한 데이터의 생성, 참조, 이동 등과 관련되어 있는 링크는 모두 28개로 표현되었다. 특히 자료흐름도에서는 50%가 넘는 표현요소가 데이터의 흐름을 서술하는데 활용되고 있는 것을 알 수 있다. 또 각 개체들이 링크를 통해 모두 연결되어서 독립적으로 수행되고 있는 업무는 발견되지 않았고 단위 업무들 간에 유기적인 표현이 가능하다는 사실을 확인할 수 있다.

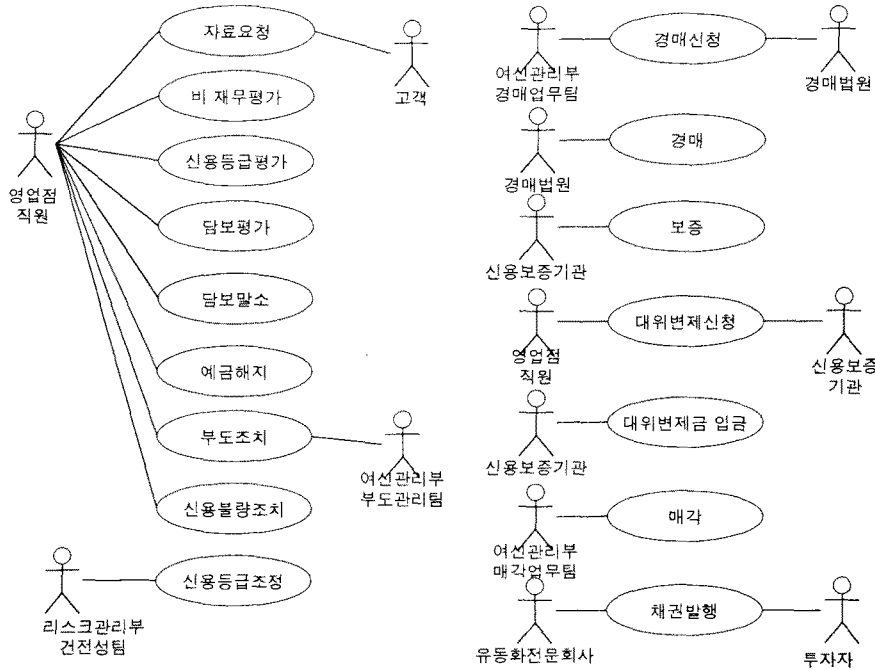
3.5 개체관계도로 표현한 자산 건전성 평가업무

개체관계도를 작성하기 위해서, 논리적 데이터 모델링 단계에서는 주요 개체를 기준으로 개체를 정의하며 개체들 간의 관계를 정의한 개념적 논리모델을 다이어그램으로 그린다. 나타난 개체를 대상으로 속성 후보를 선정하고, 구체적인 검증과정을 거치면서 속성이 결정되고 경우에 따라 보다 상세하게 표시된 개체가 추가되기도 한다.

이어서 이 단계까지 완료된 주요 개체들에 대한 식별자를 확정하며, 데이터 모델의 골격이 되는 개체들을 사례 데이터와 비교·검증하면 <그림 4>와 같은 기본 논리적 데이터 모델이 완성된다.

개체관계도에서 15개의 (직사각형 모양의) 개체가 산출되었으며, 프로세스로 볼 수 있는 (마름모로 표현된) 관계명이 21개, 개체 및 관계 양자 모두로 간주할 수 있는 경우가 5개, 그리고 개체간의 관계에서 화살표로 표현되는 31개의 행위가 도출되었다. 한편 각 표현 요소가 개체와 행위, 그리고 관계를 묘사하는 연결선에서 활용되고 있다. 개체관계도는 자료흐름도와 달리, 개체간의 관계를 모두 행위로 묘사하고 있어서, 정적인 측면의 업무 묘사임에도 불구하고 3개의 도형만으로도 표준 업무를 충분히 표현할 수 있다는 것을 알 수 있다.

3.6 유즈케이스 다이어그램으로 표현한 자산 건전성 업무



<그림 5> 유즈케이스 다이어그램으로 표현한 표준 업무

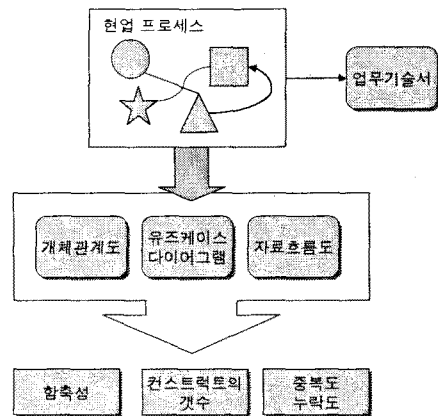
프로젝트의 기능적인 면을 위주로 사용자와의 명확한 의사소통을 목적으로 고안된 유즈케이스 다이어그램은, 행위자(actor)가 관심을 가진 작업을 달성하는 과정에서 발생하는 (행위자, 시간의 흐름, 혹은 다른 시스템이 시작하는) 시나리오의 집합이다. 행위자들은 도출된 결과물을 직관적으로 이해할 수 있지만, 결과물 구축에 대한 어떤 정보도 나타나 있지 않다. <그림 5>은 자산 건전성 업무에 대한 유즈케이스 다이어그램이다.

표준업무에 기반한 도출된 유즈케이스 다이어그램을 살펴보면 행위에 참여하고 있는 액터가 14개, 타원 형태로 나타난 유즈케이스가 16개, 그리고 액터와 유즈케이스를 이어주는 관계가 21개로 나타난다. 행위 중심의 묘사가 두드러지는 유즈케이스 다이어그램 문장으로 서술했을 때, 개체의 행위나 데이터의 흐름을 묘사하는 액터와 유즈케이스의 합계가 75%에 가깝게 활용되고 있는 것으로 보아 유즈케이스 다이어그램이 동적 행위를 묘사한다는 특성이 그대로 반영되어있음을 알 수 있다. 또한, 표준 업무로 선정한 자산 건전성 업무는 단일 시스템으로 가정했기 때문에, 시스템 영역을 구분하는 직사각형 도형이 사용되지 않았고 업무의 방향이 잘 구분이 되어있으며, 액터와 유즈케이스 사이의 관계가 단편적이어서 비교적 잘 구분된 형태로 유즈케이스들이 도출되었음을 알 수 있다.

4. 비교 분석

4.1 분석 모형

지금까지 우리는 자료흐름도, 개체관계도 그리고 유즈케이스 다이어그램을 대상을 기준으로 도출된 결과물을 놓고 파악할 수 있는 각 요소의 개수와 요소간의 비율을 비교해 보았다. 이 장에서는 현업에 대한 지식을 도식화된 산출물로 표현할 경우에 나타나는 두 가지 과정—즉 문장이나 단어에 포함된 의미가 도형으로 나타날 때 발생하는 은유나 함축과 같은 추상화 과정과 사용자의 요구사항을 구체화하는 과정—을 정량화하기 위해서, 표현되는 모든 도형 요소를 현실 세계의 최소의미 단위인 컨스트럭트(construct) 개념으로 해석해 본다(Weber 2003). 비즈니스 프로세스에서는 한 가지 특정 개념을 설명하는 단어가 업무기술서의 문장 하나의



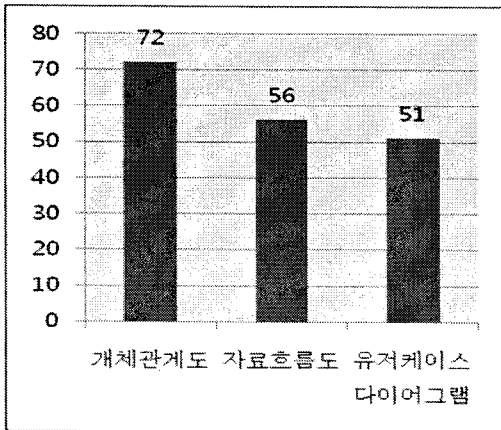
<그림 6> 기법 평가를 위한 분석 모형

주어나 동사로 서술되기 때문에, 각 의미 단위를 1개의 컨스트럭트로 간주하여 정량화 시킬 수 있다. 예를 들어, '판매부서는 실적현황을 입력한다'라는 문장은 주어인 '판매부서', 목적어인 '실적 현황', 그리고 동사인 '입력한다'라는 행위 등 세 개의 컨스트럭트로 나누어 계수할 수 있다. <그림 6>은 이번 장에서의 연구 분석모형을 도식화해서 보여준다.

우리는 세 가지 기법에서 각각 표현하고 있는 산출물의 함축성, 컨스트럭트의 개수, 그리고 세부 행위 별로 어느 정도의 중복과 누락이 발생하는지를 심층적으로 분석했고, 이를 통해 각 기법이 가진 우수성 및 관점상의 특징을 실증적으로 발견해 낼 수 있었다.

4.2 함축성 비교

표현상의 함축성은 도출된 결과물이 얼마나 간단하게 표시되었는지를 알려주는 지표이다. 묘사의 결과물이 최대한 적은 수의 컨스트럭트 개수로 나타나는 것으로 얼마나 표현이 함축되었는지를 평가할 수 있는데, 표준업무에 적용한 결과는 다음 그림과 같다.



<그림7> 기법 별로 표현된 컨스트럭트의 개수

도출 결과물의 단순성은 유즈케이스 다이어그램이 51개로 나타나 가장 함축성이 가장 높은 것으로, 이보다 21개가 더 많은 요소로써 표현된 개체관계도보다 40% 넘는 함축성을 보여주었고, 자료흐름도도 이보다 유사한 수준의 함축성을 보였다. 유즈케이스 다이어그램의 경우, 외부에서 시스템을 바라보고 업무행위를 묘사하기 때문에, 그리고 내부 수행 과정은 블랙박스 외부에 나타나지 않아야 하기 때문에 높은 함축성을 보인다.

4.3 누락도

업무기술서를 기초로 도식화된 결과물이 업무에

대한 정보를 누락하고 있는지의 여부를 알기 위해, 업무 전문가를 통해 파악한 24개의 업무 단위를 나열하고 각 기법이 어떠한 형태로든지 정확하게 표현되어있는지를 파악해 보았다. 심층적인 분석을 위해서, 소단위 업무는 테이블의 세로에, 가로에는 각 기법명을 나열하여 해당 업무가 빠짐없이 묘사되고 있는지에 대한 정도를 점수화하여(0: 묘사하지 않음, 1: 부분적으로 묘사함, 2: 온전히 묘사함) 나타낸 결과, 각 기법의 누락도 현황은 아래 표와 같다.

<표 4> 기법 별 누락도의 측정

기법명	자료 흐름도	개체 관계도	유즈케이스 다이어그램
단위업무 소개	27	43	25
누락도 = {(전체 단위업무 × 2) - 누락점수 누계} / (전체 단위 업무 수 × 2) * 100	43.8%	10.4%	47.9%

<표 4>에 의하면 자료흐름도와 유즈케이스 다이어그램의 누락도가 각각 43.8%와 47.9%인데 비해서 개체관계도의 업무누락도가 10.4%로 나타나, 개체관계도가 상대적으로 프로세스를 묘사하는데 있어서 더 우수하다는 것을 보여준다. 누락도에서의 큰 차이는 컨스트럭트의 개수로 비교할 수도 있지만, 단위업무 각각을 깊이 있게 파악하는 작업이 더 중요하다는 사실을 나타낸다. 단순 수량비교 측면에서 높은 함축성을 보였던 자료흐름도가 실제로는 잔액 입금과 같은 소단위 업무를 전혀 표현하지 못하고 있었으며, 유즈케이스 다이어그램에서도 동일한 현상이 발견되었다. 이와 같은 누락도에서의 두드러진 차이는 놓고, 요구형성 단계에서 얻어진 업무기술서를 기초로 해서 개체관계도가 작성 될 때, 요구분석 단계에서 도출된 개체와 개체 사이의 관계를 추가하는 과정이 수반되면서 업무기술서의 항목들이 개체와 개체 사이의 관계 및 행위로 자연스럽게 기록되었기 때문이라고 설명할 수 있다.

4.4 중복도

도출된 결과물을 놓고 정보의 중복이 있을 경우에도 반복적인 작업이 요구되기 때문에, 중복된 표현을 나타내는 척도, 즉 중복도에 대한 파악은

<표 5> 기법 별로 중복된 표현 요소들

구분	자료흐름도	개체관계도	유즈케이스 다이어그램
2회 이상 표현된 컨스트럭트	경매정보 고객정보	계좌 부서	영업점직원 신용보증기관

프로세스 묘사 기법의 우월성을 검증할 때 중요한 지표가 된다. A은행의 자산 건전성 평가업무에 대한 각 기법의 중복도를 조사한 결과는 아래 표와 같다.

자료흐름도의 경우, 데이터의 입출력과 관련된 흐름을 표현하는 가운데 발생하는 순환구조(circular structure)를 피하기 위해 경매정보와 고객정보가 중복되게 표현하는 것이 불가피하다. 따라서 순환구조가 자연스럽게 발생하는 자료흐름도의 경우, 반복적 의견수렴 및 정보교환이 발생하게 될 것이라는 것을 어느 정도 예상할 수 있다. 한편 개체관계도에서는 계좌(본부계좌, 대출계좌, 예금계좌)와 부서(리스크관리부, 여신관리부)가 중복 표현되는데, 중복된 계좌 혹은 부서들은 상이한 속성을 가지는 성격의 컨스트러트들이지만, 개체관계도에서 각 계좌와 부서는 별도의 프로세스를 갖는 개체로 표현되기에 중복된 표현이 발생하게 된다. 마지막으로 유즈케이스 다이어그램의 경우, 행위중심으로 표현되는 까닭에, 개체들이 중복되어 표현되는 것을 피할 수 없다. 예를 들어, <그림 5>에 나와 있는 '영업점 직원'의 경우 8개의 유즈케이스를 가지기 때문에, 각각의 유즈케이스 별로 따로따로 표현된다. 전체 의미 단위에서 발생하는 중복도를 상세히 나열하면 <표 6>과 같은 결과를 얻을 수 있다.

<표 6> 기법 별 중복도

구분	자료흐름도	개체관계도	유즈케이스 다이어그램
컨스트러트의 개수(A)	56	72	51
동일 의미단위 수 (B)	3	2	2
유사 의미단위 수 (C)	2	2	2
중복도 [(B)X2+(C)]/(A)	14.3%	8.3%	11.8%

우리는 <표 6>에서, 만약 기법 별로 도출된 결과물에서 완전히 동일한 의미의 정보가 중첩되었다면 2배의 가중치를 주었고, 유사한 의미단위로 나타났을 때는 가중치를 주지 않았으며, 도출된 중복 의미단위 개수 대비 전체 의미단위 개수와의 비율을 점수화 하여 중복도를 산출하였다. 상대적으로 많은 의미단위를 사용하고 있는 개체관계도에서는 예상과 달리 8.3%의 중복도를 나타냄으로, 14.3%와 11.8%의 중복도를 보이는 자료흐름도와 유즈케이스 다이어그램과 근소한 차이를 보이고 있다. 이런 근소한 차이가 어떤 기법이 다른 기법보다 더 우월하다는 것을 보일지, 그리고 다른 업무 프로세스의 예에서도 비슷한 결론이 나올지는 추후 다른 표준업무를 통해 살펴볼 수 있을 것이다.

5. 결론

이 연구에서는 기업의 업무 프로세스를 묘사하기 위해서 주로 사용되던 기법을, A 은행의 자산건전성 평가업무를 통해서, 누락도, 중복도, 명확도(또는 함축성)의 차원에서 비교·분석해 보았다. 기업 업무에 참여하고 있는 행위자와 이를 통해 파생되는 데이터가 누락이나 중복 없이 나타났는지, 또한 업무의 복잡도가 높을지라도 단순하고 일목요연한 묘사가 가능한지를 파악하기 위해서 각 방법론에 따라 요구사항을 분석하고 이를 통해 산출되는 결과물의 특성을 분해하여 정량적으로 비교해 보았다. 또한 단순한 수치적 비교의 한계점을 극복하기 위해 각 업무별로 특성 있는 부분을 나누어 누락과 중복의 정도 그리고 표현력을 별도로 대조하여 분석하였다.

다이어그램이나 플로우차트와 같이 단순한 표현 도구는, 데이터 중심적 관점, 프로세스 중심적 관점, 역할 중심적 관점 등의 다양한 시각에서 사용자와 분석가 모두가 이해하기 쉬운 도형들을 활용하여 프로세스를 가장 직관적으로 묘사하는 도구로 사용되어 왔다. 그러나 정보시스템의 발전과 함께 현실세계의 업무가 더욱더 복잡해지면서, 명확한 기준과 관점이 부재한 상태에서는 특정 업무 프로세스에 대한 묘사가 누락된 중복된 혹은 왜곡될 가능성이 높아진다. 따라서 이 연구에서는 프로세스 중심 기법인 자료흐름도와 유즈케이스 다이어그램과 비교할 때, 데이터 중심 기법인 개체관계도가 상대적으로 낮은 함축성과 높은 복잡성에도 불구하고 낮은 누락도와 명확한 표현력을 가지고 있다는 개연성을 금융기관의 표준업무를 통해서 살펴 보았다. 4장에서 나온 결과를 놓고 볼 때, 개체관계도의 함축성이 낮은 까닭에 복잡한 묘사가 수반되어야 할 때가 발생하는데, 이를 보완하기 위해 업무기술서를 병행하면 데이터 중심에서 행위 중심으로 프로세스를 묘사할 수 있고, 이를 기업의 부서나 개인단위의 구조화된 업무에 적용하여 프로세스 함축과 누락에 대한 부담을 크게 감소시킬 수 있을 것으로 예상된다. 또한 실제 기업의 업무사례를 각 기법에 적용하여 현업에서 발견될 수 있는 부작용을 기준지표에 따라 절대적 혹은 상대적인 점수로 산출했는데, 개체관계도의 경우 데이터 중복도를 최소화 하려고 노력할 때 시스템 성능 향상될 것으로 예상되기에 상대적으로 효용성이 높다고 할 수 있다. 물론 관계형 데이터베이스 시스템으로 전환하는데 별도의 작업이 필요하지 않기 때문에, 개체관계도의 효용성이 다른 방법론보다 더 높다고 할 수 있다. 그러나 본 연구는 요구 형성 혹은 프로세스 묘사 과정에서 사용자나 분석가의 특성이 개입될 수 있다는 한계점을 갖는다. 즉, 데이터 모델링이나 프로그래밍과 같이 시스템 구축에 관계된 업무

경험이 있는 분석가의 경우, 분석에 관한 숙련도가 높기 때문에 산출 결과물에 대한 청사진을 그리기가 쉽다는 이유로, 특정 방법론에 치우친 분석으로 기울어질 가능성이 높다. 또한 분석가나 사용자의 주관성에 따라, 도출된 결과물이 다른 형태로 나타날 가능성도 항시 존재한다. 그리고 각 기법들이 활용하고 있는 도식들을 비교함에 있어서, 시각적으로 단순할수록 우수하다는 인상을 주는 경향이 있는데, 단순화 정도에 대한 수치 비교는 구현될 시스템에 대한 묘사력을 저하시킬 수 있다는 제한점이 있다고 하겠다.

마지막으로 여러 영역의 업무 프로세스에 대해서, 본 논문이 제시하는 지표들이 이 연구에서 나온 결과와 유사한 결과가 나올지, 아니면 다른 결과가 나올 때 어떤 방법론이 문제점을 보충해 줄 수 있을지는 향후의 연구과제가 될 것 같다. 또한 분석업무에 대해 사전지식이 무관한 실무자들에게 가장 친숙한 도구일 수 있는 Class Diagram 이외에도 Static Structure Diagram 이나 Interaction Diagram 과 같이 다른 보완적 기법으로 확대하여 실험비교를 한다면 보다 실용적인 기여를 할 수 있을 것으로 기대할 수 있다.

참고문헌

- [1] 문송천, *데이터 아키텍처* (2004), 형설출판사.
- [2] 유치수, 김갑수, 이명재 (1998), *소프트웨어 공학*, 홍릉과학출판사.
- [3] Benediktsson, O, D. Dalcher, K. Reed and M, Woodman(2003), "COCOMO-Based Effort Estimation for Iterative and Incremental Software Development," *Software Quality Control*, vol.11(4), pp.265-281.
- [4] Berente, N. and K. Lyytinen (2005), "Iteration in Systems Analysis and Design: Cognitive Processes and Representational Artifacts," Case Western Reserve University, USA. *Sprouts: Working Papers on Information Systems*, 5(23).
- [5] Booch, G. (1986), "Object-Oriented Development," *IEEE Transaction on Software Engineering*, vol.12.
- [6] Brinkkemper, Sjaak (1996), "Method engineering: engineering of information systems development methods and tools," *Journal of Information and Software Technology*, vol.38, pp.275-280.
- [7] Chen, Pin-Shan Peter (1976), "The entity-relationship model: toward a unified view of data," *ACM Transactions on Database Systems*, vol.1, No.1, pp.9-36.
- [8] Cherf, G. S. (1992), "An investigation of the maintenance and support characteristics of commercial software," *Software Quality Journal*, vol.1, pp.147-158.
- [9] Hull, Elizabeth, Ken Jackson and Jeremy Dick (2002), *Requirements Engineering*, Springer.
- [10] Emam, K. E. and Madhavji, N. H. (1995), "A field study of requirements engineering practices in information systems development," *Proceedings of the 2nd IEEE International Symposium on Requirements Engineering*. IEEE Computer, pp.68-80.
- [11] Forrester, J. W. (1961), *Industrial Dynamics*, MIT Press, Cambridge, MA.
- [12] Giaglis, G. M. (2001), "A Taxonomy of Business Process Modeling and Information Systems Modeling Techniques," *International Journal of Flexible Manufacturing Systems*, vol.13, pp.209-228.
- [13] Glass, L. Robert (1997), *Software Runaways: Monumental Software Disasters*, Prentice-Hall, New York.
- [14] Godwin, A. N. J. W. Gleeson and D. Gwillian (1989), "An assessment of the IDEF notations as descriptive tools," *Information Systems*, vol.14, no.1.
- [15] Huckvale, T. and M. Ould (1995), "Process Modeling: Who, What and How: Role Activity Diagramming," in *Business Process Change: Concepts, Methods, and Technologies*, Idea Group Publishing, Harrisburg, PA, pp.330-349.
- [16] Kettinger, W. J., J. T. C. Teng and S. Guha (1997), "Business Process Change: A Study of Methodologies, Techniques, and Tools," *MIS Quarterly*, vol.21, No.1, pp.55-80.
- [17] Mayer, R. J., P. C. Benjamin, B. E. Caraway and M. K. Painter (1995), "A Framework and a Suite of Methods for Business Process Reengineering," in *Business Process Change: Concepts, Methods and Technologies*, Idea Group Publishing, Harrisburg, PA, pp.245-290.
- [18] Pressman, S. Roger (2004), *Software Engineering: A practitioner's approach*, McGraw-Hill, NY.
- [19] Schriber, T. J. (1969), *Fundamentals of Flowcharting*, Wiley, New York.
- [20] UML (1997), UML Proposal to the Object Management Group, <http://www.rational.com/uml>.
- [21] Valacich, J. S., J. F. George and J. A. Hoffer (2005), *Essentials of System Analysis and Design*, Prentice-Hall, New York.
- [22] Weber, R. (2003), "Conceptual Modeling and Ontology: Possibilities and Pitfalls," *Journal of Database Management*, vol.14, No. 3.
- [23] Wolstenholme, E. F., S. He nderson, and A. Gavine (1993), *The Evaluation of Management Information Systems: A Dynamic and Holistic Approach*, John Wiley & Sons, Chichester, U.K.
- [24] Yourdon, E. N. (1989), *Modern Structured Analysis*, Prentice-Hall International, Englewood Cliffs, NJ.