

비즈니스 프로세스 맵을 이용한 시스템 요구 분석 방안 연구

나정은, 송주석
연세대학교 컴퓨터과학과
e-mail : jenah@yonsei.ac.kr; jssong@yonsei.ac.kr;

Requirements Analysis based on Business Process Knowledge for Requirements Engineering

Jeong-Eun Nah, Joo-Seok Song
Dept. of Computer Science, Yonsei University

요 약

Waterfall 모델에서의 소프트웨어 개발 단계는 사용자로부터 요구 사항을 정의하고, 이것이 다음 단계의 input 이 되어 개발자에 의해 작업이 진행되는 개발 양식을 따른다. 시스템 요구 사항을 정의할 때에, 프로세스 관점에서 사용자 업무를 이해하고, 이 프로세스에서 사용되는 시스템과의 관계를 정의하여 요구 사항으로 함께 정의한다면, 단순한 시스템의 기능을 정의하여 개발하는 것 보다 시스템의 사용 목적을 이해하고 개발하게 됨으로써, 더 만족도 높은 시스템으로 개발할 수 있게 된다. 프로세스 관점에서 이해하고 정의하는 시스템의 요구 사항은 시스템 보수 작업(Maintenance)시에도 시스템의 상황이 현재 업무 프로세스와 어떻게 연계되어 있는지 파악하기 용이하고, 테스트 단계에서도 테스트 조건이나 테스트 시나리오로서도 활용이 가능하다. 새로운 업무를 담당하는 사람에게도 시스템 개발 요구 사항에서 정의된 업무 프로세스 맵(Map)이 관련 업무를 파악할 수 있는 기초 자료로 재 활용 될 수도 있다. 비즈니스 프로세스 리엔지니어링에서 사용하는 비즈니스 프로세스 맵핑(Mapping) 기술을 소프트웨어 개발 방법론에 적용하여, 요구 분석 단계에서 사용하는 분석 도구로서의 응용을 제안하고자 한다.

1. 서론

비즈니스 프로세스에 대한 관심은 여러 분야에서 다양하게 주목하여, 기존의 업무 기능을 프로세스의 새로운 관점으로 검토하려는 시도들로 많이 나타나고 있다. 세로로만 보던 기준을 가로의 기준으로 재 조명하려는 것과 같이, 기능 중심의 관점을 프로세스 중심의 관점으로 업무를 재검토함으로써, 효율과 효과를 극대화하려는 의도이다.

조직 내에서는 프로세스의 관점이 개인/단일 조직의 업무 분장의 영역을 벗어나, 업무와 관련된 다른 사람/다른 조직과의 연계까지 고려되어, 조직간의 업무의 흐름을 함께 파악할 수가 있게 된다. 업무 처리에 지원되어야 하는 시스템의 역할도 프로세스를 지원하는 시스템 체제로 전환되어 분석 또는 설계되어야 한다.

프로세스를 지원하는 시스템은 요구 사항을 기능(function)별로 구분하여 정의하기 보다 프로세스를 지원하는 기능으로 새롭게 정의 되어야 하며, 업무 프로세스를 이해함과 동시에 시스템의 역할 및 요구 사항에 대한 범위가 무엇인지를 파악하는 과정으로 재정의되어야 할 필요성이 있다.

최근에는 여러 산업 분야에서도 프로세스 관점으로 보려는 시각이 강조되는 것과 동시에, 전체를 총괄하는 통합적이 시각 또한 중요한 점으로 부각되면서, 시스템을 통합하거나 새로운 시스템으로 개발함으로써, 새로운 프로세스를 창출하는 역할로서의 시스템을 보려는 경향도 늘고 있다. 비즈니스 리엔지니어링 분야와 e-비즈니스 분야, 인터넷 관련 비즈니스 사업에서도 새로운 패러다임의 전환으로 업무 프로세스를 전체적으로 다시 조명하여 새로운 비즈니스 형

태와 업무 처리 형태로 재 창조하는 것과 동일하게, 소프트웨어 개발 분야에서도 개발된 시스템의 역할이 단순 지원 역할에서 벗어나 새로운 흐름을 리드하는 역할까지 담당할 수 있다고 본다.

본 논문에서는 업무 프로세스에 대한 검토에서부터 출발하여 시스템의 역할을 살펴보고, 업무 프로세스의 설계/재설계 과정에서의 시스템의 역할을 소프트웨어 공학적인 접근으로 요구 사항 정의 단계에 적용해 보고자 한다.

2. 업무 프로세스와 적용 업무 시스템

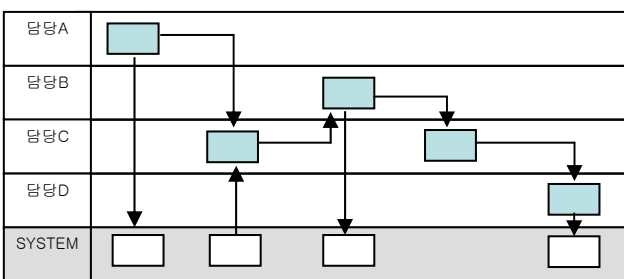
2.1 업무 프로세스 상에서의 시스템의 역할

적용 업무 시스템에서의 시스템의 역할은 업무 지원을 위한 도구로 사용되기 시작했지만, 이제는 그 도구의 역할이 작업 프로세스의 중요한 **Enabler** 가 되어 시스템 지원이 없는 작업 프로세스는 생각하기가 어렵게 되었다. 비효율적인 작업을 개선하기 위한 대안으로 출발했던 시스템의 기능이 관련 분야의 새로운 적용 기술의 등장으로 새로운 업무 프로세스로의 변화로까지 진행되어, 시스템의 지원이 e-비즈니스와 같은 새로운 비즈니스 모델로의 전환까지도 가능하게 되었다.

2.2 프로세스 맵

비즈니스 프로세스 리엔지니어링에서는 연관된 업무를 수행하면서 최대 결과를 얻기 위해 프로세스의 재설계 과정을 수행한다. 사용자에게 의해 **USE-CASE** 로 표현되었던 것들이 **BPR** 과정을 통해 수정되거나 새로운 **USE-CASE** 로 재 탄생하다. 이것을 **USE-CASE** 나 시나리오로서 설명하기 보다, 프로세스 맵을 통해 시각적으로 한 눈에 볼 수 있게 정의해 보는 방법이 있다.

비즈니스 프로세스 리엔지니어링 과정에서 사용하기도 하는 프로세스 맵을 아래 <그림 1>에서 살펴보자. 업무 프로세스의 정리하기 위한 방법으로 누가(who)-조직/담당자, 무엇을(what)-대상 업무, 어떻게(how)-업무 처리 방식이 작업 처리 흐름에 따라 기술될 수 있다. 이 때, 업무 처리 과정에서의 시스템의 기능 및 역할을 함께 정의/기술하도록 제안한다.



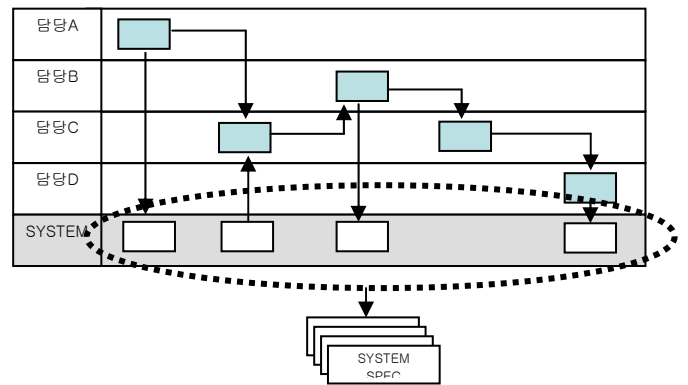
<그림 1> 업무 프로세스 맵

위와 같은 형태의 프로세스 정의는 현행 업무를 파

악함과 동시에 새로운 시스템에로의 요구 사항이 무엇인지, 업무 처리의 기능과 함께 도출될 수 있다. 누가 작업을 하게 되는지 기술되게 됨으로써 조직간의 관련성도 작업 프로세스 상에서 그대로 반영되게 된다.

2.3 시스템 기능의 재 정의(Redefinition)

이렇게 기술된 시스템의 요구 사항을 시스템 관점으로 다시 정리하면, 시스템에서 담당해야 할 업무 범위로 한정하여 정리할 수 있게 된다. 시스템 요구 사항은 처리 과정과 업무 분야에 따라 몇 개의 시스템으로 구분하여 정의될 수 있고, 이를 바탕으로 관련된 기능을 시스템으로 구분하여 각 시스템 별 요구 사항을 종합할 수 있다. <그림 2>는 시스템의 요구 사항을 수렴하여 관련 기능별로 재 정의하여 시스템 개발 범위(System Development Scope/Specification)로 구성될 수 있음을 보여 준다.



<그림 2> 시스템 개발 범위 도출

각 시스템 별로 재 정의된 업무 기능은 **Function Decomposition Diagram** 으로도 표현 가능하며, 시스템 디자인 단계에서는 프로세스의 표현보다 시스템 내에서 정의되는 기능들로 표현하는 것이 개발 시스템 스펙(Specification)으로서 더 이해하기 쉬운 경향도 있다.

2.4 프로세스 개선을 위한 새로운 기능 제시

사용자/고객의 요구 사항을 소프트웨어 기술 요구 기능으로 정의 하는 **Quality Function Deployment(QFD)** 품질 관리 기법에서도, 개발자가 정리한 요구 사항이 사용자/고객이 제시했던 요구 사항을 그대로 정리하는 것 보다 업무에 더 도움이 되는 다른 방안을 함께 제시할 때에 매우 만족해 한다고 한다[1]. 요구되는 사항만을 정리하기 보다 업무의 흐름을 파악하고 필요한 기능과 기술적으로 가능한 시스템 지원 대안을 함께 제시하려면, 업무의 흐름 파악은 기본이며, 관련 업계의 기술 동향까지도 파악하고 있어야만 가능하다. 업무 프로세스를 먼저 파악하고 있어야만, 새로운 기술의 도입을 검토하고 새로운 안을 제시할 수 있게 되며, 새로운 기술 도입으로 인해 기존의 업무 프로세스가 재 정의될 수 있게 된다. 이렇듯, 새로운 기술

의 도입으로 업무 흐름을 재 설계할 수 있다는 가능성을 보여 준다. 시스템을 기능 중심으로 볼 때 새로운 아이디어로 제시할 수 없던 것들이 프로세스 중심의 사고로 전환됨으로써 업무 지원을 더 효과적으로 지원할 수 있는 방안이 마련 될 수 있다.

3. 프로세스 지식과 요구 사항 분석

3.1 프로세스 지식(Knowledge) 모델

시스템 개발 과정의 요구 사항 분석 및 정의는 최종 시스템의 품질(Quality)을 평가하는 매우 중요한 요소이며, 어떠한 요건의 시스템으로 만들지 정확한 요구 사항을 정의한다는 것은 소프트웨어 시스템 개발 단계에서도 매우 힘든 과정이다[2].

신뢰할 수 있는 시스템 요구 사항은 데이터 처리 과정과 그것이 어느 업무 처리 과정에서 발생하는 것 인지를 동시에 표현 할 수 있어야 한다. 이것을 가능하게 하는 것이 <그림 1>의 프로세스 맵에서의 표현 처럼 업무 프로세스와 동시에 시스템 처리 과정을 표현함으로써 가능해 진다. 즉, 시스템 요구 사항은 시스템의 기능뿐 아니라 업무 처리 프로세스와 함께 기술되어야 한다.

비즈니스 리엔지니어링 시나리오를 이용한 모델링은 새로운 프로세스 설계를 위해서도 사용할 수 있지만, 현재의 시스템을 이해하는 데에도 좋은 도구로 사용 가능하며, 현재의 문제점을 이해함으로써, 새로운 프로세스 설계도 가능하게 한다[3].

소프트웨어 시스템 요구 사항은 여러 단계의 추상 레벨(Abstraction Level)로 표현 가능하지만, 사용자와의 요구 사항을 정의 하는 도구로 사용하기 위해서는 개발자의 입장이 아닌 사용자의 요구에 맞는 레벨로의 기술이 필요하다. 이 때에 프로세스 지식(knowledge)이 요구 사항을 파악하기 위한 중간 도구로써 역할을 하게 된다[4].

3.2 요구 사항의 중요성

적용 업무 시스템은 사용자/고객 또는 시장의 필요성에 맞게 구축되어야 한다. 역으로, 이 필요성은 시스템 요구 사항으로 잘 정의되어야 한다. 요구 사항이 잘 정의되어 있지 않으면, 개발자는 무엇을 개발하여야 하는지 알지 못하며, 또한 사용자도 무엇을 기대하는지 알지 못하게 된다. 이런 과정에서는 개발된 시스템을 검증하기도 힘들게 된다. 따라서 요구 사항을 제대로 이해하기 위해서는 시스템으로 처리해야 하는 과정을 포함한 전체 업무 프로세스를 이해하는 것이 보다 효과적일 것이다. Requirements Engineering 관점에서는, 요구 사항 정의 단계가 만족스럽지 못하면, 사용자/고객의 개발팀에 대한 신뢰도까지 만족스럽지 않게 되어, 완성된 전체 시스템의 신뢰도가 낮아진다고 한다[5].

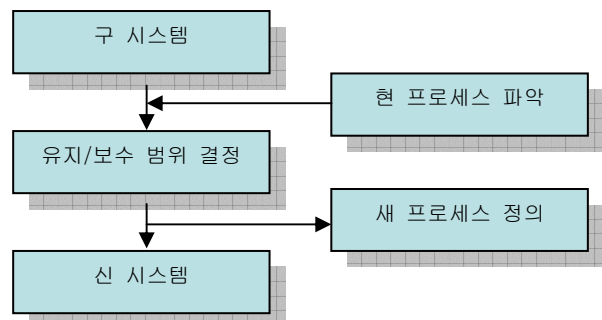
4. 요구 사항에서 도출된 프로세스 지식의 활용

4.1 현 업무 지원 시스템에 대한 이해

프로세스 맵을 이용하여 새로운 시스템의 요구 사항이 정의 되면, 여기에서부터 출발한 시스템 상세 스펙을 근거로 소프트웨어 시스템을 개발하게 된다. 개발 후에도 각 시스템이 어떤 역할과 절차로 업무 프로세스를 지원하는 지 파악할 수 있는 문서로서의 이용이 가능하다.

4.2 시스템 유지 보수

시스템 유지 보수가 필요한 시점에서 소프트웨어 관련 범위를 쉽게 평가할 수 없는 경우가 있을 수 있다. 시스템의 기능 개선이나 유지 보수 과정에서 필요한 As-Is 상황 파악은 시스템 개발 요구 단계에서 생성된 프로세스 맵으로부터 시작 할 수 있다. 요구 사항이 반영되어 변경된 시스템 기능과 프로세스의 개선 과정은 맵 상에서 상위 레벨로 재정의되어 표현 되어야 한다. Reverse Engineering 은 현재의 Source Code 로부터 디자인 과정을 추상적인 수준으로 추출하여 새로운 기능의 프로세스로 대체하는 과정인데[6], 이 과정에서도 <그림 3>에서와 같이 시스템에 대한 이해와 정보의 흐름을 추상적인 레벨로 표현할 때에 업무 프로세스 맵의 사용이 효과적일 것으로 기대한다.



<그림 3> 시스템 유지/보수 과정에서의 프로세스 이해

4.3 교육 자료

시스템 요구 단계에서 작성된 비즈니스 프로세스 맵은 업무를 파악하기 위한 교육 자료로의 활용도 가능하다. 프로세스로의 표현은 개괄적인 업무의 흐름과 업무 과정상의 시스템의 지원 관계를 명확히 파악할 수 있게 한다. 업무 분야를 프로세스 맵으로 정의해 놓은 경우, 전 프로세스 맵을 이해함으로써, 서로 연관된 업무의 흐름 파악은 물론, 동일 시스템에서 처리되는 프로세스를 이해함으로써 연관된 업무의 파악에도 매우 유용한 자료로 활용 가능하다.

4.4 테스트 시나리오

시스템 개발 이후의 테스트 단계에서는 각 단위 프

로그별 테스트(단위 테스트)와 관련된 기능들을 함께 테스트(통합 테스트)하는 과정이 끝난 후, 실제 업무 처리 과정에서 발생할 여러 가지 다양한 경우에 대한 테스트(시스템 테스트, 적용 테스트, 운용 테스트 등)를 거쳐야 한다. 사용자 관점에서는 운용 환경에서의 처리가 매우 익숙하지만, 개발자의 경우 반드시 그러한 것은 아니다. 따라서 사용자 단계의 테스트(User Acceptance Test) 이전에, 요구 사항으로 정의된 프로세스 맵을 이용하여 테스트 시나리오로 활용 가능하다.

업무 처리 과정 상의 프로세스가 항상 일정한 패턴으로 발생하지는 않더라도, 기본적인 주요 기능을 테스트 하기 위한 조건으로 사용 가능하다. 테스트 케이스 마련을 위해 다른 문서 형태로 만드는 과정을 생략할 수 있다. 요구 사항 정의 단계에서 생성된 문서가 Waterfall 모델에서처럼 다음 단계의 INPUT 자료로만 사용되는 것과 달리, 연관된 개발 단계에서 요구 사항 문서가 지속적으로 사용됨으로써, 설계 단계부터 정의된 요구 사항들이 연관된 개발 과정에서 동일한 시각으로 계속 사용되는 것이 가능하게 된다.

5. 결론

시스템 요구 사항을 정리하는 일반적인 방법은 시스템 별로 요구하는 기능이 무엇인지 기능별 나열이 우선되기 쉽다. 업무 지원 시스템의 경우, 시스템의 역할은 업무 프로세스의 효율적, 효과적인 지원과 아울러, 새로운 시스템의 도입으로 기존의 업무 흐름을 새로운 프로세스로 변화시킬 수 있는 위치에 오게 되었다.

따라서, 요구 사항을 수립하는 단계부터 단순 시스템에로의 요구 사항으로 정리하지 않고, 업무 프로세스를 어떻게 지원할 것인가의 요구 사항으로 수립한 후, 종합된 요구 사항을 연관된 기능 또는 연관된 프로세스 별로 재 정의하여 각 시스템 별 요구 사항으로 정리하려는 방법을 생각해 보게 되었다.

이전에는 시스템의 기능을 신규 정의하거나, 유지/보수 단계에서 시스템 기능 중심으로 접근했던 방식을, 비즈니스 프로세스 리엔지니어링에서 사용하는 맵핑 기술을 이용하여 프로세스 평가와 재설계에 필요한 요구 사항 수립 도구로 사용하고자 한다.

요구 사항에서 정리된 프로세스 맵핑 문서는 현 프로세스를 파악하기 위한 자료뿐만 아니라 시스템 유지/보수 과정에서의 기초 자료로 사용이 가능하며, 새로이 업무를 파악하려는 사람에게 교육 자료로 제공되거나, 시스템 테스트 과정에서도 테스트 시나리오로 사용 가능하다.

시스템 개발 이후 테스트 단계에서는 각 단위 프로그램 별 테스트와 관련된 기능들을 함께 테스트하는 과정이 끝난 후, 실제 업무 처리 과정에서 발생할 수 있는 다양한 경우에 대비한 테스트를 거쳐야 한다. 사용자 관점에서는 운용 환경에서의 처리가 매우 익숙하지만, 개발자들에게는 작업 흐름에 따른 테스트가

쉽지 않을 수 있다. 이런 경우 개발자들에게는 요구 사항으로 정의된 프로세스 맵을 테스트 시나리오로 활용할 수 있다.

시스템의 요구 사항을 지속적인 프로세스 관점에서 파악하는 것이 시스템 요구 사항을 기능으로 정의하는 것보다 시스템의 역할을 향상 시키는 방안이다. 이제 시스템의 역할은 지원 도구로서가 아니라 새로운 프로세스로의 전환을 가능케 하는 Enabler로서의 역할을 새로이 담당하게 되었기 때문이다.

사용자 요구 사항 관리(Requirements Management)는 Waterfall 모델의 요구 사항 정의 단계로 국한되는 것이 아니라, 전 시스템 Life Cycle 과정에서 지원되어야 하는 변화 관리(Change Management)이며[7], 이 때문에 업무 프로세스와 함께 리엔지니어링(Reengineering) 프로세스 관점에서 시스템 요구 사항이 정의되어야 한다.

참고문헌

- [1] Roger S. Pressman, "Software Engineering: A Practitioner's Approach," 5th Ed. McGraw-Hill, 2001.
- [2] F. P. Brooks, "No Silver Bullet: Essence and Accidents of Software Engineering," IEEE Computer, April 1987.
- [3] Jarzabek, S., Tok Wang Ling. "Model-based design of tools for business understanding and re-engineering," Computer-Aided Software Engineering, Seventh International Workshop on, 10-14 July 1995, pp.328 - 337.
- [4] Ramesh, B., Luqi, "Process knowledge based rapid prototyping for requirements engineering," Proceedings of IEEE International Symposium on, 4-6 Jan. 1993, pp.248-255.
- [5] IEEE Task Force on Requirements Engineering Home Page; <http://www.shu.ac.uk/tfre/index.html>
- [6] Chikofsky, E.J., and J.H. Cross, II, "Reverse Engineering and Design Recovery: A Taxonomy," IEEE Software, January 1990, pp.13-17.
- [7] P. Bourque, et al, The Guide to the Software Engineering Body of Knowledge, IEEE Software, Vol. 6, Nov/Dec.1999, pp. 35-44.