금융정보시스템 위험관리의 현황 및 개선을 위한 제언

A Proposal for Improvement and Current Situation of Risk Management on Financial Information System

강 태 홍 (TaeHong Kang) 숭실대학교 일반대학원 컴퓨터학과, 교신저자
류 성 열 (SungYul Rhew) 숭실대학교 IT대학 교수

요 약

금융정보시스템의 효율적인 운영을 위해서는 사전 예방관리를 기반으로 한 위험 대처능력의 향상이 매우 중요하다. 위험 대처능력을 향상하기 위해서는 기 발생된 위험에 대한 철저한 이해와 분석 및 대책 수립은 필수적이다. 본 연구에서는 금융정보시스템의 정보계, 계정계, 업무계 및 매매체결시스템에서 발생한 4년 5개월간의 장애 데이터를 도메인, 장애요소, 기간 및 요일, 개발단계 및 장애 원인별로 분류·분석하였다. 분석 결과 장애 데이터가 개발단계, 요일, 그리고 원인 별 특성과 추이를 가지고 있음을 확인 하였으며, 정보시스템 간 업무의 연관성으로 인하여 금융 도메인 전체의 위험관리를 위한 위험예보모델 구축의 필요성이 발견되었다.

키워드 : 금융정보시스템, 위험관리, 위험예보모델

Ⅰ. 서 론

우리나라 증권투자자 수는 2010년 말 기준 479만 명으로 경제활동인구 5명 중 1명은 주식투자를 하고 있으며(윤재수, 2011), 하루 평균 거래대금이 10조 원에 육박할 만큼 거대한 규모(이철환, 2011)이다. 이와 같은 경제활동이 가능한 것은 정보시스템의 발달에 기인한 것으로, 정보시스템의 결함은 이러한 경제활동에 영향을 주어 시장을 통하여 자금을 조달해야 하는 기업이나 개인들에게 매우 큰 위험이 된다. 따라서 금융정보시스템은 국가적으로 사회적으로 매우 중요한 시스템(이영희 역, 2012)으로써 금융정보시스템에 대한 위험은 곧 국민 경제활동에 대한 위험인 동시에, 국가와 사회에 대한 위험으로 간주된다.

예를 들어, 2009년 3월 23일 한국거래소가 차세대시스템을 가동한 이후 채권시스템의 장애로 인하여 3일 간 투자자들이 큰 혼란을 격은 일(www.nocutnews.co.kr)이 발생했으며, 해외에서는 2012년 5월 18일 미국 시장에 상장 예정이었던 페이스북이 정보시스템의 장애로 인하여 30분 간 지연됨으로써 수 많은 투자자와 은행 및 기관들이 거래를 할 수 없어, 1억 달러가 넘는 손실을 보았을 뿐만 아니라, 수백만 명의 투자자들이 불편을 겪었다(www.zdnet.co.kr).

금융정보시스템은 금융회사의 정보계, 계정계 그리고 업무계를 이루는 컴퓨터시스템을 의미한다. 정보계는 주문, 체결에 관한 정보를 처리하

여 투자자에게 제공하며, 계정계는 계좌관리, 출납, 주문 등을 처리하는 원장시스템의 업무를, 업무계는 회계, 위험관리 등의 업무를 의미한다. 이러한 금융정보시스템은 거래소의 매매체결시스템과 긴밀하게 연결되어 있다. 본 연구에서 금융정보시스템은 거래소의 매매체결 시스템(유가증권시장, 코스닥시장, 선물시장, 옵션시장)과 정보계 시스템, 그리고 이와 연결된 계정계와 업무계 시스템을 의미한다. 이러한 금융정보시스템은 국민의 경제활동을 위한 기반인 동시에, 국가 경제의 기반으로서, 금융정보시스템에 문제가 생기면 서비스의 차원을 넘어서 국가적으로나 사회적으로 매우 큰 영향을 받는다. 그러므로 금융정보시스템에 대한 위험관리는 매우 중요한 사항이다. 금융정보시스템의 위험관리에 있어, 사전예방이 사후처리보다 효과적이라는 것은 자명한 사실이다. 2011년 소프트웨어공학 백서(정보통신산업진흥원, 2011)에 따르면(<표 1> 참조), 42개의 프로젝트에서 품질 비용이 61%로 조사되었는데, 이는 결함을 예방하기 위한 예방비용과 평가비용이 늘어나면 식별되는 결함이나 문제점이 많아지게 되나, 이를 해결함으로써 전체적인 품질비용은 낮아지게 된다. 그러므로 정보시스템에 대한 위험관리를 예방체계로 전환하여 전체적인 품질비용의 향상을 꾀하는 것이 바람직하다.

〈표 1〉 품질비용 구성 비율

구 분	예방 비용	평가 비용	내부 실패비용	외부 실패비용	총 품질비용
비율	3%	8%	28%	21%	61%

정보시스템 위험관리를 위하여 필요한 자료는 과거 결함에 대한 자료이다. 그러나 결함에 대한 추이나 특징을 알 수 있는 자료는 조직의 기밀 자료로써 공개되지 않거나, 축적된 자료의 부족으로 인하여 이러한 자료를 이용하여 진행한 연구가 드물다. 본 연구에서는 개별 금융회사나 기관의 정보시스템 위험관리에 도움이 되도록, 4년 5개월 간 자본시장의 인프라 및 금융정보시스템에서 발생한 문제점, 결함, 실패 그리고 장애를 요소 별, 업무 도메인 별, 원인 별, 시기별로 빈도 수와 추이, 특징 등 과거 결함 자료의 분석된 내용을 제공한다. 제공된 결과를 활용하면 이를 기반으로 결함을 미리 예측하는 것이 가능하다. 본 연구는 금융정보시스템의 결함을 예측하는 모델에 필요한 기초 자료를 제공함과 동시에 효과적인 위험관리를 위한 방향에 대하여 제안하였다.

Ⅱ. 관련 연구

2.1 예보

'예측'의 사전적 의미는 '미리 헤아려 짐작하는 것'이며, '예보'의 사전적 의미는 '앞으로 일어날 일을 미리 알리는 것'이다. 우리가 가장 접하기 쉬운 일기예보는 일정 수준의 확률을 기본으로 여러 가지 날씨에 대한 정보를 분석하여 예측한 후 일반에게 공개(예보)한다. 이 예보는 항상 정확하게 맞지는 않지만 이 정보를 통해 많은 사람과 기업들이 일상생활과 업무에 많은 도움을 받는다. 예보를 위해서는 관측소나 Agent 같이 자료를 수집할 수 있는 도구 및 체계가 필요하다. 금융정보시스템의 경우에도 과거 자료 수집 및 분석이 매우 중요하며, 이를 기반으로 예보를 위한 모델 설정이 가능하다. 일반적인 위험의 정의와 정보시스템에 대한 위험의 정의는 다르다. 일기예보나 산불예보에 있어서 위험은 가뭄이나 홍수, 혹은 산불 같은 것들이다. 그러나 정보시스템에서의 위험은 여러 가지로 정의될 수 있다.

2.2 정보시스템의 위험

미국 상무성의 정보시스템 위험관리지침은 위험을 "발생 가능성과 그 결과로 인한 영향에 대

한 함수"로서 정의한다. 다시 말해, 잠재적인 취약성이 발현할 가능성과 그것이 미치는 영향으로 파악하고 있는 것이다(Gary Stonebumer, 2002). 또한 위험을 요소(component), 요인(factor), 확률(Probability)로 분류하였는데, Risk Component는 부정적인 결과의 형태를 분류한 것이며 Risk Factor는 손실(loss)을 유발하는 요인으로서 위험을 분류하였다. 또한 위험 확률은 위험을 경제적 손실에 중점을 두어, 부정적인 결과의 가능성을 통계적인 기술로 추정하여 금전적인 손실로서 표현하고자 하였다(CAIS, 2004). IEEE에서는 위험을 Defect(결함), Failure(실패), Fault(장애), Error(오류), Problem(문제)로 분류하였다(IEEE Std 1044, 2010). 본 논문에서는 이 분류 기준을 채택하였는데, 그 이유는 금융정보시스템의 결함은 서비스에 미치는 영향의 정도에 따라 경미한 결함부터 심각한 결함까지 다르게 구분되는데 제 3.2절의 정의에서 언급한 바와 같이 IEEE Std 1044가 이 기준에 가장 적합하기 때문이다. <표 2>와 같이 2005년 당시 정보통신부에서 작성한 정보시스템 운영관리지침의 분류 기준을 준용하였다. 이 지침은 장애를 통제 가능한 것과 통제 불가능한 것으로 분류하여 정의하였다. 즉, "장애"는 정보시스템의 통제 가능한 요인들로 인한 기능저하, 오류, 고장을 의미하며, 발생원인 관점에서 인적 장애, 시스템 장애, 기반구조 장애 등과 같은 통제 가능한 요인들과 자연재해와 같이 통제 불가능한 요인들로 구분하였다.

2.3 정보시스템의 위험예측

정보시스템에 대한 위험의 정의와 함께 장애나 결함 등을 예측하고자 하는 많은 시도들이 있었다. (Graves *et al.*, 1998)는 소프트웨어를 노화(Aging)되는 것으로 정의하고, 노화의 정도를 일정 기간 동안 코드에서 발생하는 장애의 수(number of fault)로 정의하였다. (Gunter *et al.*, 2004)는 소프트웨어의 가용성 증가를 위하여 공간적 redundancy 기술을 이용하거나 시간적 redundancy를 이용할 수 있는 선제적 대응방법(proactive method)를 개발하였다. (Zimmermann *et al.*, 2008)은 소프트웨어 출시 전에 결함이 가장 많을 가능성이 있는 모듈에 자원을 집중 투입함으로써 결함(defect)를 제거하기 위하여 결함을 야기하는 요인들을 선택하여 검증하였다. (이영재 등, 2007)은 하드웨어 장애로 인한 피해를 감소시키기 위

〈표 2〉 정보시스템 재해 및 장애의 분류와 대응방안

통제	재해 및 장애			재해 및 장애의 요인	장애 대응방안
통제 불가능 요인	자연 재해			화재(전산실, 사무실), 지진 및 지반침하, 장마 및 폭우 등의 수재, 태풍 등	재해복구센터 구축을 통한 기기 및 프로그램의 이중화, 데이터 백업 및 소산 철저
	인적 재해			노조파업, 시민폭동, 폭탄테러 등	백업 또는 대체요원 확보
통제 가능 요인	인적 장애		운영 장애	시스템운영실수, 단말기 및 디스켓 등의 파괴 및 절취, 해커의 침입, 컴퓨터 바이러스의 피해, 자료누출 등	
	기술적 장애	시스템 장애		운영체제 결함, 응용프로그램의 결함, 통신프로토콜의 결함, 통신소프트웨어의 결함, 하드웨어의 손상 등	전산기기 이중화 및 프로그램 변경통제 강화, 재해복구 센터 구축을 통한 기기 및 프로그램의 이중화, 통신망 이중화, 전력공급 중단에 대비한 무정전설비(UPS) 및 발전 설비 구축
		기반 구조 장애		정전사고, 단수, 설비 장애(항온항습, 공기정화시설, 통신시설, 발전기, 공조기 등), 건물의 손상 등	

하여 모델을 설정하고 디스크 장애에 적용하여 검증하였다. 정보시스템 관점 외에 업무 별 도메인의 관점에서 살펴보면, 초기 회계정보 시스템에 대한 위험은 정보시스템을 다루는 인력과 시스템의 수준의 저하로 인식하고 이를 해결하고자 노력하였고(한인구 등, 1993) 그 후에는 해킹 등 보안문제를 위험으로 인식하여 이를 통합적인 관점에서 조명하고자 하였다(김영걸 등, 1998). 또한 자원의 손실을 위험의 한 요소로 인식(김법진 등, 2000) 하기도 하였고, 은행을 중심으로 신용위험에 의한 통합적인 위험관리시스템에 대한 연구가 이루어져왔다(정철용, 2002). 그 후에는 주로 개발프로젝트를 중심으로 프로젝트의 위험을 감소시키기 위한 연구들이 이루어져왔다(조숙진 등, 2006; 배재권 등, 2011). 그러나 회계정보시스템이나 카드사 또는 은행의 정보시스템과는 달리 금융정보시스템은 주문·체결 데이터의 송수신을 위하여 금융정보시스템들이 거래소 시장을 중심으로 매우 긴밀하게 연관되어 하나의 매우 큰 체제를 이룬다. 따라서 소프트웨어 자체의 결함뿐 만 아니라, 외부 상황의 변화와 처리하는 정보의 성격, 볼륨, 특성에 따라 지대한 영향을 받는다. 지금까지 대규모의 단일 업무 domain에서 발생한 결함 데이터가 체계적으로 정리된 바는 없다. 본 논문에서는 금융업계의 위험관리 현황을 파악하고, 결함이나 장애와 같은 위험 데이터를 체계적으로 정리하였다.

Ⅲ. 금융정보시스템의 위험관리 현황 및 분석

3.1 위험관리 현황 및 문제점

3.1.1 위험관리 현황

금융회사의 위험관리는 매우 초보적인 수준에 머물러 있다. 금융위원회의 권고에 따라 통제불가능한 요인인 지진이나 홍수 등 재난에 대비한 백업센터는 구축되어 있다. 또한 장애관리를 위한 업무 프로세스는 현황파악, 원인규명, 조치, 재발방지대책 등으로 설정되어 있다. 그러나 대부분이 일과성에 그칠 뿐만 아니라, 계속적인 추적관리가 이루어지지 않으므로 동일한 장애가 반복된다(이영희 역, 2012). 또한 한 조직에서 발생한 동일한 장애는 다른 조직에서도 반복되며 그로 인한 금전적 손실 또한 막대하다. 그러므로 조직마다 자체적으로 장애관리를 수행하기 보다는 업계 전체를 아우를 수 있는 통합된 장애관리체계가 필요한 상황이다. 서론에서 언급한 바와 같이 금융정보시스템은 개별 금융회사의 정보시스템들과 한국거래소의 매매체결 시스템, 정보분배 시스템이 매우 긴밀하게 연관되어 있다. 따라서 한 시스템의 결함으로 인하여 다른 시스템에 영향을 미치는 파급효과가 매우 크다. 그러나 이를 통제할 수 있는 체제는 갖추어져 있지 않다.

3.1.2 위험관리 문제점

장애발생은 특성 상 장애 데이터를 축적하고 관리하기가 어렵다. 장애 데이터의 축적은 곧 정보시스템 운영을 제대로 못한 증거로 인식되어 있기 때문이다. 위험관리에 있어 첫 번째 문제는 장애 데이터의 축적이 이루어지지 않는 것이다. 이로 인하여 장애의 현황 파악이나 분석이 어렵고, 동일한 실수와 장애가 반복되며 담당자가 바뀌면 데이터도 사라지는 경우가 많다. 둘째는 사전적 예방보다는 장애 발생 이 후의 복구 프로세스에 초점이 맞추어져 있다는 것이다. 장애 발생에는 반드시 원인이 있으며 그러한 원인들은 주로 장비의 노후화, 운영에서의 실수, 불완전한 개발, 그리고 절차화되지 못하거나 지켜지지 않는 프로세스 등에 기인한다. 이러한 원인과 요인들을 파악하고 예방에 관심을 갖는다면 사후조치 보다 훨씬 적은 비용으로 장애 발생률을 낮출 수 있다. 셋째는 금융정보시스템과 같은 중요한 시스템의 위험관리에 대한 통합적 관리의 부재이다. 금융기관 자체적인 위험관리에 따른 문제점

은 앞에서 지적한 바와 같다. 효과적인 위험관리를 통한 정보시스템의 효율적인 운영을 위하여 업계 공동의 노력이 필요하며 이를 위해서는 통합적 관리가 매우 중요하다.

3.2 위험의 정의 및 종류

본 논문에서 수집·분석한 위험의 정의 및 종류는 결함, 문제, 실패, 장애이다. 결함(defect)는 결과물이 불완전하거나 결핍된 것으로, 요구 또는 Spec을 만족시키지 못하는 것 또는 수리되거나 대체되어야 할 필요가 있는 것으로, 정보시스템의 포괄적인 위험을 결함이라 한다. 문제(problem)는 결함이나 장애로 발현되지는 않았으나, 시간의 경과나 조건의 충족 여부에 따라 결함이나 장애로 발전할 수 있는 위험을 의미한다. 실패(failure)는 고객의 불편을 초래하였거나 불만을 야기한 위험으로, 단순히 고객의 불편이나 불만을 야기한 경우와 서비스의 중단을 초래한 경우로 나눌 수 있다. 장애(fault)는 실패보다 심각한 것으로 고객에게 직접적인 손실을 입힌 경우와 시장의 중단을 초래한 경우이다. 두 가지 경우 모두 매우 심각한 상황을 초래한 것이며 금융정보시스템에 있어서 가장 치명적인 위험이라 할 수 있다.

3.3 자료 수집 및 분석 방법

본 연구의 조사·분석 자료는 금융업계에서 4년 5개월(2007년부터 2011년 5월)에 걸쳐 파악된 1,258건의 결함 가운데 서비스와 관련이 없는 건을 제외한, 797건의 문제(problem), 실패(failure), 그리고 장애(fault)를 대상으로 분류 및 분석하였고, 테스트 완료 후 시범 가동 기간에 발생한 장애도 포함되었다. 금융회사나 기관마다 장애의 기준이 매우 다르므로 여기서 정한 분석의 대상은 IEEE에서 정의한 문제, 실패, 그리고 장애를 대상으로 했으며, 분류 방법은 정보시스템 장애관리 지침에 따라 분류하였고 통제 불가능한 요인은 제외하였다.

3.4 장애현황분석

3.4.1 Domain별 시점 별 현황분석

<표 3>은 연도 별로 각 domain에서 발생하거나 검출된 문제, 실패, 그리고 장애의 건수를 나타낸다. A domain은 시장의 거래시스템을 의미하며, B domain은 계정계와 업무계 시스템을, C domain은 시세정보처리시스템을, D domain은 인증시스템을, E domain은 보안침해방지 시스템을, F domain은 BCP 시스템을, G domain은 부대설비 및 하드웨어운영시스템을 의미한다. 전체 결함 가운데 domain에 따라 상대적 비율이 3.8%에서 31.6%까지 큰 차이를 보이고 있다. 이를 좀 더 상세하게 알아보기 위하여 Domain별로 월별 결함 발생 데이터를 <표 4>에 정리하였으나 정보시스템의 결함은 일기예보나 산불예보와는 달리 월

〈표 3〉 Domain별 연도별 장애 현황

연도	A	B	C	D	E	F	G	합 계
2007	47	32	15	23		55		172
2008	43	26	23	37		34		163
2009	115	30	27	8	13	27	30	250
2010	25	34	26	6	17	24	21	153
2011	22	19	5	3		3	7	59
합계	252	141	96	77	30	143	58	797
비율(%)	31.6	17.7	12.0	9.7	3.8	17.9	7.3	100.0

〈표 4〉 Domain별 월 별 현황

구 분	1	2	3	4	5	6	7	8	9	10	11	12	합계
A	29	6	8	20	18	20	24	24	21	17	23	20	230
B	10	6	11	14	10	4	14	8	9	16	15	5	122
C	11	7	10	6	11	4	7	6	8	6	9	6	91
D	6	6	3	4	10	5	8	3	9	7	7	6	74
E		1	2	2	2	3	6	3	1	2	4	4	30
F	11	7	19	9	17	12	11	12	13	10	11	8	140
G	1	3	3	5	5	9	4	4	6	4	4	3	51
합계	68	36	56	60	73	57	74	60	67	62	73	52	738
비율(%)	9.2	4.9	7.6	8.1	9.9	7.7	10.0	8.1	9.1	8.4	9.9	7.0	100.0

별 특성은 파악되지 않는다.

<표 5>은 전체 장애를 요일 별로 분석한 결과로 월요일의 장애 비율이 다른 요일에 비하여 약 5%P~7%P 높은 것으로 나타났다. 이를 좀 더 상세히 분류하기 위하여 벤더의 제품에 의존하는 요소와 개발 및 운용에 관련된 요소로 분리하여 분석하면 <표 6>과 같다.

<표 6>은 개발 및 운용에 관련된 인적 장애로써 월요일의 장애 비율이 다른 요일에 비하여 15%P~35%P까지 매우 높다. 이는 주말을 이용하여 이루어지는 변경작업이 정보시스템의 안정성에 영향을 주기 때문이다. 즉, 벤더의 제품에 의존하는 시스템장애는 요일 별 특성 없이 골고루 발생하며 개발 및 운용에 관련된 장애는 요일 특성이 존재함을 알 수 있다.

3.4.2 Domain별 발생원인 별 현황

각 Domain별 결함발생의 원인 <표 7>과 같다. 모든 domain에서 가장 높은 비중을 차지하는 결함은 하드웨어 결함이다. 그러나 대부분의 하드웨어 장비는 Backup 체계를 갖추고 있어 실제 업무 중단이나 서비스 연속성에 미치는 영향은 크지 않다. 반면 데이터오류, 용량부족, 운용오류, 코딩오류로 인한 장애는 전체의 16.5%이나, 이로 인한 손해배상이나 고객의 불만 등 영향도(severity of impact)(Yuming Zhou, 2006)는 대단히 크다. 그러므로 소프트웨어 개발단계에서의 품질관리와 운영단계에서의 시스템 변경에 따른 결함관리가

〈표 5〉 Domain별 요일 별 현황

Domain	월	화	수	목	금	토	일	합계
A	57	35	36	43	33	20	28	252
B	25	23	20	23	20	13	17	141
C	22	19	12	20	13	6	4	96
D	27	7	7	12	15	2	7	77
E	5	7	5	1	2	7	3	30
F	21	28	20	22	26	10	16	143
G	9	8	9	7	9	11	5	58
합계	166	127	109	128	118	69	80	797
비율(%)	21	16	14	16	15	9	10	100

〈표 6〉 Domain별 요일 별 현황(데이터오류, 용량부족, 운용오류, 코딩오류)

Domain	월	화	수	목	금	토	일	합 계
A	34	11	10	16	2		1	74
B	13	9	5	10	3	1		41
C	9	3	3	7	6	3		31
D	6	2		2	10		1	21
E					1			1
F								
G		1		1				2
합 계	62	26	18	36	22	4	2	170
비율(%)	36	15	11	21	13	2	1	100

〈표 7〉 Domain별 장애요소 별 현황

속 성	A	B	C	D	E	F	G	합계	비율(%)
HW	150	95	71	55	30	134		535	67.1
NW	12		1	3		7		23	2.9
SSW	17	5	3	11		1	2	39	4.9
부대설비	2			4		1	56	63	7.9
데이터 오류	11	1	5					17	2.1
용량부족	6	7	5					18	2.3
운용오류	12	7	3					22	2.8
개발(코딩)오류	39	24	8	3				74	9.3
원인불명	3	2		1				6	0.8
합계	252	141	96	77	30	143	58	797	100.0

매우 중요하다.

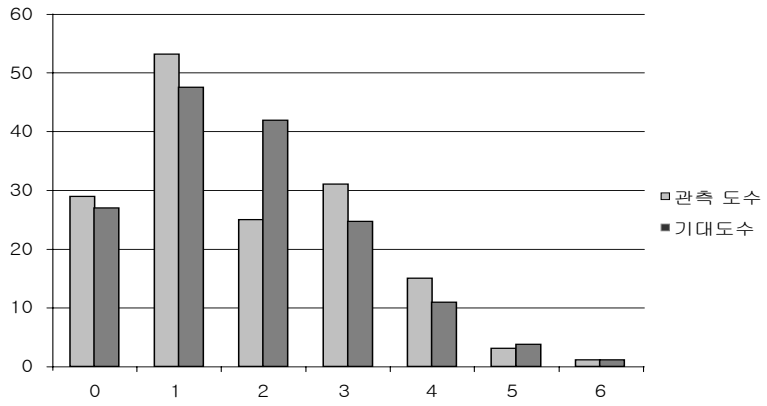
하드웨어 결함이 실제 서비스에 미치는 영향은 작으나, 빈도 수가 높으므로 정보시스템의 운영에 잠재적인 위험요소이다. 하드웨어의 결함 발생 데이터를 1주일 단위로 Grouping하여 157건의 주간 시계열 데이터로 만든 후, 그 분포를 알아본 결과는 <그림 1>와 같다.

데이터가 포아송 분포를 따른다고 가정하고 5%의 유의수준을 적용한 적합도 판정결과 포아송 분포라는 가정이 타당함을 알 수 있다. 이는 근사적으로 일주일에 평균 1.76건의 하드웨어 장애가 포아송 분포 형태로 발생한다는 의미이다. 포아송 분포의 특성은 주어진 공간이나 시간의 한 단위 내에서 사건이 발생하는 확률을 구하는 것이다. 즉, 하드웨어 장애는 무작위로 발생하는 것이 아니라, 일정한 확률분포의 특성에 따라 발생하므로, 발생 횟수나 가능성에 따라 사전 대책을 세울 수 있다. 또한 네트워크 장애도 포아송 분포를 따르나, 시스템 소프트웨어와 부대설비, 그리고 다른 장애는 통계적으로 유의한 분포를 갖지 않는다.

3.4.3 SDLC 상의 장애 현황

소프트웨어개발 생명주기 상에서의 결함 74건에 대하여 상세히 분석한 결과는 <표 8>와 같다.

일반적인 소프트웨어의 오류는 대부분 개발 초



주간장애건수	관측도수	포아송 분포 확률	기대도수	카이제곱
0	29	0.1713	26.89	0.164866
1	53	0.3022	47.45	0.649042
2	25	0.2666	41.86	6.790186
3	31	0.1568	24.62	1.654588
4	15	0.0692	10.86	1.5796
5	3	0.0244	3.83	0.180487
6	1	0.0072	1.13	0.014248
합계	157	1.00	157	**11.033**

유의수준	0.05
자유도	6
임계치	**12.59**
유의확률(p값)	0.0874

〈그림 1〉 하드웨어 결함발생 분포

〈표 8〉 SDLC 상의 장애현황

구 분		건수	비율	비율(소계)
요구정의	요구기능 누락	2	2.7%	9.5%
	기능정의 부정확	5	6.8%	
설계	부정확한 IF 설계	2	2.7%	14.9%
	IF/Timing 오류	1	1.4%	
	초기 설정 값 오류	5	6.8%	
	변수타입 오류	3	4.1%	
구현	logic 구현 오류	24	32.4%	64.9%
	변수 값 오류	1	1.4%	
	데이터처리 오류	11	14.9%	
	데이터 초기화 오류	2	2.7%	
	계산오류	5	6.8%	
	함수/변수 사용 오류	5	6.8%	
테스트	테스트범위 오류	1	1.4%	1.4%
이행	파일복사/이동 오류	3	4.1%	9.5%
	프로그램 복사/이동 오류	4	5.4%	
합계		74	100.0%	100.0%

기의 문제에 주로 기인한다. 기존의 논문에서 발표된 오류 발생 원인은 명세(Specification)의 오류가 56%, 설계(Design)의 오류가 27%, 구현(Code)의 오류가 10%, 기타 오류가 7%를 차지한다. 그러나 금융 도메인의 경우, 응용 프로그램 오류는 대부분 구현단계의 문제에 기인하는 경우가 많다. 그 이유는 타 도메인에 비해 업무가 사전에 명확히 정의될 수 있기 때문이다. 이는 소프트웨어의 요구사항이 명확하다는 것으로 요구정의나 요구분석, 설계단계보다는 구현과 테스트, 이행단계의 오류가 장애의 대부분을 차지하는 원인일 가능성이 높음을 의미한다. 실제로 <표 8>에서 보는 바와 같이 SDLC 상에서의 장애를 분석해 본 결과, 요구정의나 설계에 기인하는 문제보다는 구현 단계에서의 오류로 인한 장애가 많다. 따라서 일반적인 소프트웨어 결함 예방활동을 적용하기보다는 금융 도메인의 특성을 감안한 결함예방활동의 수립 및 적용이 필요하다고 볼 수 있다.

3.5 분석결과

연도 별 Domain별 분석 결과, 결함 발생 빈도수가 domain에 따라 큰 차이가 있다. 그 이유는 업무의 복잡도나 특성, 변경업무의 볼륨 및 빈도, 리소스, 신규 시스템 도입 등의 이벤트로 인하여 결함발생 빈도가 다른 것이다. 시기 별로는 월별 특성은 존재하지 않으나, 요일 별 특성이 존재하는 것으로 확인되었다. 하드웨어 벤더에 의존적인 시스템장애는 요일에 관계없이 발생하지만, 시스템 운영이나 소프트웨어 개발과 관계된 인적 장애는 월요일의 비율이 가장 높다. 이는 주말을 이용한 시스템 개선, 제도 개편, 고객요구에 의한 프로그램의 변경으로 인한 것이며, 그 외 다른 시기 별 특성은 존재하지 않는다. Domain 별 발생원인 별 분석 결과, 하드웨어의 결함 비율이 가장 높다. 특히 신규시스템의 가동이나 하드웨어 의존적인 업무 domain의 경우 더 높게 나타났다. 이는 신규 시스템의 경우 안정화되기까지 결함발생과 같은 위험이 더 높다는 것을 의미한다. 또한 하드웨어의 결함과 네트워크의 결함은 통계적으로 포아송 분포를 따르는 것으로 나타났으며 다른 결함은 유의한 통계적 분포를 갖지 않는다. SDLC 상의 분석 결과, 일반 소프트웨어의 결함과는 달리 금융정보시스템의 결함은 구현 단계에서의 오류가 많은 것으로 나타났다. 그 이유는 업무와 사용자의 운용패턴이 법규나 규정 혹은 업무절차에 의해 명확하게 사전에 정해지므로, 요구정의나 설계단계보다 구현단계에서 오류가 상대적으로 더 많은 것이다.

이와 같은 결과는 업무 담당자들이나 CIO가 자원의 배치 및 운용에 있어 시기, 구성요소, Domain 별 특성을 이해하고 그에 맞게 정보시스템을 운용하는 것이 결함의 예방 및 조치에 효율적임을 나타낸다. 또한 다른 소프트웨어와 달리 금융정보시스템의 소프트웨어 결함은 주로 구현 단계에서 많이 발생함으로 이를 줄이기 위하여 구현에 대한 도급화 지양, code inspection 강화 등의 조치가 필요함을 나타낸다.

Ⅳ. 결 론

본 연구에서 제공한 것처럼 결함에 대한 자료의 축적은 정보시스템의 발전뿐 만 아니라, 조직의 발전 및 서비스의 향상을 위해서 반드시 필요하다. 지금까지 분석된 결함 자료는 금융회사뿐만 아니라, 관련 기관이나 제조사, 그리고 외주용역회사에도 매우 중요한데 이는 각 기관들이 담당하는 분야에서 결함의 특징 및 추이를 알 수 있고, 그에 따라 대비할 수 있기 때문이다. 물론 이 자료는 다른 도메인에도 활용 가능하며 도메인에 맞게 별도의 분석이 가능할 것이다. 이러한 과거 결함 데이터의 분석을 통하여 업계 전체뿐 만 아니라 개별 금융회사에서 활용할 수 있는 정보시스템 위험예보 모델 구축이 가능하다. 이러한 예보가 필요한 이유는 첫째, 금융업계는 제도 및 규정의 변화가 많다. 또한 각 시스템들

이 긴밀하게 연관되어 한 부분의 변화가 다른 시스템에 미치는 영향이 크다. 하지만 어느 부분이 어떻게 변경되는 지에 대한 정보는 거의 없어 그에 대한 대비가 불가능하다. 둘째는 Trading 환경의 변화이다. 금융업계는 이론적으로나 실제적으로 완벽한 시장에 의하여 운영된다. 이러한 시장에서 Algorithm Trading, High Frequency Trading, Low Latency 등 IT 기술에 의존하여 투자 수익을 창출하고자 하는 노력들이 맹렬하다. 하지만 이러한 기술들은 일반 투자자들은 잘 알 수 없을 뿐 만 아니라, 매우 큰 규모의 거래로 인하여 정보시스템의 안정성에 큰 영향을 줄 수 있다. 따라서 예보를 통하여 거래의 현황을 파악하고 대비하는 것이 필요하다. 셋째는 핵심시스템을 위험으로부터 사전에 보호할 수 있다. 예를 들어 계정계 시스템과 부속 시스템들은 Trading의 핵심 시스템이다. 이 시스템에 대하여 변경이나 이전, 또는 신규업무의 추가 등이 동시에 발생할 경우, 예보시스템의 예보를 근거로 하여 이를 제어함으로써 위험을 관리할 수 있다. 넷째는 미리 예보가 되므로 위험발생에 미리 대비함으로써 이미 발생한 결함이라 하더라도 신속한 조치가 가능하다. 정보시스템의 결함 예측에 있어서 중요한 것은 각 결함의 요인이 되는 요소들에 대한 예측지표이다. 결함예측지표는 많으면 많을수록 결함을 예측하는 모델의 정확도는 더 높아질 수 있어, 업계의 장애예보 모델 구축 가능성은 높아질 것이다. 이러한 모델이 가능하기 위해서는 업계 전체를 아우를 수 있는 통합장애관리체제가 필요하다. 금융업계는 각 금융회사의 정보시스템과 부속장비들이 한국거래소를 중심으로 긴밀하게 연관되어 있다. 그러므로 한 정보시스템의 장애는 다른 정보시스템의 장애를 유발하여 업계 전체로 확산될 수 있으므로, 서비스에 영향을 주는 정보시스템의 위험을 최소화하고 개별 정보시스템의 결함으로 인한 서비스의 중단을 막기 위해서 업계 통합장애예보시스템이 필요하다. 금융정보시스템에 대한 위험예보시스템을 위해서는 추가로 환경적 요소에 대한 예측변수들과, 소프트웨어의 결함을 유발하는 요소들에 대한 예측변수들에 대한 연구가 더 진행되어야 한다. 그리고 이러한 예측변수들을 결합한 예측변수들의 타당성 연구도 진행되어야 한다. 향후 이러한 연구들이 진행된다면 위험을 미리 인지하여 관리함으로써 금융정보시스템의 안정적 운영에 크게 기여할 것이다.

참 고 문 헌

국무조정실, 정보시스템 운영관리 지침, 정보통신부, 2005.

김명식, ITIL v3 Foundation 수험서 및 ITSM 컨설턴트 핸드북, 도서출판 아진, 2009.

김법진, 한인구, 이상재, “CRAMM을 이용한 정보시스템 위험관리: 신용카드회사 사례연구”, 경영정보학연구, 제10권, 제2호, 2000, pp. 149-176.

김영걸, 이종만, 이재남, “정보시스템의 위험도 분석에 관한 연구: 통합적인 분석 틀을 중심으로”, 경영정보학연구, 제8권, 제2호, 1998, pp. 37-51.

박성운, 정재민, 박수용, “반응적 에이전트 프레임워크를 위한 패턴언어”, 정보과학회논문지, 소프트웨어 및 응용, 제31권, 제3호, 2004, pp. 255-386.

배재권, 김진화, 김동욱, “데이터마이닝을 이용한 정보시스템 개발 프로젝트 리스크 예측에 관한 연구”, 한국경영정보학회, 추계통합학술대회, 2011, pp. 474-480.

서한준, ITIL 기반의 IT 서비스 관리, Nemo Books, 2006.

윤재수, 주식투자 무작정 따라하기, 길벗, 2011.

이영재, 황영수, “IT운영리스크 최소화를 위한 피해저감모델 구현에 관한 연구”, Journal of Information Technology Applications and Management”, 제14권, 제3호, 2007, pp. 95-113.

이영희 역, 시스템 장애는 왜 두 번 일어났을까, 한빛미디어, 2012.

이철환, 숫자로 보는 한국의 자본시장, 브레인스토어, 2011.

정보통신산업진흥원, Software Engineering, White Book, 정보통신산업진흥원, 2011.

정철용, "국내 은행금융기관의 통합 위험관리시스템 개발에 대한 연구: 객체지향적 접근", Information Systems Review, 제4권, 제2호, 2002, pp. 361-376.

조숙진, 이석준, 함유근, "정보시스템 프로젝트의 위험요인에 관한 실증 연구", 경영정보학연구, 제16권, 제3호, 2006, pp. 144-158.

한인구, 전영승, 김은홍, "우리나라 회계정보시스템의 현황 및 개선방안", 경영정보학연구, 제3권, 제2호, 1993, pp. 93-116.

CAIS, "Information System Risks and Risk Factors: Are they mostly about Information Systems?", *Communications of the Association for Information Systems*, Vol.14, 2004.

Graves, T., A. F. Karr, J. S. Marron, and H. Siy, "Predicting Fault Incident Using Software change history", *IEEE Transaction on Software Engineering*, Vol.26, 2000, pp. 653-661.

Grottke, M. and K. Dussa-Zieger:, "Prediction of Software Failures Based on Systematic Testing", *Electronic Proc. 9th European Conference on Software Testing Analysis and Review*, (EuroSTAR), Stockholm, 2001.

Hoffmann, G. A., F. Salfner, and M. Malek, *Advanced Failure Prediction in Complex Software Systems*, Proc. of SRDS, 2004.

http://www.nocutnews.co.kr/show.asp?idx = 1101908.

http://www.zdnet.co.kr/news/news_view.asp?artice_id = 20120523230818&type = det.

http://www.zdnet.co.kr/news/news_view.asp?artice_id = 20120606182607&type = det.

IEEE, IEEE Std 1044, Standard Classification for SW Anomalies, IEEE Society, 2009.

ISO/IEC 9126-1: 2001 "Software Engineering-Product Quality-Part1: Quality model", 2001.

Padayachee, I., P. Kotze, and A. van Der Merwe, "ISO 9126 external systems quality characteristics, sub-characteristics and domain specific criteria for evaluating e-Learning systems", *Proceedings of South African Computer Lecturers, Association* (SACLA), ACM, 2010.

Stefan Lessmann, "Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings", *IEEE Transaction on Software Engineering*, Vol.34, No.4, 2008, pp. 485-496.

Stoneburner, G., A. Goguen, and A. Feringa, *Risk management guide for information technology systems: Recommendations of the national institute of standards and technology*, National Institute of Standards and Technology (NIST) Special Publication 800-30, U.S. Government Printing Office, 2001.

Zhou, Y., "Empirical Analysis of Object-Oriented Design Metrics for Predicting High and Low Severity Faults", *IEEE Transactions on Software Engineering*, Vol.32, No.10, 2006, pp. 771-789.

Zimmermann, T. N. Nagappan, and A. Zeller, Predicting Bugs from History, Springer, 2008.

Information Systems Review
Volume 14 Number 2
August 2012

A Proposal for Improvement and Current Situation of Risk Management on Financial Information System

TaeHong Kang* · SungYul Rhew**

Abstract

Improvement of the capability to cope with risk based on prior preventive management is very important for efficient operation of financial information system. In order to do this, understanding, analysis, and countermeasures for the risk that happened already in the past is essential. In this study, the defect data which happened in the financial information system including account system, business system and data feeding system during 4 years and 5 months were categorized and analyzed by the domain, defect factors, period, day of the week, phases of software development, and defect cause. As a result, it was identified that the defect data had characteristics and trends along the phase of software development, day of the week, and the cause, also that building risk prediction model was necessary for the risk management of whole financial domain due to the relation of the information systems.

Keywords: Financial Information System, Risk Management, Predictive Risk Model

* Graduated student, School of Computer Science and Engineering, Soongsil University
** Professor, School of Computer Science and Engineering, Soongsil University

◐ 저 자 소 개 ◑

강 태 홍 (thkang317@gmail.com)

숭실대학교 전자계산학과를 졸업하고 석사학위를 취득하였으며 현재 박사과정을 수료하였다. 주요 관심분야는 정보시스템의 장애예측이나 장애예보모델구축 등이다.

류 성 열 (syrhew@ssu.ac.kr)

현재 숭실대학교 컴퓨터학부 교수로 재직 중이며 숭실대 전자계산원 원장 및 중앙전자계산소 소장, 숭실대 정보과학대학원장 등을 역임하였다. 관심분야로는 소프트웨어 요구공학, 소프트웨어 유지보수, 오픈소스 소프트웨어 등이다.

논문접수일 : 2012년 06월 26일 **게재확정일 : 2012년 08월 17일**
1차 수정일 : 2012년 08월 01일