

기술분류체계 기반의 장애 점검포인트와 이벤트 룰엔진을 적용한 무장애체계 구현

Fault-Free Process for IT System with TRM(Technical Reference Model) based Fault Check Point and Event Rule Engine

현 병 탁 (Byeong Tag Hyun) LG CNS
김 태 우 (Tae Woo Kim) LG CNS
엄 창 섭 (Chang Sup Um) LG CNS
서 종 현 (Jong Hyen Seo) 한국산업기술대학교 e-비즈니스학과, 교신저자

요 약

글로벌 싱글 인스턴스(GSI) 기반의 기업의 정보시스템은 기업 내부정보 및 자원/자산을 통합관리하고 프로세스의 동질성을 확보하여 업무의 효율성 및 전사적인 생산성을 향상시키고 있지만 정보시스템의 장애가 발생하게 되면 비즈니스가 마비될 수 있고 그로 인해 엄청난 금전적 손실이 따르는 위험을 동시에 안고 있다. 여분의 부품 사용을 전제하는 수많은 결함허용(Fault-Tolerance) 기법들이 안정적인 정보시스템을 위해 연구되어 왔다. 결함허용 기법의 설계 및 적용의 어려운 점은 결함의 유형 및 빈도를 정보시스템을 운영하기 전에는 알기 어렵다는 것이다. 따라서 정보시스템 구축 단계에서의 결함허용 기법의 적용과 함께, 구축 후, 운영적인 측면에서의 장애관리 기법을 동시에 고려하여 안정적으로 정보시스템을 운영하는 것이 매우 중요하다.

본 연구에서는 기술분류체계(TRM)기반의 점검포인트와 이벤트 룰엔진 적용으로 시스템 장애를 사전예측 하는 방법을 포함하는 정보시스템 장애관리 기법인 무장애체계 프로세스를 제시 하였다. 또한, 무장애체계 방법을 적용한 장애관리 정보시스템(PICS)을 하이테크 장치산업의 대표기업에 구축하여 무장애체계 방법 적용 전, 후의 효과도 함께 제시 하였다. 무장애체계 도입 후 월별 장애건수, 장애시간은 각각 46%, 56% 감소하였으며, 장애로 인한 매출에 대한 기회 손실금액은 77% 절감하는 효과를 보였다.

키워드 : 기술분류체계, 룰 엔진, 무장애체계, 장애관리

I. 서 론

최근 몇 년 간 전기, 전자 산업계의 최대 화두는 세계 각지와 본사를 하나의 프로세스와 정보시스템으로 통합, 표준화하는 글로벌싱글인스턴

스(GSI)였다. 세계 시장을 무대로 하는 글로벌 기업들은 이러한 GSI 개념의 전사적자원관리(ERP), 생산실행시스템(MES), 공급망관리(SCM), 제품수명주기관리(PLM)를 가동하고 있거나, 구축 진행 중에 있다.

GSI를 통해 기업들은 프로세스와 시스템을 하나로 통합하여 기업 내부정보 및 자원/자산을 통합 관리함으로써, 프로세스의 동질성을 확보하고 데이터의 표준화된 관리를 통해 업무의 효율성 및 전사적인 생산성을 향상시킨 것으로 보고되고 있다(이치현, 강승원, 서종현, 2007).

하지만, 글로벌싱글인스턴스(GSI) 환경이 유지보수 비용을 절감하고 업무 효율성을 향상시킬 수 있다는 사례들이 보고되고 있지만 우리가 간과할 수 없는 커다란 위험도 존재한다. 즉 GSI 환경의 가장 커다란 위험 요소는 바로 시스템 장애로 인해 전 세계적으로 비즈니스가 마비될 수 있다는 점이다(백운기, 2010).

컨틴전시 리서치 플래닝(Contingency Planning Research)의 조사에 의하면 정보시스템 장애 시 손실비용을 업종별로 구분하면, 증권 거래 업무의 경우 시간당 645만 달러, 신용카드/판매의 경우 260만 달러라는 엄청난 손실을 입게 된다고 한다. 또한 미국의 연 매출 2억 달러 이상인 제조/유통 회사를 대상으로 한 조사에 의하면 장애 발생 후 4일 이내에는 하루에 평균 10만 달러, 10일 이내에는 1만 달러의 손실을 보며, 10일 이상으로 넘어갈 경우에는 50%의 기업이 파산에 이르게 된다고 보고되고 있다.

1.1 연구 목적

ERP, MES와 같이 기업의 비즈니스와 직결되고 고 신뢰도가 요구되는 정보시스템들에 대해서는 결함을 없애려는 노력이 오랜 기간 수행되어 왔다. 결함허용(Fault-Tolerance) 기법은 하드웨어 오동작, 소프트웨어 에러, 정보의 오염 등의 장애가 발생하더라도 사용자가 이를 인지하지 못하도록 하고, 주어진 임무를 올바르게 수행하는 기술적 장애관리방법으로, Triple Modular Redundancy 기법 등의 하드웨어 결함허용, Parity 기법 등의 정보 결함허용, Check Pointing 기법 등의 소프트웨어 결함허용 방법이 영역별로 연

구되어 왔다. 이와 같은 결함허용 기법은 개별 정보시스템 구축 시에 기본적으로 여분의 부품의 사용에 근거하고 있다(Johnson, 1989). 각 영역별 대표적인 결함허용 기법은 다음 장에서 소개하고자 한다. 여분의 부품을 사용하여 개별 정보시스템에 결함허용 기법들은 그 개념은 단순하지만, 실제 정보 시스템의 설계 및 적용은 간단하지 않다. 그 이유는 정보 시스템을 동작해 보기 이전에는 결함의 유형 및 발생빈도를 알기 어렵기 때문이다(김문희, 1993). 따라서 정보시스템 구축 단계에서의 결함허용 기법의 적용과 함께, 구축 후, 운영적 측면에서의 장애관리 기법을 동시에 고려하여 안정적으로 정보시스템을 운영하는 것이 매우 중요하다(조성민, 한혁수, 2007). 최근의 기업들은 수많은 정보 시스템이 서로 신경세포처럼 연결하여 운영하고 있으며, 하나의 시스템에서 발생한 아주 사소한 결함이 다른 여타의 시스템에서는 치명적인 장애로 이어질 가능성도 존재한다. 따라서 수많은 정보 시스템을 통합적으로 모니터링하여 개별 정보시스템에 설계 및 적용된 결함허용기법에서 처리되지 못하는 예측하지 못한 결함에 대해서도 조치하여 장애로 이어지지 않도록 하는 통합 모니터링 시스템이 요구되고 있다.

운영 단계에서는 정보시스템 구축 후 모니터링 시스템을 구축하여 장애가 발생하면 담당자가 인지하고 원인분석을 하여 신속하고 정확하게 조치를 취하는 장애관리 프로세스를 수립하고 운영하는 것이 매우 중요하다. 안정적 운영을 위한 시스템 구축과 운영단계 두 가지 영역 모두 결함이 발생한 후의 영향을 최소화 하는 방법이라 할 수 있다. 시스템 구축단계에서 요구되는 기술적 안정화 방안인 결함허용에 대해서는 많은 연구가 진행 되었지만 정보시스템의 안정적 운영을 위한 장애관리 프로세스 및 방법에 관한 연구는 상대적으로 부족한 실정이다. 본 연구에서는 기존의 운영적 측면에서의 정보시스템의 장애관리 방법을 살펴보고, 기술분류체계

기반 점검포인트와 이벤트 타입 롤엔진 기법, 기존의 장애관리 프로세스를 개선 보완한 무장애체계 프로세스를 통합한 무장애체계 방법을 제안하고 각 프로세스 구성요소들의 차별점을 설명한다. 제안한 무장애체계 방법의 효용성을 검증하기 위하여 글로벌 하이테크 장치산업의 A 기업에 적용하여 그 결과를 확인 하였다. 무장애체계 용어는 정보시스템이 구축되어 운영 중에 장애발생이 예상되는 모든 인자들을 찾아내어 사전에 조치하여 정보시스템에 심각한 영향이 없도록 관리하는 것으로 정의 하였다.

1.2 결함허용 관련 연구

기존의 결함허용 기법은 하드웨어, 정보, 소프트웨어 영역별로 수많은 기법들이 각 구성요소별로 제안되어 왔다. 모든 기법들이 근본적으로 여분의 부품의 사용에 근거를 두고 있으며, 대표적인 연구를 간단히 살펴보면 다음과 같다.

- 하드웨어 결함허용 기법
 - Triple Modular Redundancy

TMR은 결함차단(Fault Masking)에 기초를 둔 기법으로 기본개념은 하드웨어를 삼 중 복하여 그 출력을 비교하여 2개 이상의 같은 출력을 TMR의 출력으로 선택하는 것이다.
 - Duplication with Comparison

하드웨어를 이 중복하여 그 출력을 비교함으로써 결함을 감지하는 기법이다. 단지 결함 감지의 기능만 제공한다.
 - Stand-by Sparing

동작중인 하드웨어에서 결함이 감지 되었을 때 대기중인 여분의 하드웨어로 대체하는 기법으로 결함을 감지할 수 있는 기법과 더불어 사용되어야 한다.
- 정보 결함허용 기법
 - Parity Code

정보가 2진수로 표시될 때 한 정보에 1의 개수가 홀수개 또는 짝수개 인지를 알리는 여분의 정보를 추가한다. 정보결함 감지에 사용되며, 결함이 짝수개 발생하면 감지할 수 없는 단점이 있다.

- m of n code

정보의 길이가 n bit일때 1의 개수가 반드시 한 정보에 m개가 되도록 코드화 한 것으로 1의 개수가 m개가 아닐 때 결함이 있음을 감지하는 기법이다.
- 소프트웨어 결함허용 기법
 - Check Pointing

소프트웨어 실행 중에 검사 시점을 설정하여 오류가 발생하는지를 검사하여 이상이 없으면 계속 수행하고 이상이 감지되면 그 이전의 검사시점으로 되돌아가 재수행 하는 방식이다(Johnson, 1989).
 - Recovery Block

재수행(Rollback and Retry)에 근거한 기법으로 검사시점에서 오류가 감지되면 지정된 이전 시점으로 되돌아가 같은 기능을 가진 다른 소프트웨어 모듈을 실행하는 방식으로 단일 프로세스 내에 적용될 수 있다(Randell, 1975).
 - Distributed Recovery Block

Recovery Block을 분산 환경으로 확장 적용하여 하드웨어 결함과 소프트웨어 결함을 동일한 방법으로 극복할 수 있도록 한 기법이다(Kim, 1992).
 - N Self Checking Programming

두 개 이상의 자가진단(Self -Checking) 모듈이 실행되면서 그 중의 하나가 주어진 기능을 수행하고 나머지는 여분으로 대기 상태에 있다가 자가진단에 의해 수행 부품에 결함이 발견되면 대기 부품중의 하나가 새로운 수행 부품이 되어 주어진 기능을 수행하는 기법이다(Laprie et al., 1990).

1.3 정보시스템 운영측면 장애관리 관련 연구

기업의 정보시스템이 증가하고 복잡해짐에 따라, 개별 정보시스템에 설계 및 적용된 결합허용 기법에서 처리되지 못하는 예측하지 못한 결합에 대해서도 조치하여 장애로 이어지지 않도록 하는 통합 모니터링 시스템이 요구되고 있다. 기존의 운영측면의 장애관리를 위해 모니터링 중심의 연구가 진행 되어왔다. 공장 자동화 영역에 특화된 실시간 모니터링 기능, 성능 튜닝 및 분석기능, 문제 조치를 위한 SQL추적 방법이 포함된 모니터링 툴을 이용한 장애검출방안은 그 한 예라 할 수 있다(홍지윤, 조광문, 2003). 또한, 실시간 모니터링 기술을 이용한 웹서버 장애관리 시스템을 제안하여 웹서버를 실시간으로 모니터링하고 장애발생시 장애정보를 저장하여 담당자에게 통보하는 기능을 비교적 상세하게 설명한 연구도 발표 되었다(김승남, 박민현, 한옥표, 정연중, 2006). 쉬운 GUI를 통해 전문지식이 없는 일반인도 효율적으로 관리가 가능한 점과 SMS 서비스 등일 가능성을 강조 하였다. 하지만, 기존의 모니터링 기능을 강조한 장애관리 연구들은 정보시스템의 DB의 한정된 영역만을 모니터링 하여 장애에 대한 조치를 취하거나 전체 장애관리 프로세스에 대해서는 개념적인 내용만을 제시하는 한계를 나타내었다(정아주, 박준상, 이상우, 김명섭, 2009). 웹서버 장애관리 시스템에서는 전체 시스템의 이상 징후를 감지하는 것은 가능하지만, 감지 후 효율적인 조치방안에 대해서는 제시하지 못하였다(김승남, 박민현, 한옥표, 정연중, 2006).

최근에는 정보시스템의 가동율에 직접적인 영향을 미치는 정보시스템의 장애현황을 분석함으로써 IT서비스 관리(ITSM) 프레임워크를 기반으로 한 프로세스의 도입 및 내재화에 대한 효율성을 입증한 연구가 발표 되었다. 14개 기업의 정보시스템의 장애 현황 데이터를 분석하여 장애관리

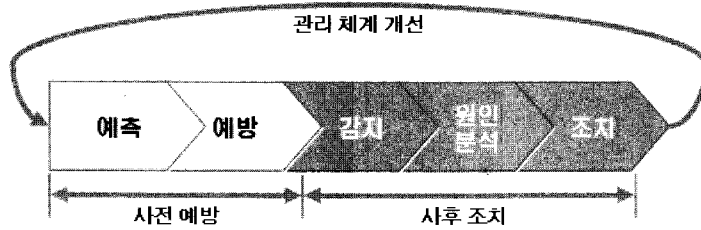
프로세스를 적용한 기업이 그렇지 않은 기업보다 장애 발생건수 및 발생시간이 낮다는 내용을 통계적 분석을 통해 제시 하였으며, 통제가 불가능할 것 같은 기술적 요인에 의한 장애율도 상당부분 낮출 수 있음을 확인 하였다. 장애관리 프로세스를 인적 요인, 프로세스적 요인, 기술적 요인으로 구분하여 제시하였지만, 실제 기업에 적용하여 검증해야 하는 기술적 요인의 구체적 장애조치 및 예방 방법에 대한 제시는 미흡하였다(박선철, 이원영, 2008).

다음 장에서는 무장애체계 프로세스를 정의하고, 장애 발생 후 처리 리드타임을 최소화 하는 기술분류체계기반 점검포인트를 이용한 장애 처리 프로세스와 이벤트 톨엔진을 적용해 장애 발생 인자를 사전에 차단하여 장애 사전 예방이 가능한 프로세스를 설명한다. 마지막으로, 무장애체계를 정보시스템으로 구성하는 방법에 대하여도 함께 제시한다.

II. 방 법

2.1 무장애체계 프로세스

무장애체계 프로세스는 <그림 1>과 같이 세 가지 영역으로 구분 할 수 있다. 첫 번째는 장애 인자 사전 차단을 통해 장애를 예방하는 사전 예방 영역, 두 번째는 장애 발생 후 조치를 취하는 사후 장애 조치 영역, 마지막으로 장애 처리 후 재발을 막기 위한 장애관리 체계개선 영역이다. 이 세 가지 영역은 하나의 사이클(Cycle)로 운영되어야 한다. 우선 기업의 기간 정보시스템으로부터 나오는 이벤트의 감지 및 분석을 통하여 사전예측 및 예방을 지속적으로 수행하여 장애의 발생을 원천적으로 차단한다. 하지만, 현실적으로 사전예방이 100% 불가하기 때문에 발생된 장애에 대해서는 즉각적이고 정확한 조치가 필요하다. 장애가 발생하게 되면 시스템 혹은 담당자가 장애를 감지하고 접수 하게 된다. 접수된



프로세스	내용
장애 예측	장애관리 대상 및 이벤트에 대한 모니터링 및 추이를 확인하고, 이상 징후를 분석 및 검증하는 프로세스
장애 예방	장애 사전 감지 모니터링 및 시스템 일일 점검을 통해 장애 발생 요소를 제거하여 장애를 사전에 예방하는 프로세스
장애 감지	장애가 발생한 경우, 장애관리 대상과 현상을 인식/식별하는 프로세스
장애 원인분석	발생된 장애에 대해 1선에서 신속한 전파 및 원인 분석을 수행하여, 조치 가능 시 복구 작업을 수행하고, 불가능 시 2/3선으로 Escalation 하여 상세 분석을 통해 복구를 수행하는 프로세스
장애 조치	
장애 관리 체계 개선	완료된 경우 확산 방지를 위해 상세원인 분석/검증으로 장애관리 체계를 개선하는 프로세스

〈그림 1〉 무장애체계 프로세스

장애는 전문가의 원인분석을 거친 후 장애 등급에 따라 1선 조치 또는 2, 3선으로 Escalation의 조치를 취하게 된다. 조치 완료 후에는 재발 방지를 위하여 장애의 상세 원인분석/검증을 통해 장애관리 체계를 개선한다. 개선된 장애관리 체계는 다시 예측 및 예방 능력을 향상 시키는 순환 사이클이 완성된다.

2.2 장애 등급 및 시스템 관리등급

정보시스템의 장애 정도에 따라 조치를 위한 투입인력, 가용시간, 비용 등의 처리규모가 다르므로, 정보시스템의 운영 중요도와 장애의 영향도를 관리 하는 것은 매우 중요하다. 무장애체계에서는 장애관리의 효율을 높이기 위해 정보시스템 별로 시스템운영의 중요도를 산정하여 시스템 관리등급을 정의하고, 발생한 장애가 기업에 미치는 영향을 장애등급으로 역시 정의하여 관리한다. <표 1>은 무장애체계의 장애등급을 나타낸다. A등급은 각 개별 공장의 생산이 100% 중

단된 경우를 생산 전면 중단으로 정의하였다. B 등급은 생산의 부분 중단과, 생산 외 중요업무가 중단된 경우로 정의 하였다. 생산 외 중요업무란 장애발생시 업무 연속성 유지를 위해 수행이 필요한 최소한의 업무를 뜻한다. 시스템 관리등급은 <표 2>와 같이 등급별로 정보시스템 가용성 목표와 장애 발생시 복구목표를 정리하여 관리한다.

2.3 기술분류체계 기반 점검포인트 도출

사전 예방에서 근본적으로 차단하지 못한 위험 이벤트는 장애로 이어지게 된다. 사후 장애 조치 단계에서는 장애가 발생하더라도 정보시스템 사용자가 이를 인지하기 전에 조치가 이루어져야 하며, 기업의 입장에서는 장애로 인한 시간적, 금전적 손실을 최소화 해야 한다.

그러므로 장애발생, 장애감지, 장애원인분석, 장애조치의 Process로 이루어지는 사후 장애 조치는 발생부터 조치까지의 리드타임을 단축하는

〈표 1〉 무장애 체계 장애 등급

장애등급	등급 기준	비고
A등급	생산 전면 중단	생산 전면 중단의 단위는 각 공장임, 공장 생산이 100% 중단된 경우를 '생산 전면 중단'으로 정의함
B등급	생산 부분 중단 생산 외 중요업무 중단	공장내 일부공정(1~99%) 중단될 경우를 '생산 부분중단'으로 정의함. 단, 생산 부분중단 장애가 조치 지연되어 생산 전면중단으로 확대되는 경우 A등급 장애로 판단함.
C등급	A등급과 B등급의 장애를 제외한 생산에 영향이 없는 장애	

〈표 2〉 서비스 관리 등급

서비스 등급		1등급 (Vital)	2등급 (Critical)	3등급 (Important)	4등급 (Deferrable)
업무 등급 분류	업무 중요도	정보나 활동의 손실은 생산활동 불가능을 포함, 회사수익을 위한 주요 사업 기능에 치명적이고 막대한 손실을 발생	정보나 활동의 손실은 외부사업 기능 지연 발생 및 전사 임직원의 시스템 사용 및 접근 제한으로 전 사적인 업무 수행 지연 발생	정보나 활동의 손실은 내부 사업 기능에 불리하게 작용을 할 것이다. 이 정보의 손실은 회사 내부 사업 기능 처리 지연 발생	정보나 활동의 손실은 사업 지원 기능에 불리하게 작용할 것이다. 이 정보의 손실은 일반 업무에 있어서 사업 지원 지연 발생
	업무 유형	◦ 생산에 직접 영향	◦ 대외 서비스 지원 ◦ 전사 업무 직접 영향 ◦ Cash Flow 직접 영향 ◦ 법적 제재 사항 ◦ 1등급 시스템에 연계 정보를 제공함으로써 생산에 간접적인 영향	◦ 단위 지원시스템	◦ 지원시스템
가용성 목표	서비스 가동률	99.99966%	99.995%	99.990%	99.950%
	년간 계획되지 않은 중단 시간	< 1.7분	< 26분	< 52분	< 262분
장애 복구 목표	목표 복구 시점 (RPO)	≈ 0시간	≤ 8시간	≤ 24시간	≤ 1주
	목표 복구 시간 (RTO)	≤ 15분	≤ 1시간	≤ 24시간	≤ 24시간

것이 매우 중요하다.

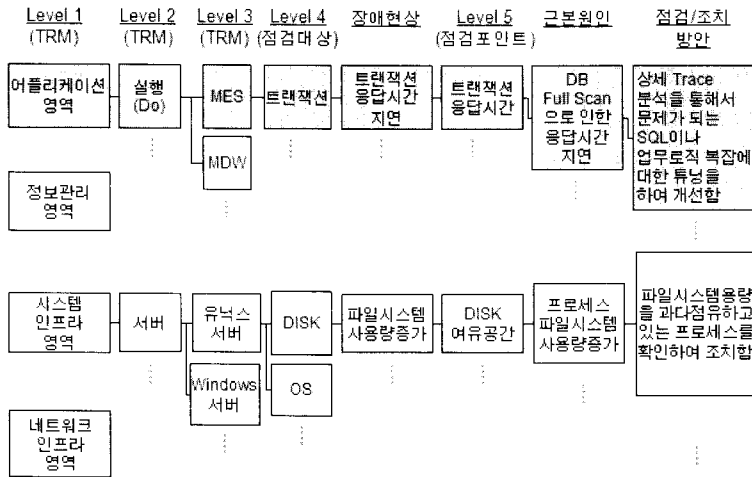
기존의 장애조치 방법은 담당자가 육안으로 장애를 감지하거나, 사용자로부터 유선으로 장애사실을 통보 받는다. 담당자는 인지한 장애에 대하여 기존 처리내역을 수작업으로 검토하거나, 전임자, 동료, 시스템 벤더 문의를 통해 원인을 파악 하였다. 또한 조치를 위해서는 매뉴얼을 참조 하거나, 매뉴얼에 없는 상황에 대하여는 경험

자 또는 전문가 자문을 통해 조치를 하게 된다.

기존의 방법은 장애 발생에서 조치까지 걸리는 시간이 장애의 난이도, 담당자에 따라 수시간에서 수일까지 걸릴 수 있다. 하이테크 장치산업의 대표기업인 A사를 사례로 기술분류체계 기반의 점검포인트를 도출하였다.

장애 점검포인트를 누락 없이 모두 도출하기 위하여 A사의 기술분류체계(Technical Reference

▪ 장애점검포인트 정의



<그림 2> 장애 점검 포인트 예시

<표 3> 기 발생장애 유형 분석 예시

장애 유형	상세원인
Application 문제로 인한 서비스 중단	TIB/RV 처리 한계에 대한 Bug 존재
	NanoTrack 재 기동 테스트 환경 부족
	타 사례 공유 및 개선작업 미흡
	프로그램 인스펙션 및 점검 테스트(회귀) 미흡
	운영과 동일한 환경의 실전 테스트 부족
	변경 영향분석 및 인스펙션부족
	신종 바이러스로 Application 수행 지연(CPU 부하 발생)
	스트레스 테스트가 미흡하다
	PM 및 패치등 사전 예방 활동 부족
	변경 영향분석 및 개선활동 미흡
서버 및 주변장치의 장애요인들이 존재	변경관리 프로세스 미준수
	RVRD 데몬 Hang 으로 공정 진행 중단
	H/W Fail 장애가 자주발생
	H/W Firmware Bug
	S/W결함 Bug 존재
Database 성능저하 및 결함	스크립트 적용시 사전 검증 절차 미비
	H/W 불량 및 노후화(EOS)로 인한 취약점이 내포되어 있다.
	배전반 전원 이중화 설계 구조의 오류
	DB OBJECT 최적화된 관리체계 미흡
	파일 시스템 삭제로 Database Down
최적화되지 않은 SQL의 튜닝 활동 미비	
ORACLE 제품 자체 결함에 의한 Bug 존재	

Model)를 준용 하여 <그림 2>와 같이 레벨 1~레벨 3의 구조를 정의 하였다. 레벨 4는 장애관리를 위한 점검대상으로 정보통신산업진흥원의 '정보시스템 장애처리 지침', IT 전문자료 등의 사외자료와 A사의 정보시스템 운영지수 자료를 통해 정리 하였다. A사의 정보시스템 운영지수는 정보시스템 운영을 통해 축적된 장애 발생 가능성이 높은 대상을 정리한 자료이다. 점검대상 별로는 A사의 기 장애발생 및 조치 이력 3년치를 분석한 결과와 시스템 전문가에 의해 레벨 5의 점검포인트를 정의하고 그에 대한 근본원인과 점검 및 조치 방안을 정리하여 데이터베이스화 하였다. <표 3>은 A사의 3년 간의 기 발생한 장애에 대해 유형 및 원인을 분석한 예시이다.

2.4 룰엔진 적용을 통한 사전 장애 예방

기업의 정보시스템 장애관리에 있어서 심각하면서도 현실적인 문제는 이벤트가 적절하게 관리되지 못한다는 것이다. 즉, 관리대상이 되는 정보시스템에서 장애로 이어질 수 있는 수많은 이벤트들을 관리자는 직관적으로 판단하고 대응하는데 어려움이 있다. 폭발적으로 증가하는 이벤트들 중에서 장애로 이어질 수 있는 유의미한 정보를 찾아 관리자가 신속하고 정확하게 조치를 취하도록 수많은 이벤트들을 쉽게 분석할 수 있는 정보로 가공하여 제공해 주는 것이 장애

사전 예방의 핵심 포인트이다. 본 연구에서는 이를 위해 이벤트의 패턴 분석을 통한 룰엔진(Rule engine) 기법을 사용한 장애 사전 예방 방법을 제시 하였다.

2.4.1 이벤트 타입 정의를 통한 룰엔진

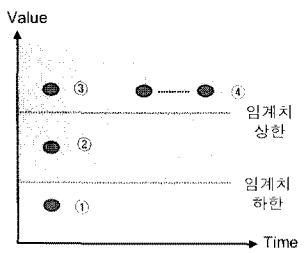
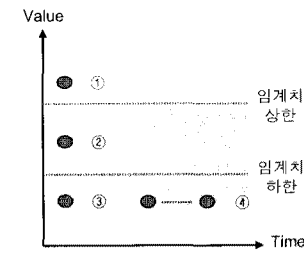
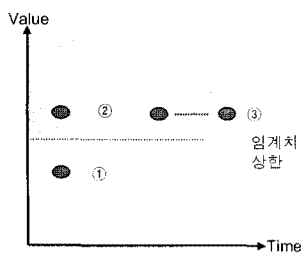
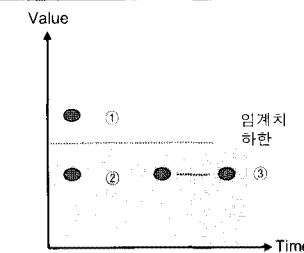
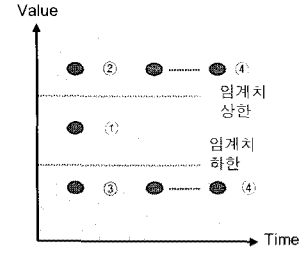
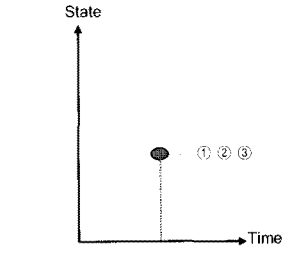
정보시스템에서는 순간적으로 수십, 수백개의 이벤트가 발생하게 된다. 순간적으로 발생한 수백개의 이벤트를 하나하나 분석하여 장애원인을 분석하는 것은 대단히 어려운 일이다. 또한, 이와 같은 장애조치는 신속히 이루어져야 하므로, 시스템 관리자는 빠르게 이벤트 발생 원인 및 장애로 이어질 가능성에 대한 판단이 필요하다. 따라서, 자동화된 모듈이 요구되며 자동화 모듈의 핵심기능이 바로 이벤트 패턴 분석을 통한 룰엔진 구성 및 적용이다. 룰엔진을 적용함으로써 시스템 관리자는 관리해야 할 유효한 이벤트만을 감시하고 조치함으로써 신속 정확하게 장애에 대처할 수 있다. 본 연구에서는 414개 장애 점검 포인트에 대하여 6가지의 기본 이벤트 타입을 정의하여 룰엔진을 구성 하였다.

각 이벤트 타입은 시간과 측정값을 축으로 하여 4단계의 이벤트 등급을 정의 하였고 이벤트 등급별로는 조치를 취하기 위한 이관 프로세스를 지정 하였다. 4가지 이벤트 등급에 정의 및 이관 프로세스를 <표 4>에서 설명 하였다.

<그림 3>은 무장애체계에서 관리하는 6가지

<표 4> 이벤트 등급 정의

Code	Rule	Description	이관프로세스
Down(Red)	생산관련 AP, Process Down 서버운영등급 1, 2, 3등급 장비 Down	한곳 이상의 사업장 또는 한개 업무 단위 이상이 서비스 불가한 긴급 상황	장애감지
		이중화된 장비중 하나의 장비 중단으로 서비스 영향 없음	장애예방
Critical(Orange)	(Event Type별 정의 참고)	일부 업무 단위가 서비스 불가 또는 급격한 성능 저하	장애예방
Warning(Yellow)	(Event Type별 정의 참고)	한개 업무 단위 이상의 성능 저하	장애예방
Normal(Green)	일반적 Event	정상적인 상태	-

<p style="text-align: center;">Type 1</p> 	<p>Memory, CPU와 같이 임계치의 상한, 하한 사이에서 자원을 관리해야 하는 Type</p>	<p style="text-align: center;">Type 2</p> 	<p>가용성과 같이 임계치의 하한, 상한 사이에서 자원을 관리해야 하는 Type</p>
<p style="text-align: center;">Type 3</p> 	<p>화재 Sensor와 같이 임계치를 초과하지 않도록 자원을 관리해야 하는 Type</p>	<p style="text-align: center;">Type 4</p> 	<p>프로세스 구동 상태와 같이 임계치 미만으로 자원을 관리해야 하는 Type</p>
<p style="text-align: center;">Type 5</p> 	<p>전압과 같이 임계치 상/하한내에서 자원을 관리해야 하는 Type</p>	<p style="text-align: center;">Type 6</p> 	<p>System Trap와 같이 임계치 없이 자원내에서 Event가 발생한 경우</p>

주) ① Normal, ② Warning, ③ Critical, ④ Down.

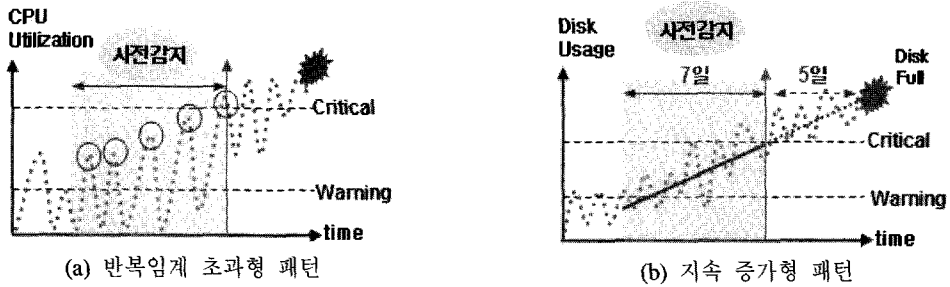
<그림 3> 이벤트 타입 정의

이벤트 타입을 설명하였다. 타입 1은 메모리, CPU와 같이 임계치의 상한, 하한 사이에서 자원을 관리 해야 하는 타입으로 크리티컬, 다운 등급이 임계치 상한에 위치한다. 타입 2는 가용성과 같이 임계치의 하한, 상한 사이에서 자원을 관리해야 하는 타입으로 크리티컬, 다운 등급이 임계치 하한 영역에 존재하게 된다.

2.4.2 이상 패턴 분석

6가지 기본 이벤트 타입은 수많은 이벤트에 대해서 조치를 취해야 할 유효한 이벤트를 감지하고 효율적으로 대처할 수 있는 방법이다. 하지

만, 임계치의 설정값에 따라 무의미한 이벤트도 조치를 취하거나 장애로 이어질 이벤트를 감지하지 못할 수도 있다. 이를 보완하기 위하여 지속적인 이상패턴 분석을 통한 적정 임계치의 설정과 기본 이벤트 타입의 추가가 필요하다. <그림 4>는 대표적인 이상패턴 2가지를 분석한 결과이다. <그림 4>의 왼쪽 차트는 반복 임계 초과형 패턴으로 특정 기간 또는 시간 동안 반복적으로 임계치를 초과한 횟수를 바탕으로 장애를 사전 감지하게 되며 CPU utilization 등 성능 값의 급증 내역을 반복적인 임계초과로 예측하여 장애 점검포인트에 적용하게 된다. 오른쪽 그림



〈그림 4〉 이상 패턴 분석 예시

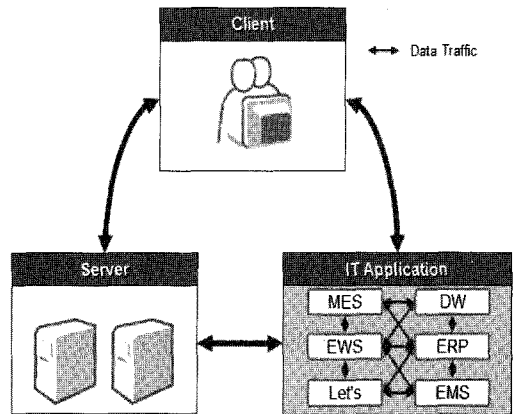
은 지속 증가형 패턴으로 특정 기간 또는 시간 동안 성능값의 통계량으로 회귀곡선을 구하여 미래 성능 값의 과다시점을 사전감지 하게 된다. 이는 Disk 사용량 등 성능값의 과다시점을 예측하여 장애 점검포인트에 적용하게 된다.

2.5 사전감지 시스템을 통한 장애 예측

장애예측과 장애예방으로 구성되는 사전예방 방법을 설명하기 위해 앞장에서 룰엔진을 소개 하였다. 이벤트타입을 정의하고 이상패턴 분석을 통해서 장애의 사전 예방이 가능한 영역은, 기존에 발생한 장애에 대해서만 가능하다. 즉, 기존에 발생한 장애분석을 통해 이벤트타입을 정의하고, 미리 설정한 이상패턴에 대해서만 예방이 가능한 것이다. 본 연구에서는 기 발생한 장애뿐만 아니라, 잠재된 발생 가능한 장애에 대해서도 예측이 가능하도록 사전감지 시스템을 제시한다.

지금까지 설명한 무장애체계의 사전예방, 사후조치 방법 모두, 개별 정보시스템을 대상으로 이벤트 모니터링, 감지, 조치 등을 취하게 된다. 하지만, 기업의 정보시스템들은 정보시스템간 유기적으로 연결되어 정보를 인터페이스하고 있으며, 그 복잡도가 매우 높다. 사전감지 시스템은 정보시스템간 데이터의 트래픽 형태를 모니터링하고 분석하여 장애를 예측하는 것이 목적이다. <그림 5>와 같이 정보시스템, 서버, 네트

워크 장비, 생산장비, End-User가 유기적으로 연결되어 데이터의 Transaction이 일어난다.



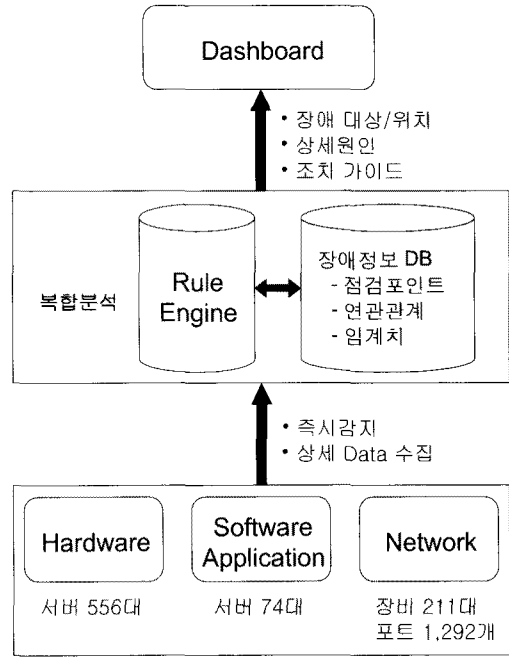
〈그림 5〉 사전감지 시스템 모니터링 구간

사전감지 시스템은 연결 네트워크상에 설치되어 시스템간 데이터 트래픽 형태를 모니터링 하고 이상현상이 발생할 경우 즉시 담당자에게 통보하게 된다. 사전감지 시스템에서 관리하는 트래픽 항목은 다음과 같다.

- 백업 트래픽 관리
특정 정보시스템의 백업 수행시 트래픽의 증가량을 관리
- 파일공유 트래픽 관리
End-User 간 파일공유시 비정상적 트래픽 증가량을 관리

- P2P, Instant Message 등 비업무 트래픽 관리
- Virus, DDOS 등 공격 트래픽 관리
- 5. 미식별 어플리케이션 트래픽 관리

장애예방을 위한 롤엔진 적용, 빠른 사후조치를 위한 점검 포인트 도출과 함께 사전감지 시스템 적용을 통해, 기업의 정보시스템에서 장애가 일어날 개연성이 있는 전 영역에 대하여 모니터링 체계를 구축할 수 있으며, 특정 시스템에서 장애 발생시, 그로 인해 영향이 미치는 타 시스템들에 대한 파악 및 장애예상 구간의 파악이 용이하다. 트래픽 성능 관리 솔루션 업체인 Netscout Systems의 138개 업체 대상으로 진행한 2007년 Survey 결과에 따르면, 트래픽 성능 관리 솔루션을 적용하여 평균 69%의 MTTR(Mean Time to Repair)을 시간으로는 약 6시간을 단축하는 것으로 보고되고 있다(Brian Robertson, 2007).

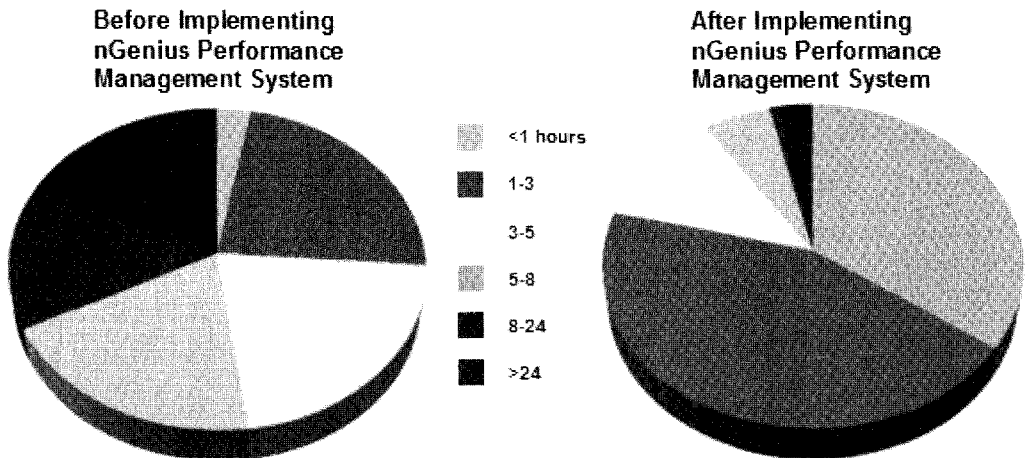


〈그림 7〉 무장애체계 시스템 개념도

2.6 무장애체계 구현

지금까지 설명한 장애관리 프로세스와 장애 점검포인트, 이벤트 롤엔진 적용을 통해 장애 사전예방이 가능한 무장애체계 방법을 적용하여 장

애크리 시스템을 A사에 구축 하였다. <그림 7>은 A사에 구현한 무장애체계 시스템의 개념도이다. A사의 정보시스템을 하드웨어, 어플리케이션, 네트워크 영역으로 구분하고, 각 영역의



69% Time Savings ≈ 6 Hours Time Savings

〈그림 6〉 Netscout Systems Survey 결과

운영중인 서버 630대, 네트워크 장비 211대, 네트워크 포트 1,292개 대상으로 이벤트를 즉시 감지하고 상세 Data를 수집하게 된다. 감지된 이벤트와 상세 Data는 장애점검포인트를 포함하는 장애정보 DB와 룰엔진 간 복합분석을 수행하게 되며 수행된 결과는 Dashboard에서 실시간으로 확인이 가능하다. Dashboard에서는 룰엔진에서 정의된 임계치를 벗어나는 이상 이벤트들에 대해서 경고를 해주고, 만약 장애로 발생될 경우 장애 대상 및 위치, 상세원인, 조치가이드를 장애정보 DB로부터 검색하여 실시간으로 제공해준다.

<그림 8>은 실제 A사에 구축된 무장애체계 시스템 구성도이며 6개의 관리계층으로 구성된다.

1) 사용자 연동 계층

관리자, 운영자를 위한 UI 인터페이스 계층으로, 웹 상에서 권한/인증에 따른 차별화된 사용자 인터페이스를 제공하며, 실시간 모니터링을 위한 기술을 도입한다.

2) 서비스 관리 계층

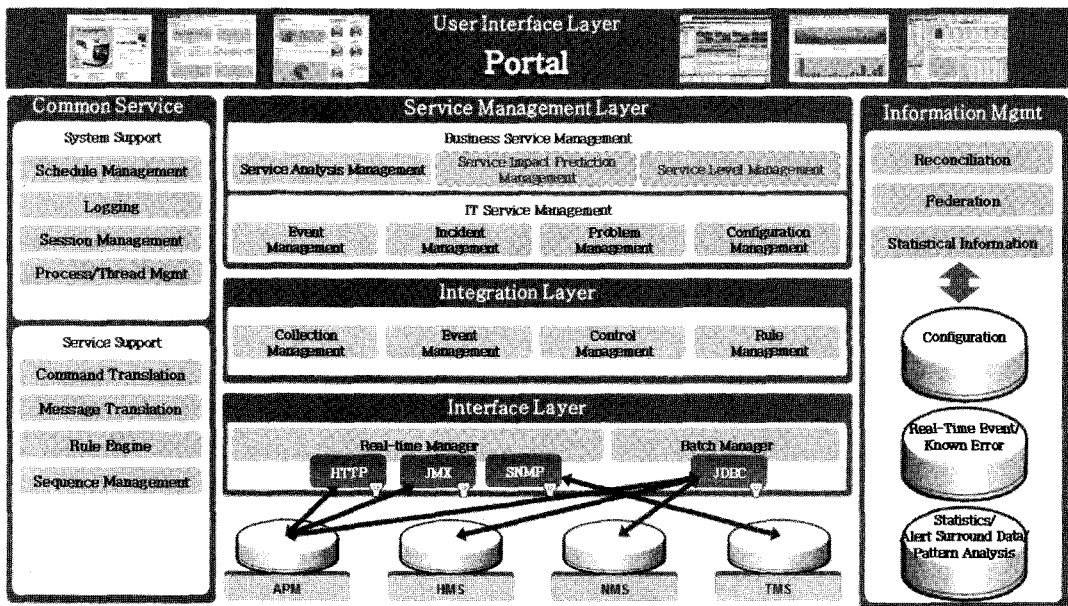
정보 관리 계층에서 제공되는 구성/상태/관계 정보를 이용하여 Business Service와 IT Service를 제공하는 계층이다. 또한, 실시간 정보를 연동 통합 계층으로부터 수신하여 실시간 서비스를 제공한다.

3) 연동 통합 계층

관제 서비스로부터 데이터를 수집하여 필요 데이터 추출(Data Extraction), 패턴 분석(Pattern Analysis), 메시지 변환(Message Transformation), 그리고 정보 관리 계층 또는 서비스 관리 계층(Real-Time Event 또는 Provisioning)으로 전달하는 계층.

4) 요소 관리 계층

JDBC Connection, HTTP Listener, SNMP Server, JMX 등 다양한 인터페이스(Interface) 채널을 통하여 APM, HMS, NMS로부터 성능 및 이벤트 정보를 수집하기 위한 계층.



<그림 8> 무장애체계 시스템 구성도

5) 정보 관리 계층

수집된 구성, 성능, 장애 정보를 관리 하는 계층으로 Reconciliation, Federation, Statistical Information 기능을 제공한다.

- **Reconciliation:** 여러 소스에 정보를 하나의 CI로 인식하는 방법 제공
- **Federation:** 여러 소스에서 제공되는 정보를 하나의 통합 VIEW로 제공.
- **Statistical Information:** 여러 소스에서 제공되는 Data를 통계 정보로 재 가공

6) 공통 서비스 계층

IT 서비스(Job Scheduling, Logging etc)와 비즈니스 서비스에서 공통으로 사용하기 위한 지원 유틸리티 및 비즈니스 서비스를 제공하는 계층이다.

MTBF = Mean Time Between Failures

$$= 1/\lambda; \lambda \text{는 일정 기간동안의 장애건수}$$

- 가용성(Availability)
 - 가용성 지표는 시스템 또는 장비가 어느 순간에도 안정적이고 효과적으로 운영될 가능성으로 정의한다.
 - 측정법

$$A = \frac{MTBF}{MTBF + MTTR}$$

MTTR = Mean Time To Repair

$$= \frac{\sum_{i=1}^N t_i}{N}$$

t_i: 장애 인지에서 조치까지 걸리는 리드타임 (단위: 시간)

N: 특정 기간 동안의 장애건수

III. 결 과

3.1 무장애체계 효용성 검증

무장애체계 프로세스와 정보시스템을 글로벌 하이테크 장치산업의 대표기업인 A사에 적용하여 무장애체계의 효용성을 검증 하였다. 앞장에서 설명한 사전예측과 사후조치의 장애관리 프로세스가 모두 적용된 정보시스템 통합 모니터링 시스템(PICS)을 구축하여 A사 정보시스템의 하드웨어, 어플리케이션, 데이터베이스, 네트워크의 4개 영역 모두에 대해서 장애관리를 수행 하였다.

무장애체계 효용성 검증을 위해 신뢰성(Reliability)과 가용성(Availability)의 정량적 지표를 사용하였다(Isermann and Ball, 1997).

- 신뢰성(Reliability)
 - 신뢰성 지표는 시스템 또는 장비가 정상적인 상태에서 주어진 기능을 요구되는 능력으로 정의한다.
 - 측정법

무장애체계 방법을 적용하기 전인 2009년과 적용 후인 2010년의 각 1월부터 4월까지의 A사의 Data 기준으로 장애 시 생산에 영향을 미치는 A, B등급 장애건수는 2009년 15건과 2010년 4건으로 73% 줄어 들었으며, 대상 장애 건에 대하여 원인분석 및 조치 리드타임(Lead Time)인 MTTR을 분석한 결과를 보면 발생한 장애에 대한 원인분석 시간은 75%, 조치는 33% 개선되어 무장애체계 방법 도입 후 전체적으로 장애대응 시간이 46% 감소하는 결과를 보였다. 이러한 Data를 통해 신뢰성 지표와 가용성 지표를 산출해보면 신뢰성 지표는 0.07에서 0.25로, 가용성 지표는 0.04에서 0.23으로 크게 상승하는 결과를 보였다.

<표 5> 무장애체계 효용성 검증 결과

지표	도입 전	도입 후
신뢰성(Reliability)	0.07	0.25
가용성(Availability)	0.04	0.23

DFS(Defect File System)에서의 실제 사례를 통해 무장애체계 도입을 통한 사전예방단계 개선

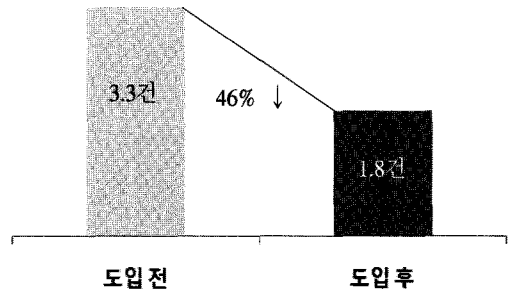
결과를 표현 하였다. DFS는 검사장비의 Defect Image를 전송하기 위한 Framework으로 2007년부터 2009년까지 7건의 장애가 발생하였으며, 원인 분석에서 조치까지 걸리는 평균시간인 MTTR은 118분 이었다. 이제까지 알려진 장애원인은 4가지로, Disk Full, Script 오작동, Script Exception 처리, CPU 과집이었다. 4가지 장애원인에 대해 PICS 시스템에서 DFS에서의 장애 사전 예방을 위한 Script Event 감지를 위해 하기의 개선활동을 진행하였다.

- Defect file 보관 주기 개선
- 검사장비 증가로 인한 서버 Capa Up
- Defect File 삭제 Script 개선
- Script Exception 발생시 Event 발생
- Defect file 사용량 모니터링 신규 적용
- Defect file 처리 모니터링 툴 개선 및 Event 발생
- 운영 Script Audit 및 작성 표준 개선

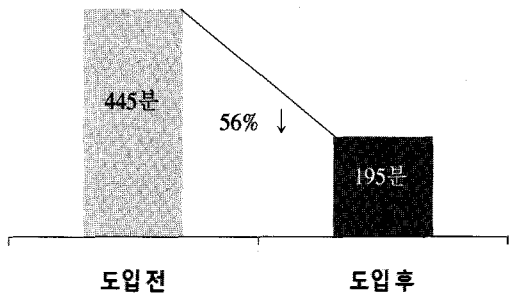
사전예방 Process를 PICS 적용 후 2010년 1월 ~5월 간 DFS 장애 관련 70건의 Script Event를 실제로 선 감지하였으며, Script 오류 수정을 통해 사전 감지한 Event가 실제 장애로 연결되지 않도록 예방하는 결과를 보였다.

3.2 기대 효과

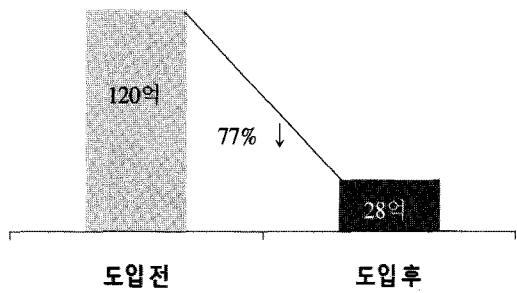
정량적인 효과를 보게되면 <그림 9>와 같이 무장애체계 도입전 월별 장애 발생건 3.3건에서 도입 후 1.8건으로 46% 감소하였으며, 장애발생 시간은 평균 56% 감소하는 효과를 보였다. <그림 11>과 같이 무장애체계 도입 전, 후의 매출에 대한 기회 손실 비용을 계산 하였다. 도입 전은 상반기 6개월 간 A사의 공장별 분당 매출액에 공장별 장애시간과 공장별 가중치를 곱하여 계산하였다. 도입 후는 동일한 산식으로 하반기 5개월 간을 기준으로 계산하였다. 비교 결과 92.4억 감소하여 도입 전 대비 매출에 대한 기회손실을 77% 절감했음을 알 수 있다.



<그림 9> 월평균 장애건수 비교



<그림 10> 월평균 장애 시간 비교



<그림 11> 도입 후 기회 손실비용 절감액

IV. 결 론

본 연구에서는 장애관리 프로세스를 정의한 후, 사후 장애조치 영역에 기술분류체계 기반의 장애 점검포인트를 도입하고, 이벤트 타입 정의를 통한 롤엔진을 적용하여 장애의 사전 예방과 예측이 가능한 무장애체계 방법을 제안 하였다. 또한, 기존의 장애관리 프로세스 연구와 차별화하여 제안한 무장애체계 방법을 글로벌 하이테크

크 장치 산업의 대표기업인 A사에 구축하여 도입 전, 후의 결과를 실제 데이터를 근거로 제시한 데에 의의가 있다. 앞으로 많은 기업들이 현재보다 더 많은 정보시스템을 구축하고 운영할 것이며, 그 복잡도의 증가로 인해 관리의 어려움은 더욱 커져갈 것이다. 이 정보시스템들에서 일어나는 수분 동안의 장애도 해당 기업에게는 엄청난 금액적 손실을 가져올 수 있다. 향후의 장애관리체계는 사후조치가 아닌 사전 예방 및 예측 중심이 되어야 할 것이다.

본 연구에서 제시한 기본 이벤트 타입 6가지와 이상 패턴 분석을 통한 롤엔진은 전자, 제조 산업 중심으로 개발 되었다. 하지만, 금융, 통신, 서비스 등의 타 산업 및 해당 기업에 특화된 정보시스템의 이벤트를 감지하고 분석하는 롤엔진으로 수정 적용 한다면 무장애체계 도입이 가능할 것이며, 이로 인해 정보시스템으로 인한 기업의 손실을 최소화 할 수 있을 것으로 기대된다.

참 고 문 헌

- 김문희, “결합허용 시스템의 설계 고려사항 및 동향”, 한국정보과학회지, 제11권, 제3호, 1993, pp. 7-16.
- 김승남, 박민현, 한육표, 정연중, “실시간 모니터링 기술을 이용한 웹서버 장애관리 시스템”, 강원대학교, 2006.
- 박선철, 이원영, “정보시스템 장애요인 분석을 통한 운영 프로세스 개선안 연구”, 한국경영과학회, 추계 학술대회논문집, 2008, pp. 136-140.
- 백운기, “애플리케이션 거버넌스는 GSI 환경의 필수”, CIO BIZ, 2010.
- 신용우, 안영덕, “IT Governance 실현을 위한 통합 ITMS 구축 방안”, SAMSUNG SDS Consulting Review, 제2권, 2006, pp. 54-68.
- 이치현, 강승원, 서종현, “Global Single Instance 기반의 ERP 시스템 통합 Framework”, *Entruse Journal of Information Technology*, 제6권, 제2호, 2007, pp. 127-138.
- 정보통신산업진흥원, “정보시스템 장애처리 지침”, 2005.
- 정아주, 박준상, 이상우, 김명섭, “SNMP를 이용한 실시간 장애관리 시스템의 개발”, 한국정보처리학회, 제16권, 제1호, 2009, pp. 1317-1320.
- 조성민, 한혁수, “CMMI 기반의 결합 분석 및 통제 시스템 개발”, 인터넷정보학회논문지, 2007, pp. 15-22.
- 홍지윤, 조광문, “생산 자동화 시스템 운영에서의 장애 검출 방안”, 한국콘텐츠학회, 제1권, 제2호, 2003, pp. 167-170.
- Brian, R., Netscout Systems Survey, 2007.
- Isermann, R. and P. Ball, “Trends in The Application of Model-Based Fault Detection and Diagnosis of Technical Processes”, *Control Eng. Practice*, Vol.5, No.5, 1997, pp. 709-719.
- Johnson, B. W., *Design and Analysis of Fault Tolerant Digital Systems*, Addison Wesley, 1989.
- Kim, K. H., “Design of Real-Time Fault Tolerant Computing Stations”, Lecture Note in the NATO Advanced Science Institute on Real-Time Computing, Sint Maarten, 1992.
- Laprie, J.-C. et al., “Definition and Analysis of Hardware and Software-Fault-Tolerant Architectures”, *Computer*, 1990, pp. 39-51.
- Randell, B., “System Structure for Software fault tolerance”, *IEEE Trans, on Software Engr*, 1975, pp. 220-232.

Fault-Free Process for IT System with TRM(Technical Reference Model) based Fault Check Point and Event Rule Engine

Byeong Tag Hyun* · Tae Woo Kim* · Chang Sup Um* · Jong Hyen Seo**

Abstract

IT Systems based on Global Single Instance (GSI) can manage a corporation's internal information, resources and assets effectively and raise business efficiency through consolidation of their business process and productivity. But, It has also dangerous factor that IT system fault failure can cause a state of paralysis of a business itself, followed by huge loss of money. Many of studies have been conducted about fault-tolerance based on using redundant component. The concept of fault tolerance is rather simple but, designing and adopting fault-tolerance system is not easy due to uncertainty of a type and frequency of faults. So, Operational fault management that working after developed IT system is important more and more along with technical fault management. This study proposes the fault management process that including a pre-estimation method using TRM (Technical Reference Model) check point and event rule engine. And also proposes a effect of fault-free process through built fault management system to representative company of Hi-tech industry. After adopting fault-free process, a number of failure decreased by 46%, a failure time decreased by 56% and the Opportunity loss costs decreased by 77%.

Keywords: Fault Management, IT System, Technical Reference Model, Fault-Tolerance, Rule Engine

* LG CNS

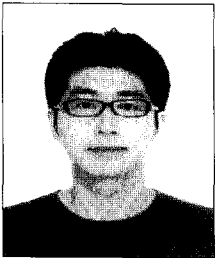
** Department of e-Business, Korea Polytechnic University

◎ 저 자 소 개 ◎



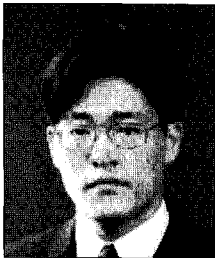
현 병 탁

LG 전자에서 20년간 재직하면서 프로세스혁신 및 GSI 관점의 ERP구축을 진행하였으며, LG 디스플레이에서 업무혁신담당 상무로 재직하면서 전사의 ERP, MES 혁신을 주도하였다. 최근에는 U.C Sandiego에서 Executive MBA를 수료하였으며, 현재 LG CNS 서비스 사업부 상무로 재직 중이다. 관심분야는 전자 IT 혁신, 스마트워킹, GSI 등이다.



김 태 우 (tw.kim@lgcns.com)

서울대학교에서 Biomechanics를 전공으로 석사학위를 취득하였다. 현재 LG CNS 컨설팅 부문에 재직 중이며, 하이테크 산업 분야의 다수의 컨설팅 프로젝트를 수행하였다. 주요 관심분야는 PLM, CAD/CAE, SCM, 패턴기반전략 등이다. 최근에는 기업의 스마트워킹을 위한 전자차원의 업무 Digital화에 관심을 갖고 연구 중이다.



엄 창 섭 (csum@lgcns.com)

한국기계연구원 Robotics Lab.에서 GRAFCET(Sequential Functional Chart), AGV(Automated Guided Vehicle) 국산화 및 생산관리시스템을 연구하였으며, 현재 LG CNS 엔지니어링서비스부문에서 제조 컨설팅을 수행하고 있다. 주요 관심분야는 MES, EES, FA, EA 등이며, 최근에는 Smart Factory를 위한 생산성관리 및 핵심요소기술에 관심을 갖고 연구 중이다.



서 종 현 (jhseo@kpu.ac.kr)

한국과학기술원(KAIST) 산업공학과에서 학사, 석사 및 박사학위를 취득하였다. LG CNS 컨설팅 부문에서 선임컨설턴트를 거쳐 현재 한국산업기술대학교 e-비즈니스학과 조교수로 재직 중이다. 주요 관심분야는 중소기업 지원 정책, R&D 성과평가, 품질경영 및 신뢰성분석 등이다.

논문접수일 : 2010년 08월 10일

게재확정일 : 2010년 12월 15일

1차 수정일 : 2010년 12월 07일