

학습, 민활 조직 실현을 위한 프로세스 사슬 기반 정보시스템 개발과 에이전트 기반 소세계 시뮬레이션

박 광 호*

Process Chain-Based Information Systems Development and Agent-Based Microworld Simulation As Enablers of the Learning & Agile Organization

Park, Kwang-Ho

Identifying knowledge as the single most important asset ultimately defining organizational competitiveness, enterprises are trying to move towards knowledge-oriented practices. Such practices have given rise to learning and agile organization. This paper presents applied information technologies to realize the learning and agile organization, focusing on systems thinking. Firstly, in order to establish a framework for the systems thinking, an information systems development method based on process chain is proposed. Then, an agent-based microworld simulation approach is presented. The approaches provide visible and analytical information to knowledge workers so that they can have systems thinking capabilities eventually. Various microworlds on the top of the information system can be constructed with agents and simulated for possible business events. All decision makings are dynamic in nature. To let knowledge workers look ahead the possible outcomes of the whole relevant processes is the core capability of the approaches. Through watching, the knowledge workers would be able to acquire new insights or problem solving knowledge for the problem in hand.

* 한양대학교 경상대학 경영학부 부교수

I. 서론

Drucker[1993]가 지적한 바와 같이 현대 사회는 육체 노동자에서 지식 노동자의 세계로 전환되고 있다. 지식 노동자가 전체 노동 인력의 80%를 상회하게 됨에 따라 기업들은 조직 구성원의 지속적인 지식 수준의 향상에 커다란 관심을 가지게 되었다. 이런 관심은 학습 조직[Senge, 1990], 민활 조직[Zavacki, 1997; Gunneson, 1997] 등과 같은 신 조직 개념으로의 전환을 요구하게 된다. 지식 자산에 대한 가치 부여와 이를 통한 경쟁우위 확보는 대량 요구(Mass Customization) 시대에 있어 당위론적 과제인 것이다.

90년대에 들어 프로세스 리엔지니어링[Hammer and Champy, 1993; Davenport, 1993; Davenport and Stoddard, 1994]과 같은 프로세스 중심적 경영 혁신 운동이 활성화 되었는데, 이는 프로세스가 지식 노동자의 생산성과 직접적으로 관련되며 기업의 원가와 품질을 좌우하기 때문이다. 본격적인 프로세스 경쟁시대[Taylor, 1992]가 도래함에 따라 프로세스에 대한 관심이 증대된 것이다.

학습, 민활 조직과 프로세스 중심적 경영 혁신 방법의 연관성은 목적과 도구의 관계로 설명될 수 있다. 프로세스 리엔지니어링에 의한 프로세스 혁신이나 재설계가 학습, 민활 조직의 실현을 위한 도구가 될 수 있다는 것이다. 조각나 있던 프로세스를 통합하고 이를 개인화 하는 프로세스 리엔지니어링은 학습 조직의 실현 방법 중의 하나인 시스템 사고와 맥락을 같이 한다고 볼 수 있다. 또한, 프로세스의 개인화와 프로세스의 병렬 처리는 민활 조직의 실현에 기여하게 된다. 정보기술은 프로세스 혁신을 가능하게 하는 주요 도구(Enabler)로 인식되고 있다[Hammer and Champy, 1993]. 프로세스의 조각화(Fragmentation)에 따른 부서간의 커뮤니케이션 단절과 이에 따른 일관성 상실, 순차적 업무 처리에 따른 지연, 지역 최적화 치중에 따

른 전체적 최적화 인식 결여 등 많은 문제를 해결하기 위한 방편으로 정보기술이 사용되는 것이다. 정보 기술이 프로세스 혁신을 가능하게 하는 도구로 인식되는 것과 같이 정보 기술이 학습, 민활 조직의 실현을 위한 도구가 될 수 있다는 결론도 내릴 수 있게 된다.

본 논문에서는 학습, 민활 조직으로의 성공적 전환이 기업 경쟁력 제고의 전략적 방편이 될 수 있다는 전제 하에 이를 효과적으로 실현할 수 있는 정보시스템 개발 방법과 이를 기반으로 한 시물레이션 방법을 제시하고자 한다. 프로세스 사슬(Process Chain)을 기본 도구로 정의하여 프로세스에 대한 재설계가 정보시스템으로 구현될 수 있는 방법을 제시하였다. 프로세스 사슬은 프로세스를 단순한 수준에서 가장 복잡한 수준까지 다양한 단계의 계층적 시스템 분석을 가능하게 하는 기본 도구로서 정보시스템의 적극적인 운영을 시스템화한 것이라 볼 수 있다. 또한, 적극적인 운영에 대한 동기 부여를 위해 평가 측정치를 정의하여 프로세스 사슬 수행에 대한 담당자들의 업무 평가를 위한 기본 도구로 사용하였다.

이상의 프로세스 사슬과 평가 측정치를 기반으로 소세계 시물레이션(Microworld Simulation) 방법을 개발하였다. 소세계 시물레이션은 학습, 민활 조직으로의 전환에 필수적인 시스템 사고를 가지게 하는 기본 프레임워크이다. 다수의 에이전트로 구현되는 소세계 시물레이션은 시물레이션 과정을 통해 시스템 사고를 가능하게 할 뿐만 아니라 학습을 통한 지식 수준 향상에 크게 기여한다. 이는 전통적인 전략 게임[Hickman, 1994], 비즈니스 게임[Newstrom and Scabbekk, 1996], 조직 게임[Hickman, 1995]을 통한 학습 방법이라고 볼 수 있다. 그러나, 정보시스템의 사용을 전제로 한 프로세스 중심적 시물레이션이라는 점에서 보다 구체적이고 실제적인 학습 방법이다.

프로세스 중심의 시물레이션 방법에 대한 연구

가 발표된 바 있는데 정보시스템 개발을 위한 모델[Briccarello, 1995; Warren, 1992; Eddins, 1991] 연구와 프로세스 혁신을 위한 모델[조윤희 외, 1998] 연구로 분류할 수 있다. 선행 연구와 본 논문에서 제시하는 시물레이션 방법은 다음과 같이 비교해 볼 수 있다. 첫째, 프로세스를 중심으로 시물레이션을 한다는 점에서 공통점이 있다. 둘째, 본 논문에서 제시하는 시물레이션은 통합정보시스템을 기반으로 한다는 점에서 차이가 있다. 또한, 통합정보시스템은 프로세스 사슬에 기반하여 개발되는 것을 전제로 한다. 마지막으로, 프로세스 혁신 차원이 아닌 학습, 민활 조직으로의 전환을 위한 기반 구축을 목표로 한다는 점에서 차이가 있다.

본 논문에서 제시하는 방법은 조직의 지식 창조 모델[노나카, 1990]에서 지식을 창조하는 개별 단계와 관련되기 보다는, 조직 상황, 조직적 의도, 동요, 혼돈 등 지식을 생성할 수 있는 기반 구축과 연관성이 있다고 보아야 할 것이다. 지식 생성을 위한 기반 구축은 다음과 같이 실현된다. 첫째, 프로세스 담당자가 정보시스템으로 프로세스를 처리하는 상황을 시물레이션으로 재현한다. 이 과정을 통해 암묵지, 형식지 등 지식이 생성 될 수 있을 것이다. 둘째, 다단계 시물레이션을 통해 조직의 규범적 의도와 이에 대한 개인의 인식 사이의 갈등을 표출시켜 새로운 지식을 생성하는 기반을 제공한다. 따라서, 본 논문은 지식 창조의 기반으로서 시스템적 사고와 비선형적 의사결정 지원 프레임워크를 제시하고 구현하고 있다.

II. 학습, 민활 조직 실현을 위한 개념적 접근 방법

2.1 시스템 사고 실현

학습 조직에 대한 정의는 Argyris[1977],

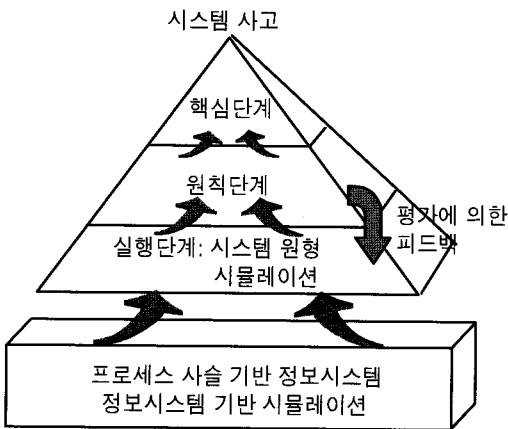
Beck[1992], Fiol and Lyles[1985], Garvin[1993], Pedler, et al.[1991], Senge[1990] 등에서 찾아볼 수 있다. 학습 조직의 실현 방법으로 제시된 시스템 사고(Systems Thinking)[Senge, 1990]는 단편적인 사태에 대한 수동적 대응이나 분석이 아니라 복잡한 전체 시스템 구조를 통찰하는 것을 목표로 한다. 시스템 사고는 실행, 원칙, 핵심 등 3 단계를 거쳐 실현될 수 있다[Senge, 1990]. 실행은 시스템 사고를 가지는데 있어 가장 구체적으로 가시화되는 단계로 시스템 원형(System Archetypes)과 시물레이션을 사용하게 된다.

시스템 사고의 실현을 위한 방법으로 <그림 1>과 같이 실행 단계에 초점을 둔 시스템 원형과 시물레이션 방법을 제시하였다. 복잡한 시스템에 내재하고 있는 구조를 인식하기 위한 원형으로 프로세스 사슬에 기반을 둔 통합정보시스템을 구축하고 통합정보시스템 운영을 시물레이션 하는 방법을 사용한 것이다. 프로세스 사슬에 의한 시스템 원형 구축은 정보시스템이 단순한 트랜잭션 처리에서 경쟁 무기로서의 정보 처리, 그리고 더 나아가 정보나 정보시스템 자체가 제품으로 인식될 수 있는 정보시스템의 역할 변화로 귀결되는 것이다[Beck, 1992; Augustine, et al., 1993]. 이런 변화의 맥락에서 학습 조직을 실현하는 정보시스템의 새로운 역할을 정의할 수 있는 것이다. 프로세스 처리의 복잡한 과정은 정보시스템으로 가시화 되어 시스템 원형으로 표출되며, 이 시스템 원형에 따라 프로세스 담당자들은 각 세부 작업을 수행하게 된다.

실행 단계의 시스템 원형과 시물레이션은 조직 구성원이 원칙에 의거하여 프로세스를 수행하도록 유도한다. 프로세스 사슬 개념에 기반하여 개발된 정보시스템을 사용하게 되는 프로세스 담당자는 프로세스 사슬 구조에 따라 프로세스를 수행해야 하기 때문에 시스템 사고의 가장 중요한 원칙인 구조가 행위를 결정한다 원칙을 따르게 된다. 또한, 통합정보시스템 운영 실적은

정보시스템의 활용도와 이에 대한 저항력을 측정할 수 있게 한다. 이런 측정 결과는 실행 단계에 피드백 되어 실행 의지를 강화시킨다.

마지막으로 핵심은 실행과 원칙 단계가 실현될 때 도달하게 되는 상태를 의미한다. 프로세스 사슬이 구현된 정보시스템의 사용과 이를 기반한 시물레이션은 프로세스 담당자가 부분보다 전체를 볼 수 있는 사고력의 확장과 상호연결성을 인식할 수 있는 상태에 도달하게 되는 것이다.



<그림 1> 시스템 사고 실현 개념도

2.2 비선형 의사결정 지원

민활 조직이란 작은 단위의 자율적 집단으로 구성되어, 예상할 수 없이 급격하게 변화하는 시장 환경에 살아 남기 위해 신속하고 효과적으로 의사결정 해야 하는 조직이다[Zavacki, 1997; Gunnesson, 1997]. 따라서, 민활 조직에서는 모든 작업이 동시에 신속하게 효율적, 효과적으로 수행될 수 있어야 한다.

민활 조직은 학습 조직과 같이 시스템 사고에 의해 실현될 수 있다. 외부의 예상하지 못한 사건의 발생에 대해 업무 담당자가 단편적인 프로세스 지식이 아닌 전체 프로세스에 미치는

영향을 정확히, 신속하게 분석할 수 있을 때 경쟁적 우위를 확보하게 되는 것이다. 기업 조직은 기본적으로 혼돈 세계의 특성을 가지고 있다. 기업의 가치 부가 과정을 살펴 보면, 부서별 담당자가 프로세스를 수행하여 개별적 가치를 추가함으로써 최종적인 제품과 서비스가 창출됨을 알 수 있다. 이상의 일련의 프로세스 수행과정은 본질적으로 불안정하다. 동일한 주문이라도 주문 시점에 따라, 주문에 대한 생산 계획 시점에 따라, 자재 입고 시점에 따라, 품질이나 원가에 큰 차이가 있을 수 있다. 또한, 사소한 주문 변경이나 사양 변경이라도 전체적으로 큰 영향을 주게 된다.

민활 조직으로의 전환은 결국 프로세스 담당자가 프로세스의 비선형적 특성에 대해 정확히 이해해야 가능하다. 수많은 의사결정이 상호 피드백을 주면서 예측할 수 없는 혼돈의 상태로 돌진하는 본질적 특성을 정확히 파악해야 하는 것이다. 따라서, 프로세스 담당자의 역할을 재현하여 개별적인 의사결정이 전체적으로 어떤 결과를 낳는지 관측할 수 있는 시물레이션 환경을 제공하는 것이 민활 조직 실현의 방법이 될 수 있다는 결론에 도달할 수 있다. 개별 프로세스 담당자는 에이전트로 구현되어 에이전트로 구성된 가상의 프로세스 처리 환경을 구축하였다. 이와 같은 에이전트 기반 시물레이션은 개별 에이전트의 행동 변화가 전체적으로 어떤 결과를 가져 오는지 관측할 수 있는 환경을 제공한다. 예를 들어, 자재 담당자의 부주의한 자재 입고 처리는 제품 불량 발생 뿐만 아니라 전체 생산성 저하, 납기 지연에 따른 고객 신뢰 상실 등 커다란 결과로 증폭될 수 있다는 것을 시물레이션을 통해 보여 줄 수 있다.

사건의 발생에 대해 정확히 그 결과를 예측할 수 있을 때 민활 조직으로 전환할 수 있음을 알게 된다. 작은 변화가 가져 다 줄 지역적 혼돈을 예측하고 전체적으로 안정된 상태로 복귀할 수 있는 시간과 공간에 대해 예측할 수

있어야 하는 것이다. 이상과 같은 현대 기업의 동태적 특성은 혼돈 이론에 기반을 둔 프랙탈 경영[이장우, 박형규, 1998]에 대한 연구에서 체계적으로 정리되었다. 민활 조직의 실현을 위해서는 집합적 프로세스의 혼돈적 행동의 재현이 매우 중요하다. 이는 개별 프로세스의 담당자들이 자신의 프로세스만을 최적화할 때 회사 전체 최적화가 실현될 수 있다는 착각을 자각시켜 줄 수 있을 것이다. 따라서, 비선형적 특성을 가지는 다수의 프로세스의 집합적 행동을 가시화해 줄 수 있는 시뮬레이션 모델을 구축하여, 프로세스 담당자들이 자신의 행동이 전체 회사에 미치는 영향을 관측할 수 있고, 이에 따라 민활하게 환경 변화에 대응할 수 있는 기반을 제공할 수 있는 것이다.

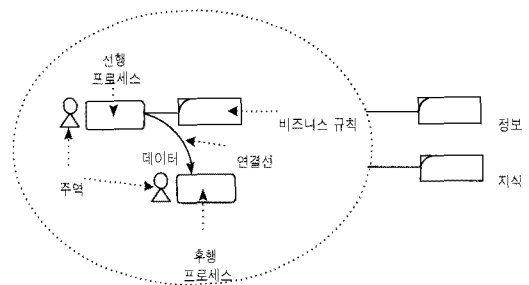
III. 프로세스 사슬 기반 정보시스템 개발

3.1 프로세스 사슬

프로세스 사슬은 상호 연관된 프로세스를 연결시킨 것으로 정의할 수 있다. 단위 프로세스 사슬은 <그림 2>와 같이 프로세스 사슬 표기법으로 정의된다. 두개의 프로세스, 즉 선행, 후행 프로세스가 있을 때, 이들 사이로 단 방향, 양방향의 연결선이 존재한다. 단방향은 프로세스가 후행 프로세스로 진행되는 경우에 사용하며, 양방향은 선행 프로세스에서 후행 프로세스로 갔다가 다시 선행 프로세스로 되돌아 오는 경우에 사용한다. 연결선 상에는 비즈니스 규칙이 부착되는데 이는 선행 프로세스에서 후행 프로세스로 프로세스가 진행될 때 적용되는 규칙을 의미한다. 또한, 각 프로세스에는 프로세스를 수행하는 주역(담당자)이 지정된다. 프로세스 사슬은 편의상 후행 프로세스를 기준으로 하여 후행 프로세스명으로 명칭을 정하기로 한다.

프로세스 사슬은 정상, 비정상, 예외 케이스 [Jacobson et al., 1992]로 각각 정의될 수 있다. 정상 케이스는 표준이 되는 처리 절차를 의미하며 비정상 케이스는 정상 케이스로 처리 중 오류 처리나 취소 등과 같이 비정상적인 사건 발생에 대한 처리 절차는 나타낸다. 마지막으로 예외 케이스란 표준 처리 절차를 벗어나는 예외적 상황에 대한 처리 절차를 나타낸다. 예외 케이스가 많을수록 프로세스 경쟁력은 떨어지며 정보시스템 개발이 보다 어려워진다.

프로세스는 핵심(Core), 지원(Support), 개선(Improvement) 등의 유형으로 분류할 수 있는데 [Kock et al., 1997] 핵심 프로세스란 가치를 추가하는데 직접적으로 기여하며 지원 프로세스는 핵심 프로세스의 효율적 수행을 지원한다. 개선 프로세스는 핵심, 지원 프로세스를 개선하는데 목적을 두고 있다. 본 논문은 이상과 같은 프로세스 유형에서 핵심 프로세스만을 대상으로 한다. 이는 핵심 프로세스가 학습 조직, 민활 조직으로의 전환에 일차적인 대상이 되기 때문이다. 특히, 개선 프로세스는 사실 본 논문에서 제시하고자 하는 방법론 자체를 의미하기 때문에 제외해도 무방하다고 볼 수 있다.



<그림 2> 프로세스 사슬

프로세스 사슬은 추상적 의미로 해석되지만 실제로 업무 현장에서 다양한 인스턴스가 발생할 수 있다. 프로세스 사슬의 수행은 필연적으로 데이터의 발생과 전달을 유발시킨다. 데이터

는 Kock et al.[1997]에서 정의된 바와 같이 지식과 정보의 전달 수단으로 생각할 수 있다. 데이터는 접수자가 해석할 때 비로서 정보나 지식이 되며, 정보나 지식은 반드시 데이터로 작성된 후 전달될 수 있는 것이다. 정보는 설명적이며 지식은 예측적 성격을 가진다[Dubin, 1976], [Camerer and Johnson, 1991]. 따라서, 프로세스 사슬에서 데이터, 정보, 지식을 각각 정의할 수 있다. Kock et al.[1997]은 프로세스의 수행과정에서 데이터가 다량 발생하며 발생한 대부분의 데이터는 예측적 성격을 가진 지식이기 보다 기술적(Descriptive)단계의 정보임을 밝힌 바 있다. 또한, 프로세스가 복잡해 질수록, 지식의 교환 비율이 상대적으로 높아짐을 발견하였다. 결국 지식의 질적 향상은 프로세스 사슬에서 교환되는 데이터가 예측적 성격을 띠게 될 때 실현된다고 볼 수 있는 것이다.

본 논문은 프로세스 사슬의 수행 과정에서 지식을 추출하는 지식 공학 방법을 제시하지는 않는다. 프로세스 사슬 구조하에서 프로세스를 처리한 결과를 제시하여 지식 공학 연구가 수행될 수 있는 기반을 제공할 뿐이다. 지식 공학 연구에 의해 추출된 지식은 프로세스 사슬에 반영되어 소세계 시뮬레이션의 효과를 향상시킬 수 있을 것이다.

프로세스 사슬은 계층적으로 정의될 수 있다. 집합 프로세스 사슬(Aggregate Process Chain)은 다수의 단위 프로세스 사슬(Unit Process Chain)이 연결된 상위 프로세스라고 할 수 있다. 예를 들어, [제품사양협의-견적-주문등록]으로 정의되는 주문 프로세스 사슬은 [제품사양협의-견적] 프로세스 사슬과 [견적-주문등록] 프로세스 사슬의 2개 단위 프로세스 사슬이 순차적으로 연결된 집합 프로세스 사슬로 정의된다. 또한, 프로세스 사슬은 기능 영역 내 연결과 기능 영역 간 연결 등으로 구분할 수도 있다. 일반적으로 프로세스는 기능 위주의 분류에 의해 정의된 작업들을 교차하며 수행된다[Gunnesson, 1997]는 것을

인정할 때, 기능 영역간 연결이 보다 보편적이라고 볼 수 있다.

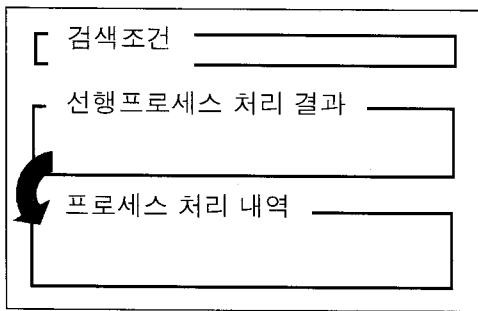
전체 프로세스 사슬은 일정계획법으로 활용되고 있는 AON(Activity On Node) 표기법으로 도식화할 수 있으며 PERT/CPM에서 사용하는 임계 경로(Critical Path) 분석이 가능하다[Taha, 1987]. 그러나, 프로세스 사슬은 일반적인 CPM 분석에서 가능한 단일 프로젝트에 대한 일정계획을 포함하고 있다. 따라서, 우리는 단순한 임계 경로 개념 보다는 사슬 분석(Chain Analysis)을 도입하였다. 사슬 분석은 가치 부가의 시작에서 끝까지 연속적인 프로세스 사슬의 집합을 대상으로 수행된다. 예를 들어, 제품 판매 사슬은 견적에서 시작하여 최종 입금까지의 연속된 프로세스 사슬이다. 기업의 경영 활동을 일정시점을 기준으로 분석해 보면 다수의 사슬이 존재함을 발견할 수 있다. 우리는 이런 다수의 사슬을 구별하기 위해 사슬 키(Chain Key)를 사용하였다. 예를 들어, 제품 판매 사슬의 사슬 키는 주문 번호일 것이다. 주문번호를 가지면 특정 제품 판매 사슬의 인스턴스를 추적할 수 있기 때문이다. 프로세스 사슬에 대한 사슬 키가 정의되면, 수행 과정에 대한 효과성, 효율성에 대한 평가를 할 수 있다. 이를 위해 평가 측정치를 정의하여 사슬에 대한 정량적 평가를 실시한다.

3.2 프로세스 사슬 템플릿

프로세스 사슬 템플릿은 프로세스 사슬 기반 정보시스템 개발에서 사용되는 기본 개발 구조를 말한다. 프로세스 사슬 템플릿은 <그림 3>과 같이 공통적으로 검색 조건, 선행 프로세스 처리 결과, 프로세스 처리 내역 등으로 구성된다. 프로세스 사슬 템플릿으로 개발된 프로그램으로 프로세스를 처리하는 절차는 다음과 같다.

작업 1: 검색 조건을 입력한다.

- 작업 2: 검색 조건에 맞는 선행 프로세스 처리 결과를 조회한다.
- 작업 3: 선행 프로세스 처리 결과를 보고 이 가운데 처리해야 할 트랜잭션 내역을 선택하여 이를 프로세스 처리 내역에 옮긴다.
- 작업 4: 프로세스 처리 내역을 최종적으로 확인하고 이를 저장하여 등록 작업을 마친다.



<그림 3> 프로세스 사슬 템플릿

프로세스 사슬 템플릿을 기반으로 개발된 정보시스템은 다음과 같은 장점을 가지게 된다.

- 프로세스 처리의 정확성이 지원된다. 선행 프로세스의 처리 결과를 가지고 이를 선택, 등록함으로써 등록되지 않은 트랜잭션 처리나 데이터 입력 오류가 방지된다.
- 프로세스 처리 속도가 향상된다. 처리해야 할 데이터를 찾아 보거나 확인할 필요가 없으므로 신속하게 트랜잭션을 처리할 수 있다.
- 프로세스가 주 키(Primary Key)와 외래 키(Foreign Key)로 연결되어 키 값을 중심으로 한 정보 추적이 가능하게 된다. 여기서 주 키와 외래 키는 사슬 키로 사용된다.
- 선후행 프로세스의 처리 시간을 각각 기록함으로써 전체 프로세스 처리 시간을 추적할 수 있다.

프로세스 사슬 템플릿에 기반한 정보시스템이 개발되면 프로세스 담당자는 해당 프로세스 사슬 프로그램에서 프로세스 인스턴스를 처리하게 된다. 이 때 데이터가 교환되며 필요 시 데이터베이스에 저장되는 경우가 발생하게 된다. 일반적으로 이런 데이터 처리 과정을 트랜잭션이라 하는데, 앞서 정의한 바와 같이 트랜잭션을 단순히 설명할 때 정보가 되고, 트랜잭션의 처리에 따라 발생할 결과를 예측하게 되면 지식이 발생 또는 교환된다고 해석할 수 있다. 따라서, 우리는 가능한 한 정보 차원에서 지식 차원으로 데이터 교환이 발생하도록 시뮬레이션을 통해 유도하는 것이 학습, 민활 조직으로의 전환 작업이라고 볼 수 있을 것이다.

IV. 에이전트 기반 소세계 시뮬레이션 (Agent-Based Microworld Simulation)

4.1 소세계 시뮬레이션 모델

시간과 공간을 압축한 소세계는 의사결정 미래에 어떤 결과를 가져 올 것인지를 실험을 통해 학습할 수 있게 하는 가상 세계를 말한다 [Senge, 1990]. 프로세스 사슬 관리에서 소세계는 몇 개의 관련 프로세스의 집합으로 정의할 수 있겠다. 예를 들어, 계약 소세계는 <견적-사양 협의-주문> 등 계약과 관련된 3개의 프로세스만으로 구성된 것으로 정의할 수 있으며, 생산 소세계는 <월생산계획-작업지시-자재출고요청-자재출고-조립-제품입고> 등 생산 계획에서 제품 생산 완료까지의 6개 프로세스를 포함하는 것으로 정의할 수 있다.

소세계 시뮬레이션 모델은 규범적 시뮬레이션 모델과 상황적 시뮬레이션으로 구성된다. 규범적 시뮬레이션 모델은 초기 상황과 일정 기간 동안에 발생했던 사건들을 입력하여 시뮬레

이선 하게 되면 규범적 규칙에 의한 활동 결과를 알 수 있게 된다. 이런 규범적 활동 결과와 실제 활동 결과를 비교하여 차이에 대한 원인을 분석함으로써 프로세스 처리에 대한 지식이 발전적으로 성장할 수 있도록 지원하게 된다.

2번째 소세계 시뮬레이션 모델은 가상적 미래 상황에 대한 학습을 실현하는 상황적 시뮬레이션 모델로 관측에 의한 수동적 대응과 피드포워드(Feedforward)에 의한 적극적 대응을 수반한다. 수동적 대응이란 측정 결과에 따라 단순히 정보 제공 차원에 그치고 이에 대한 교정 작업을 하지 않는 것을 말한다. 따라서, 수동적 대응은 사건에 따른 결과를 설명하고 인식하는 단순 정보 입수 수준에 그치게 된다. 반면에 적극적 대응이란 측정 결과를 분석한 후, 측정 결과를 교정해야 할 필요가 있을 때, 환경 변수를 변경시키고 이에 따라 다시 시뮬레이션 과정을 거쳐 결과를 측정하는 교정 과정을 만족스러운 결과가 측정될 때까지 지속하고, 최종적으로 환경 변수를 변경하여 이를 실 프로세스에 반영하는 방법이다. 상황적 시뮬레이션 모델은 다음 단계로 수행된다.

- 1단계: 측정 대상 소세계를 선정한다.
- 2단계: 결과를 측정하고자 하는 사건을 정의하고 이에 해당되는 환경 변수를 분석한다.
- 3단계: 결과를 측정하기 위한 측정치를 결정한다.
- 4단계: 사건이 발생하지 않았다 가정하고 일정 프레임 동안 에이전트가 활동하도록 한다.
- 5단계: 에이전트의 적응 결과를 3단계에서 결정한 측정치에 대해 측정한다.
- 6단계: 사건이 발생했다고 가정하고 일정 프레임 동안 에이전트가 해당 환경 변

수에 대응하도록 한다.

- 7단계: 에이전트의 적응 결과를 3단계에서 결정한 측정치에 대해 측정한다.
- 8단계: 사건의 발생에 따른 측정 결과의 차이를 분석한다.
- 9단계: 새로운 환경 변수의 변화에 의한 시뮬레이션이 필요하면 6단계로 간다. 그렇지 않으면 시뮬레이션을 중단한다.

소세계 시뮬레이션이 수행되면 반드시 결과에 대해 분석, 평가가 수반되어야 한다. 평가 없는 시뮬레이션은 효과적일 수 없는 것이다. 따라서, 시뮬레이션 결과에 대한 측정치를 도출이 요구된다. 프로세스의 성능 평가 기준은 조윤호 외[1998]에서 제시된 바 있는데 주로 활용도, 작업 소요 시간, Role 소요시간, 순환 주기 등 시간적 척도 위주로 구성되어 있다. 본 논문에서는 이상의 시간적 척도 뿐만 아니라 성과 척도를 포함시켜 측정의 대상에 따라 다음과 같이 두 가지 유형 척도를 사용하였다. 첫째, 민활도 측정치는 프로세스에 대한 처리 속도를 평가하기 위한 척도로 시간적 척도를 기준으로 측정된다. 예를 들어, 지연은 프로세스 처리 가능 착수 시간과 실제 처리 시간과의 차이를 측정하는 것으로 담당자 간 연계 프로세스의 지연 시간을 측정한다. 표준 대비 실적 시간은 표준 시간 대비 실제 처리 시간을 측정 비교한 것이다. 둘째, 성과 측정치는 프로세스 수행 결과에 대한 성과를 평가하기 위한 척도로 정확도, 달성 척도, 손실 등을 예로 들 수 있다. 정확도는 주로 지원 프로세스의 평가에 해당되는 것으로 처리 데이터의 무결성에 대한 평가로 오류율로서 측정된다. 달성 척도는 실적 수량, 실적 금액을 목표에 대비한 달성율로 측정된다. 손실은 손실 수량, 손실 금액 등을 목표에 대비한 손실율로 측정된다.

4.2 소세계 시물레이션 시스템

소세계 시물레이션 시스템은 프로세스 사슬 기반 정보시스템을 토대로 구축되었다. 정보시스템을 사용하는 프로세스 담당자의 역할은 에이전트가 맡게 된다. 에이전트는 마치 프로세스 담당자가 자재출고요청등록 프로그램에서 작업지시에 따른 자재 출고 요청을 하듯이 적절한 시점에 적절한 자재를 출고 요청 등록하게 된다.

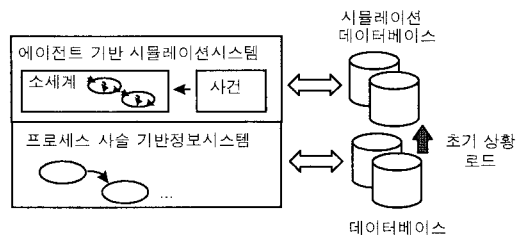
소세계는 다수의 관련 프로세스를 담당하는 에이전트 집합으로 정의되므로 시작 프로세스 에이전트부터 종료 프로세스 에이전트까지 에이전트를 추가하여 정의될 수 있다. 시작 프로세스는 내부 또는 외부 사건으로 유발된다. 따라서, 시물레이션을 시작하기에 앞서 내부, 외부 사건을 지정해야 한다. 자재 발주가 착수 프로세스인 소세계의 경우, 우선적으로 자재 발주가 수행되어야 시물레이션이 진행될 수 있으므로 자재 발주와 같은 내부 사건의 시점과 내역을 정하고 시물레이션을 시작하게 된다. 자재 접수가 착수 프로세스인 소세계의 경우에는 자재 발주 후 리드 타임을 감안한 자재 접수 사건을 정의하거나 자재 접수 사건의 시점과 내역을 사전에 정하여 시물레이션을 시작하게 된다.

일단 소세계가 정의되면 환경 변화로 시물레이션을 시작된다. 즉, 소세계에 어떤 충격을 주었을 때 어떤 결과가 나오는지 관찰하는 것이다. 충격은 사건(Event) 발생에 의한 환경 변수의 변화로 나타나며, 결과는 관련 에이전트의 대응에 따른 민활 측정치로 분석해 볼 수 있다. 이런 일련의 시물레이션 과정을 통해 프로세스 담당자가 보다 민활하게 대처하기 위해 요구되는 새로운 비즈니스 규칙을 발견하게 되며 그 결과, 새로운 지식을 학습하게 되는 것이다. 연속된 다수의 소세계에 대한 시물레이션은 전체적 사고를 갖게 하는 주요 도구가 된다. 하나의 소세계에 미치는 영향 분석은 지역 최적화에 머물고 전체적 영향에 대해 무지하게 되는 결

과를 낳을 수 있다. 따라서, 연속된 다수의 소세계를 연결시켜 시물레이션 범위를 확장하고 평가 측정치로 결과를 분석한다면 전체 최적화에 근접한 판단을 내릴 수 있을 것이다.

시물레이션은 일정 기간 동안 에이전트의 자율적인 의사결정에 따라 프로세스를 수행하는 방법으로 실행된다. 이를 위해서는 기존의 정보시스템과는 별도의 가상의 시물레이션 시스템의 개발이 요구되는 것이다. 즉, <그림 4>와 같이 일정시점을 기준으로 현재부터 일정 기간 동안 프로세스 사슬을 따라 시물레이션 사슬을 정의하여 수행하고 시물레이션 후, 사전에 정의된 측정치를 분석하여 적절한 대책을 제안하게 된다.

예를 들어, 주문 취소 사건이 발생하게 되면 생산계획 취소, 작업지시 취소, 자재소요량 등록 취소, 자재발주 취소, 자재입고 취소 등 일련의 가능한 취소 작업을 시도하며 취소 불가능한 프로세스 사슬부터 주어진 상황으로 가정하고 일정 기간 프로세스 사슬을 따라 수행한다. 한편, 주문 취소 사건이 발생하지 않았다는 가정하에 사슬을 정의하여 일정 기간 프로세스 사슬을 수행한다. 결국, 사건 발생에 따른 사슬과 정상적인 사슬의 차이 분석에 따라, 사건의 결과를 예측할 수 있게 되는 것이다.



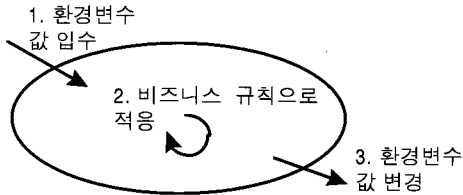
<그림 4> 시물레이션 시스템

4.3 에이전트 프레임워크

4.3.1 에이전트 단위 활동 주기 - 프레임(Frame)

에이전트는 <그림 5>와 같이 규칙적으로 외부

환경에서 필요한 데이터를 읽어 와 자신의 지능(비즈니스 규칙)에 따라 적절히 대처한 후, 그 결과를 외부 환경에 넘겨주는 일련의 단위 활동 주기를 지속적으로 수행하게 된다. 우리는 이런 단위 활동 주기를 프레임이라 부르기로 한다.



<그림 5> 에이전트의 단위 활동 주기

에이전트는 일단 시간에 대한 활동 조건 값이 입력되어 만족되면 활동에 착수한다. 예를 들어, 매일 오전 10:00에 활동해야 할 에이전트라면 오전 10:00가 되면 활동 조건이 만족되므로 활동에 착수할 것이다. 그러나, 시간에 의한 프로세스 활동 조건 만족이 반드시 실질적인 활동으로 연결되는 것은 아니다. 예를 들어, 매일 10:00에 생산계획을 수행하는 프로세스의 경우, 생산계획 대상이 되는 주문 접수가 없을 경우, 시간 조건이 만족되었지만 실질적인 생산계획을 수행되지 못하는 것이다.

일단 활동이 결정되면 사전 조건에 해당하는 데이터 값들이 입력된다. 사전에 활동 대상 객체(주로 데이터베이스 테이블이나 저장 프로시저)가 정해져 있으므로 이 값을 읽어 사전 조건을 체크하게 된다. 최종적으로 작업에 착수하게 되면 작업 규칙(저장 프로시저나 객체로 구현)에 따라 작업 대상 데이터를 가져온다.

에이전트 활동 결과는 데이터베이스에 반영된다. 데이터베이스에 기록되는 데이터는 트랜잭션 발생 결과와 활동 자체에 대한 기록으로 구분될 수 있다. 트랜잭션 발생 결과는 예를 들어 자재 입고의 경우, 자재입고와 자재입고내역 테이블에 신규 레코드가 삽입되는 것이며 활동 자체에 대

한 기록은 언제, 누가, 얼마 동안, 어떤 사슬 키에 대한 어떤 작업을 수행했는가 등이다.

4.3.2 수축적 활동(Systolic Operation)

모든 에이전트들은 같은 맥박(속도)로 동기화되어 구현되었다. 예를 들어, 모든 에이전트가 1초에 60프레임의 활동 주기를 가지게 된다고 가정하면 에이전트들이 각자 환경 변수를 읽고 비즈니스 규칙을 적용하여 환경 변수의 값을 변경하는 프레임워크를 60회 동안 같은 보조로 수행하게 된다. 한다. 이런 수축적 특성은 소세계로 구성된 프로세스 수행 상황에 대한 보다 현실적인 시뮬레이션을 가능하게 한다. 즉, 모든 에이전트가 같은 맥박으로 한 프레임을 진행하도록 함으로써, 환경 변화에 대해 같은 보조를 맞추게 되고, 그 결과 소세계의 비선형적 특성을 반영할 수 있는 것이다.

4.3.3 데이터베이스 중심 에이전트 통신

일반적으로 다수의 에이전트가 활동하는 경우, 에이전트 간의 통신 방법으로 KQML(Knowledge Query and Manipulation Language)[Labrou and Finin, 1997]과 같은 메시지 교환 방법이 사용되고 있다. 본 논문에서는 메시지 교환 방식에 의한 에이전트 통신 방법을 사용하지 않고 데이터베이스를 중심으로 한 간접적인 통신 방법을 사용하였다. 에이전트는 데이터베이스에서 환경 변수 값을 읽어 오게 되는데, 그 환경 변수 값은 다른 에이전트가 추가, 수정한 값이기 때문에 직접적인 통신은 없었어도 간접적으로 서로 메시지를 전달 받은 것과 같은 결과로 해석할 수 있기 때문이다.

데이터베이스를 중심으로 한 에이전트 간의 메시지 전달 방법을 채택한 것은 프로세스 사슬 기반 정보시스템 환경의 프로세스 담당자들도 선행 프로세스 담당자의 작업 결과를 조회해보고 이에 따라 적절한 등록, 수정 작업을 수

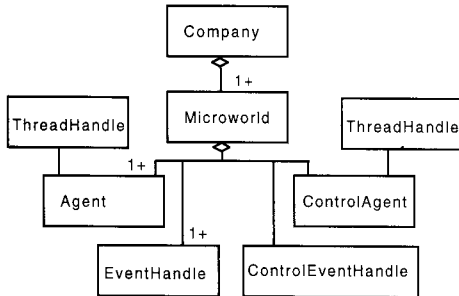
행하기 때문이다. 또한, 각 에이전트가 내부 활동 규칙에 따라 자율적으로 프레임을 지속적으로 반복하고, 다른 에이전트가 전달하는 메시지로 활동에 착수하지 않기 때문이다.

4.3.4 에이전트 클래스와 멀티 쓰레딩

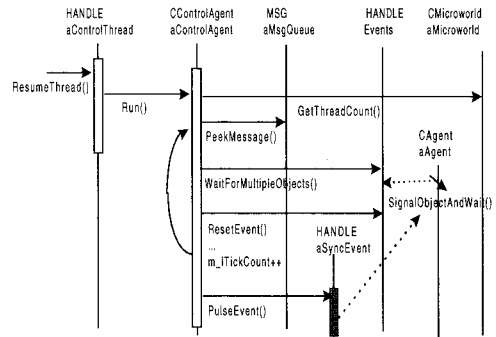
에이전트는 기본적으로 클래스로 구현되어 작업시기와 작업 모드와 관련된 속성과 작업을 위한 연산으로 정의된다. <표 1>은 에이전트 클래스의 주요 연산이다. 에이전트 클래스는 ATL COM[Grimes et al., 1998]을 사용하여 구현되어 인터페이스가 용이하도록 하였다. 소세계 시뮬레이션은 <그림 6>과 같이 소세계, 에이전트, 컨트롤 에이전트, 이벤트 핸들로 구현된다. 다수의 에이전트들로 수행되는 수축적 병렬 시뮬레이션을

구현하기 위해 멀티-쓰레딩(Multi-Threading) 기법을 사용하였다. 클래스로 구현된 에이전트는 물리적으로 개별 쓰레드로서 프로세서에 올라가게 된다. 에이전트들의 수축적 활동을 통제하기 위해 전역 시계(Global Clock, g_clock)의 흐름을 제어하는 컨트롤 에이전트를 구현하였다.

수축적 병렬 시뮬레이션은 <그림 7>의 객체 상호작용도(Object Interaction Diagram)[Jacobson, et al., 1993]와 같이 구현되었다. 모든 에이전트는 이벤트 핸들을 하나씩 가지게 되며 이들의 동기화를 통제하기 위한 동기화 이벤트 핸들(aSync-Event)이 생성된다. 컨트롤 에이전트는 전역 시계를 초기화하고, 반복적으로 모든 에이전트가 한번씩 활동했는가를 확인하기 위해 이벤트 핸들에 대기하게 된다. 모든 에이전트는 작업을



<그림 6> 소세계 시뮬레이션 구현 클래스 다이어그램



<그림 7> 수축적 멀티쓰레딩 구현 객체 상호작용도

<표 1> 에이전트 내부 연산

연 산	작 업
Launch()	활동시기와 활동 모드를 체크하여 활동 착수 여부 결정한다.
CheckPrecondition()	활동에 앞서 만족되어야 할 사전 조건을 확인한다. 예를 들어, 자재 출고 요청 시 작업 지시 계획 수립 여부를 확인한다.
Act()	처리 규칙에 따라 활동한다. 내부적으로 CheckPrecondition() 연산을 호출한다. 활동 결과는 해당 테이블에 반영된다. 예를 들어, 제품출고 시 부실 채권 여부에 따라 출고를 거부할 수 있으므로 반드시 여신 한도를 넘지 않았는지 확인한다.
Cancel()	작업 취소 작업을 수행한다. 예를 들어, 영업입금을 취소해야 한다면 영업입금 취소 뿐만 아니라 이에 대해 자동으로 발행된 전표에 대한 취소 작업을 한다.
Predict()	주어진 예측 지식으로 활동 결과에 따른 예측 정보를 기록한다. 예를 들어, 주문등록 후 2일 이내에 생산계획에 반영되지 않으면 생산 리드 타임은 2주가 넘게 된다는 정보를 기록한다.

한번씩 수행하면 자신의 이벤트 핸들의 상태를 변경하고 동기화 이벤트 핸들에 대기한다. 컨트롤 에이전트는 모든 이벤트 핸들의 상태가 변경되면 모든 에이전트가 한 프레임을 완료했으므로 전역 시계 값을 하나씩 증가 시키고 동기화 이벤트 핸들의 상태를 변경하여 다음 프레임을 시작한다.

동기화 이벤트 핸들의 상태가 변경되면 대기하고 있던 개별 에이전트는 지역 시계(Local Clock, L_clock)와 전역 시계 값과 비교해 보고 만일 작으면 활동하게 된다. 즉, 초기 조건을 체크하고 작업 규칙에 따라 데이터베이스 작업을 실행하는 것이다. 에이전트는 항상 동기화 이벤트 핸들의 상태 변화에 따라 활동의 착수 여부를 결정하므로 전역 시계를 제어하는 컨트롤 에이전트에 의해 통제되어 전체적으로 수축적인 에이전트 활동이 구현된다.

번호를 사슬 키로 하여 발생하는 주문별로 별도의 사슬로 기록될 수 있도록 개발 되었다. C사는 현 단계에 만족하지 않고 학습, 민활 조직으로의 전환을 위해 정보시스템에서 제공되는 주문 추적 정보를 중심으로 전 사원이 시스템 사고를 가지도록 하기 위한 기획하고 있다. 현재, 시범 운영 단계에 있는 소세계 시물레이션 과제를 통해 본 논문에서 제시된 접근 방법의 실용성을 검증하고자 한다.

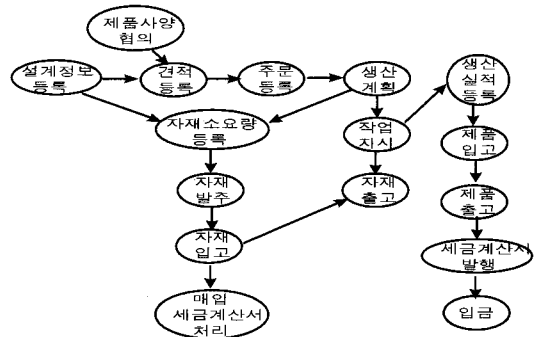
5.2 프로세스 사슬 기반 정보시스템 개발

C사는 고객의 견적 요청으로 모든 프로세스가 유발된다. 최상위 단계의 집합 프로세스 사슬은 <그림 8>과 같이 정의될 수 있다. 회계 관련 프로세스는 생략했음을 밝혀 둔다.

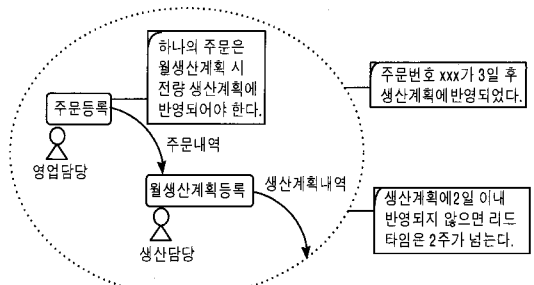
V. 사례 분석

5.1 개요

C사는 주문 생산 방식의 산업용 필터 전문 생산 업체로 견적부터 최종적인 영업 입금까지의 모든 프로세스가 고객 주문에 따라 진행되고 있다. 이와 같은 형태의 기업에 있어 주문번호에 의한 주문 추적 정보 제공과 이에 따른 적절한 의사결정은 경쟁력 제고에 필수적 역할을 가지게 된다. C사는 주문 추적과 이에 따른 전략 정보 제공을 목표로 통합 정보시스템을 개발, 운영하고 있다. C사가 채택한 방식은 철저한 프로세스 사슬화에 따른 정보시스템 개발이다. 즉, 모든 프로그램은 가능한 선행 프로세스에서 시작하여 해당 프로세스에 대한 트랜잭션 처리를 할 수 있도록 개발된 것이다. 이런 방식에 의한 개발은 선행 프로세스에서 처리되지 않은 트랜잭션이 후행 프로세스에서 발생할 수 없도록 인위적으로 막게 된다. 따라서, 주문



<그림 8> 최상위 단계의 집합 프로세스 사슬



<그림 9> 월생산계획등록 프로세스 사슬

집합 프로세스 사슬에 포함된 단위 프로세스 사슬은 정의 방법에 따라 각각 구체적으로 표기되었다. <그림 9>는 월생산계획 프로세스 사슬을 표기한 것이다. 월생산계획이 등록될 때마다 개별 주문의 생산계획 시점에 대한 정보가 발생하게 되며 이를 비즈니스 규칙에 따라 해석하게 되는 것이다. 이 경우, 생산계획에 주문 등록 후 3일만에 생산계획에 반영되었으므로 리드 타임이 2주가 넘을 것을 예상할 수 있다.

월생산계획등록 프로세스 사슬은 <그림 10>과 같이 구현되었다.

<그림 10> 월생산계획등록 프로그램

이상과 같은 프로세스 사슬의 정의를 기반으로 정보시스템을 개발하고 운영하고 있는 C사는 주문 번호를 중심으로 주문에 관련된 프로세스 처리 상태를 추적할 수 있게 되었다. 주문

번호를 알게 되면 해당 주문이 현재 어떤 프로세스까지 진행되었는지를 파악할 수 있게 된 것이다. 주문추적 시스템으로 업무 담당자의 시스템 사고를 향상시킬 수 위한 기반이 구축되었다<그림 11>. 예를 들어, 주문번호를 가지고 영업, 생산, 수급 프로세스 사슬을 포함하는 사슬에 대한 추적 정보를 조회할 수 있다. 또한, 영업, 생산, 수급의 하위 프로세스 사슬에 대한 사슬 정보도 해당 주문 번호를 더블 클릭하여 추적할 수 있다.

<그림 11> 주문 추적 프로그램

5.3 평가 측정치

C사는 프로세스 사슬에 대한 측정치를 정의하여 프로세스 수행에 대한 평가가 가능하게 되었다. 분류에 따른 측정치를 정리하면 <표 2>와 같다.

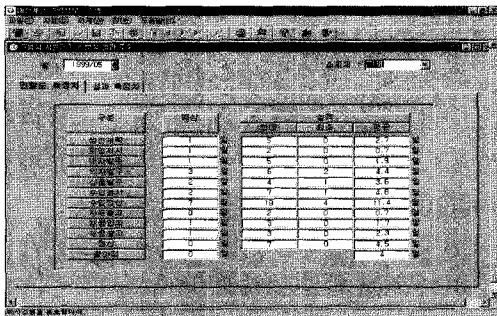
<표 2> 측정치 예

분류	소분류	항 목
민활도 측정치	지 연	생산계획, 작업지시, 내자발주, 외자발주, 상품발주, 자재수입검사, 수입정산, 자재출고, 제품입고, 제품출고, 월마감
	표 준 대비실적	내자입고일 준수도, 외자입고일 준수도, 납기일 준수도, 채권회수기일, 생산 공수기여도
성 과 측정치	달성도	매출액, 이익율, 생산량
	정확도	공정이상처리, 수입검사실시, 출하검사실시, 자재이상 처리
	손 실	자재 LOSS율, SCAN 불량율, 부실채권, 불량채권

5.4 소세계 시물레이션

5.4.1 규범적 시물레이션

규범적 시물레이션의 수행은 시물레이션 기간의 결정으로 시작된다. 현재 C사는 월 단위로 규범적 시물레이션을 수행하고 있다. 규범적 시물레이션은 정상적인 프로세스 수행을 전제로 수행되어 실적과 <표 2>의 평가 측정치를 대상으로 분석된다. <그림 12>는 규범적 시물레이션 결과에 대한 분석 요약 화면으로 생산계획 작업은 주문등록일을 기준으로 하여 1일 이내에 완료되어야 하지만 실제로, 평균 2.4일이 소요된 것을 알 수 있다.

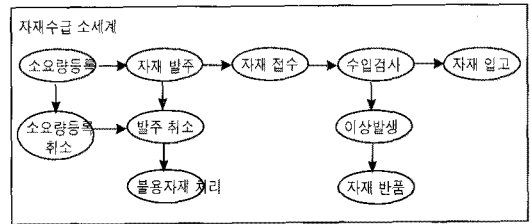


<그림 12> 규범적 시물레이션 결과

5.4.2 상황적 시물레이션

상황적 소세계 시물레이션의 1단계는 측정 대상 소세계를 선정하는 것이다. 자재수급 소세계는 <그림 13>과 같이 <소요량 등록=>자재 발주=>자재 접수=>수입검사=>자재 입고> 등 정상 사슬과 <소요량 등록 취소=> 발주 취소, 수입검사=>이상발생=>자재반품> 등 비정상 사슬로 정의할 수 있다.

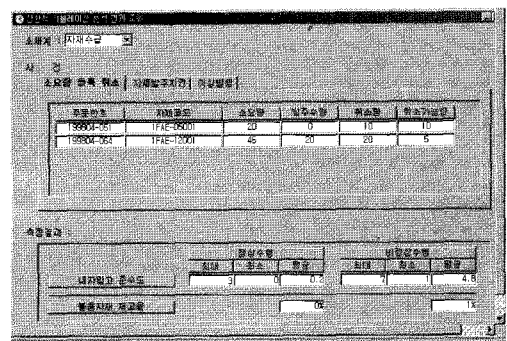
2단계에서는 결과를 측정하고자 하는 사건을 정의하고 이에 해당되는 환경 변수를 분석한다. 자재 수급 소세계의 경우, 소요량 등록 취소, 자재 발주 지연, 이상 발생 등의 사건을 전략적으로 중요한 사건을 정의하고 이에 대한 시물레이



<그림 13> 자재 수급 소세계

션을 수행하고 있다. 일단 사건이 정의되면 해당 사건에 대한 환경 변수를 결정해야 한다. 예를 들어, 특정 작업 지시에 대한 소요량 등록이 취소되고, 자재 발주 지연의 경우, 특정 자재에 대해 지연일수를 결정하며, 특정 자재에 대한 수입검사 실시 결과, 이상 발생을 등록하게 된다.

3단계에서는 측정치를 내자입고일 준수도와 불용자재 재고율로 정하였다. 4~8단계에서 사건이 발생한 경우와 사건이 발생하지 않은 경우에 대해 각각 시물레이션을 수행하면 <그림 14>와 같은 상황적 시물레이션 결과가 제시된다. 시물레이션 결과를 보면, 사건이 발생하지 않고 정상적으로 수행되었다면 내자 입고 작업은 내자 접수 후 평균 0.2일 이내에 완료되었지만, 사건이 발생한 경우, 평균 4.8일이 소요된 것을 알 수 있다. 또한, 불용자재 재고율도 정상적인 경우, 0%이지만, 소요량 등록 취소와 자재 발주 지연으로 약 1%로 상승한 것을 알 수 있다. 더 높은 불용 자재 재고율이 예상되었지



<그림 14> 상황적 시물레이션 결과

만 소요량 등록 취소 자재 중 자재 발주가 지연된 것이 있어 등록 취소 효과가 반감되었기 때문에 실제로 낮은 재고율이 나온 것이다.

VI. 결론 및 향후 연구과제

기업은 최대 자산으로 부각되고 있는 지식의 축적에 보다 많은 관심과 투자를 아끼지 않고 있다. 학습 조직이나 민활 조직은 일련의 지식 축적과 향상을 목적으로 하고 있다. 본 논문에서는 학습, 민활 조직으로의 전환을 실현하기 위한 프로세스 사슬 기반 정보시스템 개발과 소세계 시뮬레이션 방법을 소개하였다.

미래 기업은 보다 많은 프로세스를 정보기술로 처리하게 될 것이다. 정보시스템에 의한 프로세스 수행은 대량의 데이터의 축적을 낳는다. 따라서, 정보시스템에 의한 프로세스 처리는 학습, 민활 조직으로의 전환을 위한 기반 구조를 조성한 것이라고 볼 수 있다. 보다 많은 프로세스가 정보시스템으로 처리된다면 우리는 복잡한 프로세스 수행 과정과 결과를 컴퓨터로 시뮬레이션 할 수 있게 된다. 이렇게 된다면 정보기술이 학습, 민활 조직으로의 전환을 가능하게 하는 기반이 될 수 있는 것이다.

본 논문은 정보기술에 의한 프로세스 자동화가 심화되어 가고 있는 기업의 기술적 환경 변화를 전제로 할 때 정보기술의 효과를 극대화할 수 있는 방법을 제시했다고 볼 수 있다. 소세계 시뮬레이션은 과거와 미래에 대한 프로세스 수행 과정과 결과를 관측할 수 있도록 함으로써 프로세스 담당자의 시스템 지식을 향상시킬 수 있는 방법이다.

C사는 프로세스 사슬에 의한 정보시스템 개발로 초기화 작업에 많은 어려움을 겪었으나 프로세스의 연계가 시스템화 되어 근거 없는 데이터의 입력이나 조작이 원초적으로 불가능하게 됨으로써 보다 정확하고 일관성 있는 데이터 축적이 가능하게 되었다. 또한, 프로세스 사슬에

따른 각종 추적 정보는 전체 프로세스의 진행 상황을 파악하게 하여 학습 조직에서 지향하는 시스템 사고, 사실에 입각한 학습, 세계관의 확장을 실현할 수 있는 기반을 제공하였다. 또한, 평가 측정치에 의한 소세계 시뮬레이션으로 보다 적극적인 지식 학습을 기대하고 있다.

그러나, 소세계 시뮬레이션의 실용적 구현과 운영에 대해서는 아직도 해결해야 할 문제가 남아 있다. 첫째, 프로세스의 규범적, 상황적 시뮬레이션 수행 결과에 대한 평가 방법이 명확하지 않다는 것이다. 이것은 소세계 시뮬레이션 시스템의 운영에 대한 전례가 없기 때문이다. 따라서, 평가 측정치에 대한 지속적인 검증과 이에 대한 보완이 요구된다. 둘째, 시뮬레이션의 구현 기술의 보완이다. 멀티 쓰레딩을 이용한 시스템 개발 사례가 전무한 상황에서 적절한 시스템 구조 설계나 멀티 프로세서의 효과적인 사용에 대한 운영체계의 지원 등이 요구되는 것이다. 그러나, 향후, 보다 많은 프로세서의 활용을 지원할 수 있는 운영체계의 개발이 예상되므로 오히려, 보다 다양한 기술적 지원을 받을 수 있을 것이다. 셋째, 시뮬레이션의 필요성과 사용 방법에 대한 프로세스 담당자의 무관심이다. C사의 경우, 프로세스 사슬에 기반하여 개발된 정보시스템을 운영하는 시행 초기에 있어 아직 시뮬레이션에 대한 명확한 기대나 사용 방법에 대해 준비되지 못한 상태에 있다. 현재 규범적 시뮬레이션과 몇 개의 소세계에 대한 제한적 상황적 시뮬레이션만을 사용하고 있다.

프로세스 사슬 템플릿 기반 정보시스템 개발은 통합 정보시스템 개발과 효과적 운영을 위한 제도의 시스템화를 실현할 수 있는 방법론으로 실증되었다. 그러나, 소세계 시뮬레이션은 아직 규범적 시뮬레이션 중심의 실험적 단계에 머물고 있으므로 지속적인 적용에 의한 방법론의 향상으로 기업 지식의 효과적 축적이나 향상을 위한 지식 경영 방법론으로 정착될 수 있도록 해야 할 것이다.

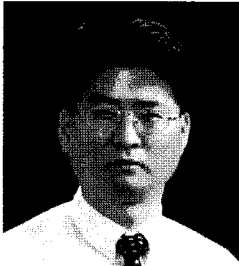
〈참 고 문 헌〉

- [1] 노나카, 지식 창조의 경영, 21세기북스, 1990.
- [2] 이순철, 신경영기법, 매일경제신문사, 1996.
- [3] 이장우, 박형규, "복잡성 이론과 기업 경영: 프랙탈 경영방식을 중심으로", 경영과학, 제15권 제 2호, 1998, pp. 239-257.
- [4] 조운호, 김재경, 김성희, "Role 개념에 근거한 비즈니스 프로세스 시뮬레이션 모형 구축 및 분석", 경영정보학 연구, 제8권 제2호, 1998, pp. 69-83.
- [5] Argyris, C., "Double Loop Learning in Organization", *Harvard Business Review*, September-October, 1977, pp. 115-125.
- [6] Augustine, E., Surynt, F., Dezoort, E., and Rosetti, D., "Organizational Impact of Decision Support Technology: Whats Ahead for the 90's?", *Journal of End User Computing*, Vol. 5, No. 2, 1993, pp. 26-30.
- [7] Beck, M., "Learning Organizations: How to Create Them", *Industrial & Commercial Training*, Spring, 1992, pp. 21-28.
- [8] Briccarello, P., Bruno, G. and Ronco, E., "REBUS: An Object-Oriented Simulator for Business Process", *Proceedings of the 28th Annual Simulation Sumposium*, 1995, pp. 269-277.
- [9] Beck, N., *Shifting Gears: Thriving in the New Economy*, HarperCollins, Toronto, 1992.
- [10] Camerer, C.F. and Johnson, E. J., "The Process-Performance Paradox in Expert Judgement", *Toward a General Theory of Expertise*, Ericsson, K.A. and J. Simth, Eds, Cambridge University Press, Cambridge, 1991, pp. 195-217.
- [11] Davenport, T. H., *Process Innovation*, Harvard Business Press, Boston, 1993.
- [12] Davenport, T.H. and Stoddard, D.B., "Reengineering: Business Change of Mythic Proportions?", *MIS Quarterly*, Vol. 18, No. 2, 1994, pp. 121-127.
- [13] Drucker, P. F., *The New Realities*, Harper & Row, New York, NY, 1989.
- [14] Drucker, P. F., "Professionals Productivity", *Across the Board*, Nov/Dec, Vol. 30, No. 9, 1993, p. 50.
- [15] Durbin, R., "Theory Building in Applied Areas", *Handbook of Industrial and Organizational Psychology*, M.D. Dunnette Ed, Wiley, New York, NY, 1976, pp. 17-39.
- [16] Eddins, W. and Sutherland II, D., "Using Modeling and Simulation in the Analysis and Design of Information Systems", *Dynamic Modeling of Information Systems*, Elsevier Science Publishers, 1991, pp. 61-88.
- [17] Fiol, C. M. and Lyles, M. A., "Organizational Learning", *Academy of Management Review*, October, 1985, pp. 803-813.
- [18] Garvin, D. A., "Building a Learning Organization", *Harvard Business Review*, July-August, 1993, pp. 78-91.
- [19] Gleck, J., *Chaos Making a New Science*, Penguin Books, New York, 1987.
- [20] Grimes, R., Stockton, A., Reilly, G., and Templeman, J., *Beginning ATL COM Programming*, Wrox Press Ltd., Olton, Birmingham, UK, 1998.
- [21] Gunneson, A. O., *Transitioning to Agility*, Addison-Wesley, Reading, MA, 1997.
- [22] Hammer, M. Champy, J., *Reengineering the*

- Corporation, Harper Business, New York, NY, 1993.
- [23] Hickman, C. R., *The Organization Game: An Interactive Business Game Where You Make or Break the Company*, Prentice Hall, Englewood Cliffs, CA, 1995.
- [24] Hickman, C. R., *The Strategy Game*, Prentice Hall, Englewood Cliffs, CA, 1994.
- [25] Jacobson, I. et al., *Object-Oriented Software Engineering*, Addison-Wesley, New York, NY, 1993.
- [26] Kock N.F., McQueen, R. J., and Baker, M., "Learning and Process Improvement in Knowledge Organizations: A Critical Analysis of Four Contemporary Myths", *The Learning Organization*, Vol. 3, No. 1, 1996, pp. 31-41.
- [27] Kock, N.F., McQueen, R. J., and Corner, J.L., "The Nature of Data, Information, and Knowledge Exchanges in Business Processes: Implications for Process Improvement and Organizational Learning", *The Learning Organization*, Vol. 4, No. 2, 1997, pp. 70-80.
- [28] Labrou, Y. and Finin, T., "Semantics and Conversations for an Agent Communication Language", *Proceedings of the Fifteenth International Conference on Artificial Intelligence*, 1997, pp. 584-591.
- [29] Licker, P., *Management Information Systems A Strategic Leadership Approach*, The Dryden Press, Orlando, FL, 1997.
- [30] Newstrom, J. and Scannell, E., *The Big Book of Business Games: Icebreakers, Creativity Exercises and Meeting Energizers*, McGraw-Hill, New York, NY, 1996.
- [31] Pedler M. J. et al., *The Learning Company: A Strategy for Sustainable Development*, McGraw-Hill, New York, NY, 1991.
- [32] Senge, P., *The Fifth Discipline*, Doubleday, New York, NY, 1990.
- [33] Taylor, *Object-Oriented Information Systems Planning and Implementation*, John Wiley & Sons, New York, NY, 1992.
- [34] Taha, H., *Operations Research An Introduction*, Macmillan, New York, NY, 1987.
- [35] Warren, J. and Stott, J., "CASE/SIMULATION: Making Performance Evaluation a Normal Part of Information Systems Development", *Dynamic Modeling of Information Systems*, Elsevier Science Publishers, 1992, pp. 219-244.

◆ 이 논문은 1999년 7월 21일 접수하여 1차 수정을 거쳐 1999년 9월 14일 게재 확정되었습니다.

◆ 저자소개 ◆



박광호 (Park, Kwang-Ho)

저자는 한양대학교 경영학과를 졸업하고 University of Iowa에서 경영학석사(MBA), 경영학박사(전공: MIS) 학위를 취득하였으며, 현재 한양대학교 경상대학 경영학부 부교수로 재직 중이다. 삼성 SDS에서 AI실장, 소프트웨어연구팀장 등을 역임하며 객체지향시스템, 객체지향기술 관련 프로젝트를 수행한 바 있다. 최근에는 객체지향 비즈니스 엔지니어링 프로젝트를 통해 프로세스의 표준화, 혁신, 이에 따른 정보시스템 개발을 추진 중이다. Naval Research Logistic, International Journal of Production Research, International Journal of Manufacturing System Design, 경영정보학연구, 데이터베이스 저널, 정보시스템 연구 등에 논문을 게재하였다.