

RTE 환경에서의 SOA 기반 협업적 분산 에이전트 개발*

최형림** · 김현수*** · 홍순구*** · 박영재**** · 최성욱***** · 조민제***** ·
박창현***** · 강시협***** · 안병선*****

〈 목 차 〉

I. 서론	3.2 서비스별 알고리즘
II. 선행연구	3.3 BAM 대시보드 구현
2.1 공급망관리와 실시간 기업	IV. 알고리즘 실험
2.2 에이전트와 서비스지향아키텍처	V. 결론
III. 에이전트 구현	참고문헌
3.1 서비스지향아키텍처 기반 에이전트 설계	<Abstract>

I. 서론

현대의 공급망은 연관된 기업의 수가 더욱 많아지고 글로벌화 되고 있으며 과거 폐쇄적이고 정적인 것에서 개방적이고 동적인 환경으로 변화되고 있다. 이러한 환경에서 효과적인 공급망관리(SCM: Supply Chain Management)는 수요변화에 시기적절하고 비용효과적인 방법으로 대응할 수 있는 능력이 중요하며, 이는 기업 경

쟁력의 핵심이 될 수 있다. 현재까지의 공급망 관리는 지속적인 거래 파트너와의 장기 계약에 따른 정적인 관계가 일반적이었다. 하지만 수요 변화에 대응하기 위한 공급망을 구축하기 위해서는 보다 유연하고 동적인 능력과 기법이 필요하지만, 이에 대한 실증적인 연구가 미비한 실정이다.

동적 공급망관리를 위해서는 다양한 형태의 시스템들이 이종의 환경에서도 제약 없이 자유

* 이 논문은 2006년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임"(KRF-2006-042-B00060)

** 동아대학교 경영정보학과, 교수, 주저자, hrchoi@dau.ac.kr

*** 동아대학교 경영정보학과, 교수, hskim@dau.ac.kr, shong@dau.ac.kr

**** 동명대학교 경영정보학과, 교수, yjpark@tu.ac.kr

***** 동아대학교 경영정보학과, 박사과정, 교신저자, mjcho78@gmail.com

***** 부산인적자원개발원, 선임연구원, 8core@bhrdi.or.kr

***** 동아대학교 경영정보학과, 석사과정, archehyun@naver.com

***** 씨아이이에스(주), 선임연구원, shkang@cies.co.kr

***** (주)넷솔루트, 이사, bsann@netsolute.net

롭게 거래를 지원할 수 있어야 하며, 예측된 환경이 아닌 역동적인 시장 상황에서 당면한 위기과 기회를 신속하게 인지하여 민첩하게 대응할 수 있는 환경이 제공 되어야 한다(Kumar, 2001).

최근 변화하는 환경에 실시간으로 대응하고자 하는 실시간기업(RTE: Real-Time Enterprise)의 개념이 확대되고 있다(Mcgee, 2004; Raskino, 2004). 이러한 실시간기업은 개별기업이 아닌 공급망 전체의 실시간화를 요구하고 있다. 따라서 기업의 실시간화 즉, 실시간기업이 되기 위해서는 공급망관리가 동적이고 실시간화 되어야 한다. 그러나 공급망 관점에서 실시간화를 위한 통합 및 최적화는 다양하고 복잡한 문제를 해결해야 한다. 즉, 시장과 고객의 요구 변화에 대해 신속한 의사결정과 민첩한 대응이 필요하며 조직과 조직 간의 긴밀한 정보와 지식공유 및 업무협조가 요구된다.

본 연구에서는 이러한 문제점을 해결하기 위해 이전과 같이 기업에 속한 관리자 즉, 사람에 의해 경영활동들이 이루어지고, 해당 직무자와 관련된 정보만이 관리되는 방식이 아닌 에이전트와 같은 지능형 기술을 적용하고자 한다. 또한 지능형 에이전트를 구현함에 있어 서비스지향아키텍처(SOA: Service Oriented Architecture) 기반의 멀티 에이전트시스템을 제안한다. 1장에서는 본 연구의 배경과 목적을 기술하고, 2장에서 공급망관리와 실시간 기업의 개념과 변화, 그리고 에이전트와 서비스지향아키텍처에 대해 선행 연구를 실시하였다. 3장에서는 에이전트 구현을 위하여 서비스지향아키텍처 기반으로 에이전트를 설계하고, 서비스를 개발하여 그 전략 및 실험 환경을 설명하고 이를 모니터링 할 수 있는 대시

보드를 제시하였다. 4장에서는 제시된 실험환경에서 개발된 알고리즘을 실험 및 검증하였고, 마지막으로 5장에서는 본 연구의 결과 및 기대효과를 제시하고, 향후 연구 방향을 설명하였다.

II. 선행연구

2.1 공급망관리와 실시간 기업

전통적인 공급망관리에서 공급망의 최적화는 제품, 서비스 그리고 정보의 흐름을 계획, 실행 및 통제하는 과정에서 이루어 질 수 있다고 보았다(Lambert, 1998). 하지만 이는 물류와 정보 흐름의 차이로 인하여 정보의 지연 및 왜곡이 발생하여 예측의 정확성이 떨어진다. e-SCM은 표준화된 비즈니스 프로세스의 등장을 요구하며, 비즈니스간 인터페이스 기준 개발 및 정보 기술의 적극적인 활용을 통하여 전통적인 공급망관리의 한계를 극복하고자한 것이다. 즉 e-SCM은 원자재 조달, 생산, 수배송, 판매 및 고객관리 프로세스에서 일어나는 물류와 이와 관련된 모든 활동을 통합적으로 관리하는 기법을 말하며, 이때 이러한 관리를 정보기술에 기반하여 실시간으로 신속하고 효율적으로 처리하는 것을 말한다(신광선, 2002). e-SCM과 관련하여 기존 연구들에서는 구축 또는 성공 전략, 부분적인 모듈 또는 프레임워크 개발 등의 분야에서 연구들이 진행되었다. 그러나 e-SCM의 개념을 제대로 지원하는 솔루션 또는 사례는 찾기는 힘든 실정이다. 기존 e-SCM 시스템들은 각 부문별로 구축 되어, 시스템의 유연성이 떨어지며, 비용부담이 크고, 확장 및 최적화에 대

인 공급망관리의 한계를 극복하고자한 것이다. 즉 e-SCM은 원자재 조달, 생산, 수배송, 판매 및 고객관리 프로세스에서 일어나는 물류와 이와 관련된 모든 활동을 통합적으로 관리하는 기법을 말하며, 이때 이러한 관리를 정보기술에 기반하여 실시간으로 신속하고 효율적으로 처리하는 것을 말한다(신광선, 2002). e-SCM과 관련하여 기존 연구들에서는 구축 또는 성공 전략, 부분적인 모듈 또는 프레임워크 개발 등의 분야에서 연구들이 진행되었다. 그러나 e-SCM의 개념을 제대로 지원하는 솔루션 또는 사례는 찾기는 힘든 실정이다. 기존 e-SCM 시스템들은 각 부문별로 구축 되어, 시스템의 유연성이 떨어지며, 비용부담이 크고, 확장 및 최적화에 대한 어려움이 있다. 즉 기존의 공급망관리에서는 실시간이라는 개념을 제대로 반영하지 못하고 있는 것이다.

실시간이라는 개념은 어떤 일을 진행함에 있어 지연요소를 모두 제거시킨 상태로 정의되고 있다. 또한 일반적으로 생각하듯이 어떤 이벤트가 발생했을 때 즉각 전달되어 처리 하는 것을 실시간이라 할 수도 있으며, 필요할 때에 감지되고 처리가 가능하다면 이 또한 실시간으로 정의할 수 있다(이명환 외, 2005).

이명환(2005)은 비즈니스 모델을 <표 1>과 같이 전통적 모델과 실시간 모델로 비교하고 있다.

위기와 기회에 신속하게 대응하는 것에 대한 논의는 이전부터 제기되어 왔으며, 이는 기업의 하부조직에서부터 실시간으로 공유되는 경영환경을 구축하기 위한 방법론이다. 즉, 기업의 주요 업무 프로세스에서 발생하는 정보를 필요한 적재적소에 실시간으로 전달하여 즉각적인 모니터링과 신속한 대응이 가능한 기업으로 탈바꿈하는 것이다. 급변하는 경영환경에서 얼마나 빨리 환경변화를 인지하고 고객의 요구사항에 신속하게 대응할 수 있는지가 기업 경쟁력 강화의 요체로 인식되어, 상황인식, 적절한 대안마련, 대처(실행)라는 위기관리 및 대응 사이클의 속도를 높이는 것이 실시간기업의 핵심 개념이라 할 수 있다. 그러나 서술한 바와 같이 더 이상 실시간의 개념은 단일 기업의 개념이 아닌 전체 공급망을 하나의 조직으로 보고 실시간화해야 한다(Mcgee, 2004). 이는 공급망에서 종단간(end-to-end) 프로세스들의 사이클을 최대한 단축하고, 실시간화하는 개념으로 이러한 자동화를 위해서는 상황인식, 대안분석, 최적대안 제시, 반응 등에 에이전트와 같은 지능형 기술이 필요하다.

<표 1> 전통적 모델과 실시간 모델의 비교

전통적 모델	실시간 모델
- 과거의 성과에 대한 보고	- 실시간으로 문제 및 원인 발견
- 보고서를 읽고 이슈를 발견하기까지 조치를 취할 수 없음	- 관련되어 있는 사람에게 즉각적으로 공지사항을 전파
- 조치가 시스템 외부 예컨대 전화나 메모 등을 통해 수행됨	- 온라인에 의한 문제 전달 및 즉각적인 조치 강구
- 예외사항/근본원인 및 솔루션에 대한 누적 데이터 부재	- 모든 예외사항/근본원인 및 솔루션에 대한 데이터베이스 보유
- 예외사항을 추적할 수 있는 공식적인 시스템이 없어 조직학습을 촉발하지 못함	- 이해관계자에 대한 교정조치의 자동 전달로 조직학습을 촉발함

<참고문헌: 실시간기업(RTE), 한국기업문화연구소, 2005.>

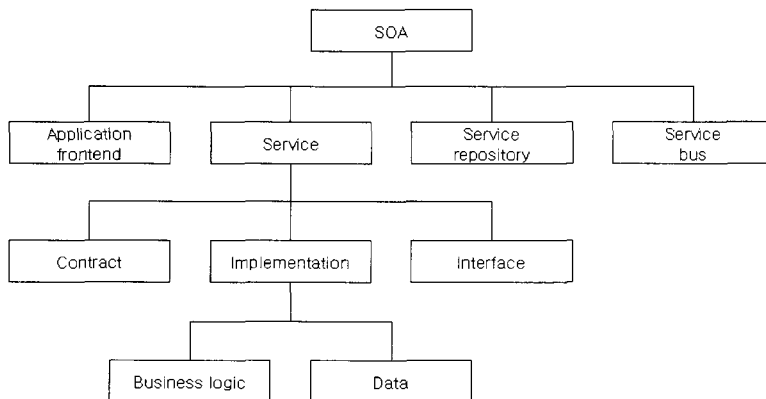
2.2 에이전트와 서비스지향아키텍처

멀티 에이전트는 여러 에이전트간의 협동에 의해서 복잡한 문제를 해결하고자 하는 시스템을 말한다. 물론 하나의 도메인에 관련된 작업처리에는 단일 에이전트가 효과적이지만, 여러 도메인을 복합적으로 고려해야 하는 문제에는 한계가 있다. 따라서 더욱 복잡해지고 속도를 요구하는 현대 비즈니스 환경에서는 여러 에이전트간의 협동에 의해서 문제를 해결하는 것이 보다 효율적이다(Chen et al., 1999). 그러나 멀티 에이전트시스템 개발은 이형질성(heterogeneity)과 분산성(distribution)에 대한 문제점이 지속적으로 지적되어 왔다(Jennings et al., 1991). 멀티 에이전트의 구현 형태는 고립된 단독 시스템 환경이 아니라, 복잡한 이기종 환경에서 상호의존하고 있는 시스템이다. 이런 상황에서 다른 시기에 다른 방법으로 다른 시스템 환경에서 구축된 시스템은 이러한 문제가 시스템의 한계가 될 수 있다. 이와 같은 문제를 해결하기 위한 노력이 현재까지의 멀티 에이전트의 주요 연구와 진화 및 변천 과정이라고도 할 수 있다.

서비스 지향의 개념은 일반 엔터프라이즈 시

스템뿐만 아니라 다양한 분야의 시스템에 적용이 가능하다. 특히 변화관리와 통신이 주요한 역할을 차지하는 멀티 에이전트시스템 같은 경우는 더욱 SOA의 적용이 필요하다. 서비스지향의 개념은 복잡성 문제의 해결 요구와 커다란 문제 해결에 필요한 로직들을 관련된 조각으로 좀 더 작게 분할하면, 더 잘 구축되고 실행되며, 관리될 수 있다는 관심의 분리(separating concern)라는 개념에서 출발하였다(Erl, 2006). SOA에 관한 대부분의 정의는 서비스들의 결합을 포함하는 분산 소프트웨어 아키텍처 또는 프레임워크로 정의되며, SOA는 비즈니스 기능들을 네트워크 상에서 재사용이 가능한 공유서비스의 집합으로 구현한 소프트웨어 설계 패러다임이라고 할 수 있다. Krafzig는 SOA의 구성을 <그림 1>과 같이 애플리케이션 프론트엔드(Application Frontend), 서비스(Service), 서비스 리파지토리(Service Repository), 서비스 버스(Service Bus)의 4가지 주요 추상적 개념을 바탕으로 구성하였다(Krafzig et al., 2004).

이중 서비스는 명확한 기능적 의미를 지닌 소프트웨어 컴포넌트로, 비즈니스 개념을 캡슐화하고 있으며, 이를 제외한 요소들은 서비스를 받



<그림 1> SOA의 구성요소(출처: Krafzig et al., Enterprise SOA, 2004)

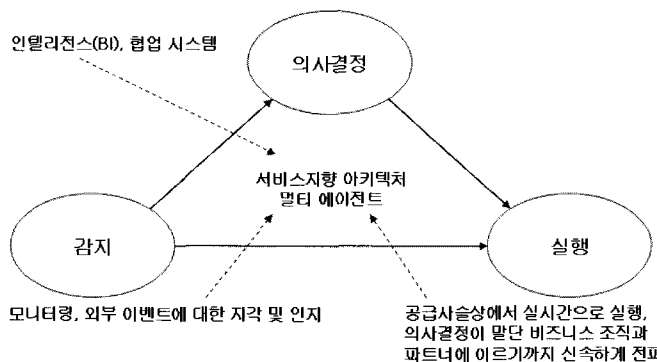
견, 조합 그리고 사용하기 위한 기반이 된다. 즉 서비스를 제외한 요소들은 시스템의 기본 구조적 성격을 가지고 있으며, 서비스는 실제 소프트웨어 컴포넌트의 형태이다. 서비스라는 개념은 보통 서비스를 호출하는 애플리케이션이 반드시 동일한 물리적인 기계에 있을 필요가 없다는 점을 가정하고, 어떤 컴퓨터 프로그램이 다른 컴퓨터 프로그램의 요청(Request)에 따라 수행하는 의미 있는 움직임을 의미한다.

결국 실시간기업의 시스템 아키텍처에 관한 기본 원리는 비즈니스 프로세스 내의 모든 개체들과 절차들을 총체적으로 바라보는 것이며, 이를 기반으로 경영의사결정에서의 시간지체를 해소하고, 특정 애플리케이션시스템에서 발생한 정보를 관련된 모든 곳으로 전달할 수 있어야 한다(이명환 외, 2005). 멀티 에이전트시스템에서 외부 에이전트의 연결문제는 결국 이기종 시스템과 통신 및 데이터 문제가 되고, 이를 극복하더라도 빠르고 정확한 자동화 처리가 요구된다. 또한 기존 에이전트들은 그 구조상 변경사항의 발생 및 이기종 시스템과의 결합시 전체 에이전트 시스템에 영향을 주게 된다. 하지만 에이전트시스템의 구현에 서비스지향아키텍처의 개념을 적용하면 변경사항 발생 및 이기종 시스템 결합시

해당 부분만을 대상으로 서비스를 재조립하여 사용할 수 있으므로 이러한 문제를 해결할 수 있다.

의사결정에 있어서도 다양한 정보의 수집 및 관리와 협업을 위한 시스템 구조가 필요하다. 뿐만 아니라 비즈니스 실행을 위한 빠른 정보전달 또한 필요하다. 이러한 요구사항들이 Gassman(2003)이 언급하였던 실시간 비즈니스 아키텍처를 구성하게 되는 이유가 되며, Gassman의 비즈니스 아키텍처에 비스 기반의 멀티 에이전트를 적용한다면 <그림 2>와 같이 실시간 비즈니스가 실현 가능하게 된다(Gaussman, 2003).

<그림 2>에서 감지와 같은 모니터링은 RTE를 추진하기 위한 요소들 중 하나인 BAM(Business Activity Monitoring)과 밀접한 연관성을 가진다. 이는 비즈니스 운영에 있어서의 속도와 효과를 향상하기 위한 정보를 지원하는 동시에 핵심적인 비즈니스 성과지표에 대한 실시간 접근을 제공하는 것으로 정의 된다. 지금까지는 핵심 이벤트를 실시간으로 통합 모니터링 하는 개념이 정착되지 않고, 개별 시스템 또는 단위 업무별로 모니터링이 수행되어 왔다. BAM은 다수의 시스템을 모니터링하고, 실시간 대시보드를 구축하여 사전에 정의된 상황의 발생을 감지해내는 연계 모니터링이 가능하다. 이때 BAM은 정보뿐만



<그림 2> 실시간 비즈니스 아키텍처에 멀티 에이전트의 적용

아니라, 경영활동을 시각적인 도표를 활용하여 관리자에게 보여주는 대시보드 기술을 활용할 수 있다.

대시보드는 일반적으로 KPI(Key Performance Indicator)로 표현되는 비즈니스 상황과 관련 정보를 가지는 개인화된 사용자 인터페이스의 한 형식으로 정의된다. 결국 대시보드를 활용하면 경영현황을 파악하고 시장 환경의 변화를 실시간으로 감지하며 내외부적인 변화가 미치는 영향을 실시간으로 감지할 수 있다. 특히 기업 내부의 주요 성과지표와 기업 외부의 비즈니스 환경 요인에 대한 거시적인 정보를 실시간으로 수집하고 보여준다. 또한 추진하고 있는 사안들에 대한 미시적인 정보도 실시간으로 제공한다 (Caroline, 1998).

Ⅲ. 에이전트 구현

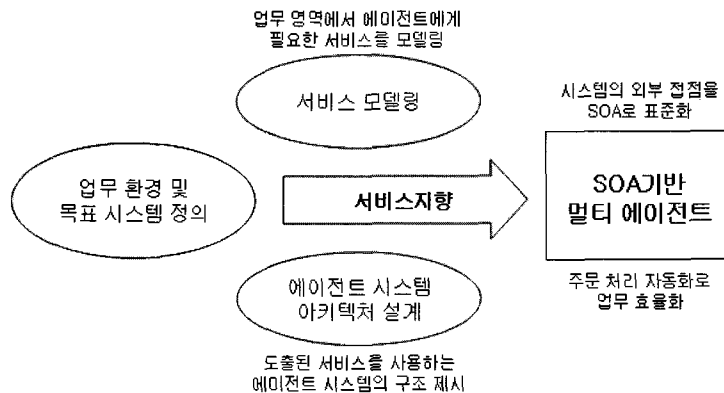
3.1 서비스지향아키텍처 기반 에이전트 설계

시스템 유연성 확보, 복잡도 감소, 그리고 비즈니스 자동화 문제점들을 해결하기 위한 에이

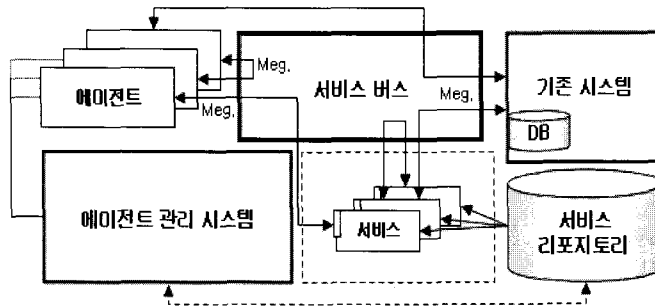
전트 시스템은 업무 담당자의 역할을 멀티 에이전트로 대체하고, SOA 개념을 적용하여 자동화 및 전략 문제를 서비스화 하고 이들 서비스를 에이전트들이 사용한다.

이를 위해 <그림 3>과 같이 정의된 업무 환경 및 목표 시스템에 따라서 먼저 주어진 업무 영역에서 필요한 서비스를 도출하기 위한 모델링 과정을 거쳐야한다. 그리고 전체 시스템에서 도출된 서비스를 사용하는 에이전트시스템의 구조를 제시한다. 이러한 시스템의 궁극적인 목표는 시스템의 외부 접점을 SOA로 표준화하여 이질적 환경을 극복하고, 업무 처리를 자동화하여 효율성을 높이는 것이다.

본 연구에서는 에이전트의 역할을 상태 및 타 에이전트와의 통신만을 유지하고, 실제 업무는 서비스를 사용하여 수행하는 멀티 에이전트 시스템인 SOMAS(Service Oriented Multi-Agent System)를 제안한다. SOMAS는 에이전트가 필요로 하는 기능에 해당되는 서비스를 호출하여 업무를 수행하며, 업무 및 역할 변경 시 필요한 다른 서비스를 호출하여 사용할 수 있다. 또한 서비스 버스를 통한 통신방법을 사용함으로써 시스템 통합 시 발생하는 문제들을 사전에 방지



<그림 3> SOA기반의 멀티 에이전트시스템



<그림 4> 서비스지향 멀티 에이전트 시스템(SOMAS)의 구조

할 수 있다.

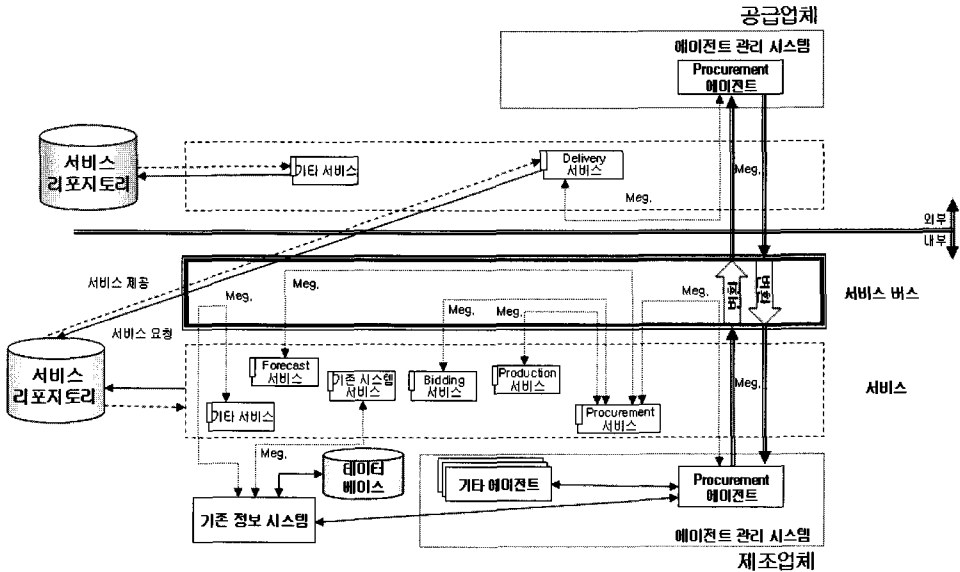
SOMAS의 구조는 <그림 4>와 같다. 기존 시스템과 데이터베이스는 그대로 유지 될 수 있으며, 기존 시스템에서 필요로 하는 서비스를 호출하여 사용할 수도 있다. 또한 에이전트의 요청에 따른 업무를 수행할 수도 있다. 이때 기존 시스템과 서비스, 기존 시스템과 에이전트간의 기능 호출은 서비스 버스를 통하여 이루어진다.

서비스는 에이전트에 의해 사용되며 주어진 업무와 시스템 목표에 의해 정의되었다. 이렇게 정의된 서비스는 <그림 5>와 같이 사용되어질 수 있다. <그림 5>에서는 이전에 제시하였던 업무영역에서와 같이 제조업체를 중심으로 각 구성요소들과 서비스들의 관계를 보여준다. 제조업체를 중심으로 보았을 때, Procurement 에이전트들은 공급업체의 에이전트와 서비스 버스를 통하여 서로 메시지를 교환하며 주문과 이에 대한 응답을 할 수 있다. 서비스 버스는 외부와의 통신뿐만 아니라 내부 에이전트들간의 통신 및 기존 정보시스템과의 통신에도 이용되어 각각의 에이전트들이 이기종 또는 다른 특성을 가지고 있더라도 이들 상호간의 통신과 메시지 전달을 보장해주게 된다. 제조업체는 각 에이전트의 서비스를 서비스 리포지토리를 통하여 공개하며, 거래를 원하는 공급업체는 거래와 관련된 업무

를 처리하기 위해 제조업체가 제공하는 서비스 리포지토리를 통하여 에이전트의 업무 서비스를 가져올 수 있다.

SOMAS에서 제공되는 서비스들은 다음과 같다.

- (1) Forecast 서비스: 단순회귀분석을 기반으로 사용자의 향후 수요를 예측하고, Procurement 서비스에게 구매를 요청한다.
- (2) Procurement 서비스: 예측된 수요를 충족시키기 위해 공급업체의 에이전트와 거래하여 필요한 부품을 구매한다. 공급업체의 에이전트는 구매자와 거래를 위해 구매자가 제공하거나, 공개된 서비스 리포지토리에서 필요한 Procurement 서비스를 제공받아 사용할 수 있다.
- (3) Production 서비스: 부품이 납입되면 납입된 부품정보를 바탕으로 일정계획을 수립하고 이를 바탕으로 ATP(Available-To-Promise) 정보를 생성한다.
- (4) Bidding 서비스: Production 서비스의 ATP(Available-To-Promise) 정보를 기반으로 고객의 RFQ에 대응하여 Offer를 한다.
- (5) Delivery 서비스: 고객으로부터 주문이 들어오면 납기일 내에 배달할 수 있도록 Delivery 일정을 수립한다. Delivery 서비스 역시 Procurement 서비스와 마찬가지로



<그림 5> SOMAS의 서비스 사용 개념도

로 서비스를 제공받아 사용할 수 있다.

3.2 서비스지향아키텍처 기반 에이전트의 적용

3.2.1 TAC-SCM 개요

현재 SOA에 대한 산업 표준이나 서비스 지향 원칙이 국제적으로 인정된 것이 없어 비즈니스 서비스를 모델링하는 표준화된 방법론은 없다. 그러나 서비스지향 솔루션 프로젝트를 진행함에 있어 고려해야 할 부분은 분석 및 설계 단계에서부터 SOA의 특징과 서비스지향 개념이 반영되어 있어야 한다(Carriere, 2005). 본 연구에서는 서비스 지향적인 비즈니스 프로세스를 만들고, 서비스를 생성하기 위해 TAC-SCM (Trading Agent Competition Supply Chain Management)의 게임 시나리오를 기반으로 하여 서비스를 모델링하고 각 서비스에서 필요로 하는 알고리즘을 설계하였다. TAC-SCM은 간단한 게임 규칙이지만 동적 공급사슬에 많은 참

여자들이 충분한 경쟁 속에서 다양한 변화를 실증적으로 실험할 수 있는 기회를 제공하기 위해 설계되었다. TAC-SCM은 자동화된 전략을 지닌 독립적인 에이전트를 구현하여 다른 에이전트들과 경쟁하는 다양한 모의실험을 통해 공급사슬관리를 위한 에이전트를 연구할 수 있는 유일한 테스트베드를 제공하고 있다. TAC-SCM 게임 시나리오와 규칙에 관한 보다 자세한 사항은 사이트(http://www.cs.cmu.edu/~sadeh/TAC_SCM_Challenges.htm, 2008)의 공식 게임 규칙서를 참조할 수 있다(Collins, 2006).

TAC-SCM은 고객-제조업체(PC조립업자)-공급자(PC 부품 제조 및 공급자) 3단계의 공급사슬환경으로 구성되어 있다. 6개의 제조업체들이 게임에 동시에 참여할 수 있으며 이들은 각각 에이전트로 구현된다. 따라서 본 연구에서는 개발된 에이전트들을 TAC-SCM의 에이전트 엔진을 사용하여 구동되도록 하였으며, 적용 환경 역시 TAC-SCM에서 가정된 시나리오와 동일하도

록 정의하였다.

3.2.2 TAC-SCM 시나리오 및 게임 규칙

TAC-SCM의 시나리오는 다음과 같다. 고객은 제조업체 즉, 에이전트들에게 구매하고자 하는 PC의 유형, 수량, 구매가격, 납기일의 내용이 담긴 RFQ(Request For Qualifications)를 보낸다. RFQ를 받은 에이전트들은 고객으로부터 받은 RFQ중 원하는 RFQ에 대해 판매가격을 제시한다. 이때 에이전트는 고객이 제시한 PC유형, 수량, 납기일은 반드시 만족시켜야 하며, 고객이 제시한 구매가격을 초과하는 판매가격을 제시할 수 없다. 고객은 6개의 에이전트들이 제시한 계약조건들 중 최소가격을 제시한 에이전트에게 주문을 낸다. 6개의 에이전트들은 고객과 주문을 성사시키기 위해 서로 경쟁을 해야만 한다.

한편 에이전트가 고객 주문 PC의 생산에 필요한 부품을 공급받기 위해서는 해당 부품들을 제조하는 공급업체로부터 부품을 조달해야 한다. CPU, 마더보드, 메모리, 디스크 드라이브를 공급하는 8개의 공급업체로 구성된 공급자들로부터 에이전트들은 해당 부품을 조달하기 위해서 서로 경쟁을 해야 한다. 공급업체 및 부품은 부품번호, 기본가격, 공급자, 부품 설명의 항목으로 테이블이 구성되어 있다. 부품이 조달되면 에이전트는 부품들을 조립하여 고객이 원하는 납기일 내에 해당 제품을 고객에게 배달하여야 한다. 만약 납기일을 초과하면 가격의 일정 비율만큼 벌금이 부과되며, 납기일에서 5일을 초과하면 고객의 주문은 자동취소 된다. 이때 PC는 부품조합, 필요생산시간, 시장유형에 따라 각각 다른 BOM(Build of Material) 을 가지며 이 역시 테이블로 제공된다.

TAC-SCM에서는 15초가 하루이며 220일 동안 진행된다. 따라서 에이전트는 15초 안에 고객의 RFQ들 중에 입찰한 RFQ를 정해서 가격을 제시해야 하며, 에이전트 및 공급업체의 일일 생산능력은 한정되어 있기 때문에 다른 에이전트보다 더 많은 이윤을 남기기 위한 부품 조달계획, 일정계획 그리고 배달계획 등의 전략과 알고리즘을 거래 환경에 맞게 효과적으로 수립해야 한다. 에이전트가 판매한 금액은 TAC-SCM 서버 은행에 예금되며 부품 구매비용, 재고유지비용, 그리고 납기일을 초과한 벌금 등이 은행의 잔고에서 차감된다. 6개의 에이전트들 중 거래가 끝났을 때 은행의 잔고가 가장 많은 에이전트가 최종 승자가 된다.

3.2.3 TAC-SCM 환경에서의 SOMAS 서비스 전략

TAC-SCM환경에서는 다양한 전략을 실험해 볼 수 있다. 예를 들어 생산 방식으로는 재고기반생산방식(Make-To-Stock) 혹은 주문기반생산방식(Make-To-Order) 등을 고려할 수 있다. 생산전략을 어떻게 수립하느냐에 따라 이후의 연관된 프로세스들의 전략도 달라진다. 재고기반생산방식을 위해서는 수요예측이 기반이 되며 이에 따라 BOM에 따른 MRP(Material Requirement Planning)를 전개하고 필요한 부품을 구매하기 위한 구매전략을 수립해야 한다. 그러나 주문기반으로 할 경우 주문된 제품을 구매하기 위한 일정계획과 부품구매전략이 수립된다. 이와 같이 어떠한 전략을 수립하는가에 따라 관련된 각 프로세스 별로 전략이 구체화 되어야 한다. 예를 들어 수요예측이나 일정계획의 방법론도 다양한 방법론들이 고려될 수 있다. 본

연구에서는 시나리오상에서 PC제조를 가정하고 있고, PC제조라는 그 특성에 따라 재고기반 생산 방식으로 생산전략을 수립하였다.

(1) 수요예측 서비스

수요예측 서비스는 과거 고객의 수요 정보를 바탕으로 미래 수요를 예측하여 필요한 부품의 조달 및 생산계획을 세우기 위해 가장 중요한 정보를 제공하는 서비스이다. 고객의 수요, 즉 d 날에 발행되는 RFQ(Request For Quote)의 수는 컴퓨터 시장의 세 가지 유형별로 아래의 수식과 같이 포아송 분포에 따르며 Trend(τ)는 -0.1에서 0.01사이의 변수를 발생시켜 특정일에 발행되는 RFQ의 수를 변화시키고 있다. 게임 첫째 날의 Q_0 는 컴퓨터시장의 세 가지 유형에 따라 High Range, Middle Range, Low Range내에 주어진 범위에서 임의로 정해진다.

고객의 RFQ상의 납기일은 현재일에서 향후 3일부터 12일까지의 사이에서 임의로 정해지며 가격은 해당 PC를 조립하기 위해 필요한 부품의 기준가격(Base Price)의 합에서 75~125%의 범위에서 임의로 정해진다. 각 RFQ상에서의 주문수량은 역시 임의로 정해진다. 이와 같은 고객의 수요변화를 예측하기 위해서 수요예측은 최소자승법을 이용한 단순회귀분석을 사용하였다. 수요와 관련된 수식을 보면 계절적 요소나 변동적인 요소는 포함되지 않고 경향만을 포함하고 있기 때문에 경향을 분석하면 수요를 예측할 수 있다. 경향만을 가지는 수요예측에서는 지수평활법, 이동평균법, 회귀분석예측이 있다. 본 연구에서는 제한된 시간 내에 여러 가지 예측과 계획을 수립해야 하므로 단순회귀분석을 사용하였다. 입력데이터는 과거 10일 동안의 고객 RFQ이며 향후 12일간의 고객 수요를 예측한다.

향후 12일간의 수요를 예측하는 이유는 고객이 발행하는 RFQ상의 납기(Due date)가 향후 3일부터 12까지의 범위를 가지기 때문이다.

수요예측 서비스에서 예측된 수요정보를 바탕으로 부품조달계획을 수립한다. 예측된 수요는 전체시장의 수요이므로 전체 시장의 수요 중 자신의 에이전트가 생산하여 판매할 목표시장 및 수량을 결정해야 한다. 본 연구에서는 예측된 전체 수요를 6개 에이전트가 평등하게 기회를 가진다는 가정 하에 예측된 수요를 주어진 범위에서 나눈 값을 목표시장 및 수량 값으로 사용하였다. TAC-SCM에서는 공급자의 생산능력이 한정되어 있기 때문에 에이전트가 필요한 부품을 원하는 시기에 원하는 수량을 공급받기 위해서는 경쟁을 벌여야 한다. 따라서 경쟁의 패배로 부품을 공급받지 못하면 입찰 또는 고객의 주문을 충족시킬 수 없게 된다. 필요한 부품을 조달하기 위해 에이전트가 공급자에게 RFQ를 보내면 공급자는 세 가지의 계약조건을 보내오게 된다. 첫 번째는 에이전트가 요구한 RFQ 상의 수량과 다른, 즉 요구수량보다 낮은 수량을 제시하거나, 두 번째는 요구한 납기일보다 빠른 납기일을 제시할 수 있다. 세 번째는 고객이 요구한 수량과 납기일을 모두 준수한 계약조건이다. 세 번째의 경우는 문제 될 것이 없다. 그러나 첫 번째의 경우 요구수량보다 낮기 때문에 생산계획에 차질이 생길 수 있으며, 두 번째의 경우는 재고비용이 발생한다. 에이전트는 이러한 경우 Offer를 수락하여 주문을 할 것인지에 대한 의사결정이 필요하다.

(2) 부품조달 서비스

부품조달 서비스에서는 수요예측을 바탕으로 공급자에게 RFQ를 보내는데 현재일로부터 5일

과 6일 후에 생산할 부품들에 대해서 작성한다. 공급자로부터 부품이 입고된 다음날에 그 부품을 이용해 생산할 수 있으므로 부품구매를 위한 RFQ상의 납기일은 생산일보다 하루 이른 시간으로 정하며 가격은 기준가격 \times 1.25 의 값으로 고정하였다.

(3) 생산 서비스

생산 서비스에서는 부품조달 서비스와 같은 방식으로 정한 목표시장 및 수량을 생산하기 위한 일정계획을 수립한다. 에이전트의 일일 생산 능력은 2,000 사이클로 한정되어 있다. 본 연구에서의 일정계획방법은 배낭문제로 모델링하여 생산순서만 결정하도록 하였다. 즉 조달된 부품을 바탕으로 수요가 높은 제품, 제품 사이클이 낮은 제품, 기본가격이 높은 제품부터 배낭에 넣어서 생산하는 하도록 하였다. 생산일정계획이 수립되면 생산 서비스는 이를 바탕으로 ATP (Available-To-Promise) 정보를 생성한다. 생산 일정계획을 수립하여야 할 대상은 당일 배달되어 온 부품들이다. 하루 만에 다 생산해야 하는데 하루 즉 구매한 부품들의 조립을 위해 필요한 사이클이 2,000 사이클을 넘을 경우 다음날로 넘겨서 생산일정계획을 수립한다.

(4) 입찰 서비스

입찰 서비스는 고객의 RFQ에 대응하여 계약 조건을 제시해야 한다. 이때 고객이 제시한 납기일과 수량을 반드시 만족시켜야 한다. 따라서 입찰 서비스는 생산 서비스에서 생성된 ATP 정보와 비교하여 고객이 제시한 수량과 납기일을 지킬 수 있는 RFQ에 한해서 Offer한다. 이때 어떤 RFQ부터 ATP와 비교해야 하는 문제가 있는데 다음 수식의 값이 가장 큰 RFQ부터 ATP 정보와 비교 및 할당하도록 하였다.

- 고객이 제시한 가격 + ((납기일 - 현재일) \times ((기준가격 \times S \times 보관기일) \div 수량)),
단 S = 게임 시작시 [25%, 50%]에서 게임 서버에서 임의로 정함

입찰 서비스가 결정해야 할 또 하나 중요한 내용은 가격결정이다. 고객이 제시한 가격 이내에서 가격을 입찰해야 한다. 입찰 서비스의 가격 결정 방식은 다음과 같다.

- 입찰가격 = 부품조달 가격 \times S \times [0.6, 1.3], 단 S(재고비용) = 게임 시작시 [25%, 50%]에서 게임서버에서 임의로 정함

마진율인 [0.6, 1.3]의 범위는 수요 변화와 목적인 승률의 달성 정도에 따라 변동되며 목표 승률 즉, 오피한 RFQ 대비 주문의 비율은 80%로 주었다. ATP에 할당할 RFQ와 가격이 정해지면 해당 RFQ의 수량을 ATP에서 차감하여 ATP를 갱신한다. 더 이상 할당할 RFQ 및 ATP가 없으면 Bidding을 종료한다.

(5) 배송 서비스

배송 서비스는 생산되어 창고에 보관된 제품을 납기일 내에 고객에게 배달하기 위한 일정을 수립한다. 배달 일정계획이 수립되어 배달을 요청하면 배달은 그 다음날에 이루어지기 때문에 배달요청메시지는 납기일 하루 이전에 서버로 보내어 저장 해야 한다.

(6) 조정 서비스

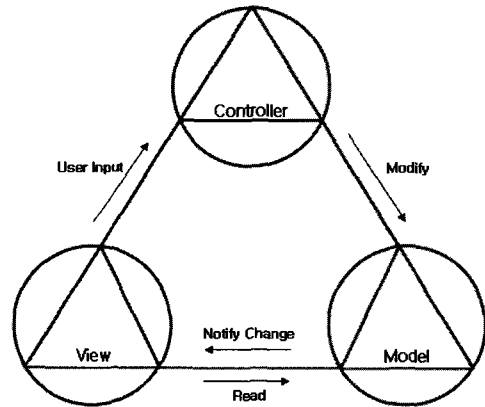
조정 서비스는 TAC-SCM 서버와의 통신과 내부 서비스간의 메시지를 관리한다. 메시지는 시기와 순서가 중요하다. 조정 서비스는 각 서비스에서 생성된 메시지들의 시기와 순서를 고려하여 외부에 전달한다. 또한 각 서비스를 호출하

여 필요한 정보와 메시지를 생성할 수 있도록 한다.

게임이 시작되면 조정 서비스는 고객의 RFQ와 주문을 접수하고 공급자에게 RFQ를 제시하고 공급자의 계약조건에 대해 주문서를 발행한다. 재고를 관리하고 공장에 생산일정계획을 보낸다. 이를 위해 조정 서비스는 수요예측 서비스를 호출하여 고객의 수요를 예측하게 한다. 조정 서비스는 재고정보와 부품구매정보 그리고 수요예측 정보를 생산 서비스에 보내고 이를 바탕으로 생산 서비스는 생산일정계획을 수립한다. 생산일정계획이 수립되면 조정 서비스가 이를 받아 서버로 전송한다. 조정 서비스는 입찰 서비스를 호출하여 고객으로 받은 RFQ 정보를 전송하고 이를 바탕으로 생성한 서비스에 보내고 넘겨받아 서버로 전송하여 입찰이 이루어지도록 한다. 부품조달 서비스를 호출하여 구매해야 할 부품에 대한 RFQ를 생성하여 서비스를 호출한다. 공급자로부터 예측된 계약조건을 받으면 해당 계약조건에 대해 모두 주문서를 발송한다. 완성된 제품들에 대해서는 납기일에 제품이 배달될 수 있도록 배송일정을 생성하여 서버에 전송한다. 게임이 다음날로 넘어 가기 전에 로그 데이터, 정보 그리고 메시지 큐에 들어 있는 데이터들을 청소한다.

3.3 BAM 대시보드 구현

SOMAS는 전술한 바와 같이 6개의 서비스들을 Java를 이용하여 구현하였다. 이러한 “TAC-SCM Server”는 개발된 알고리즘을 에이전트가 사용하도록 하여 그 운영환경으로서 제공이 되지만 실제로 각각의 에이전트들을 통제



<그림 6> MVC 아키텍처

하고 진행 업무 상황에 대한 사용자의 파악이 어려울 수 있다. 때문에 앞서 설명한 바와 같이 SOMAS를 모니터링하고 통제 할 수 있는 대시보드도 함께 개발하여 본 연구의 시스템의 인터페이스로 사용하였다. 대시보드의 개발은 플래시 애니메이션 기술을 웹 애플리케이션 서버 (WAS: Web Application Server)에 접속해 기업의 데이터베이스와 연동시킨 기술인 RIA(Rich Internet Application)를 적용하여 Flex로 구현하였다(Tapper, 2007). WAS를 통해 데이터베이스와 연동시킴으로서 여러 번의 페이지 로드 없이 하나의 창에서 한꺼번에 처리가 된다. 또한 다양한 상호작용이 가능하도록 최종 사용자가 직접 데이터를 조작할 수도 있다. 이때 MVC(Model View Controller) 모델로 구성하여 애플리케이션의 역할 세분화와 애플리케이션 아키텍처 확장이라는 두 가지 관점의 요구조건을 만족시키고자 하였다. 또한 본 연구에서는 Cairngorm 프레임워크를 이용하여 이를 가능하게 하였다. Cairngorm 프레임워크는 <그림 6>과 같이 비즈니스 로직과 화면 로직, 컨트롤 로직을 분리한 MVC(Model/View/Control) 아키텍처 패턴에 기반한 RIA 개발 프레임워크로, 화면을 표현하

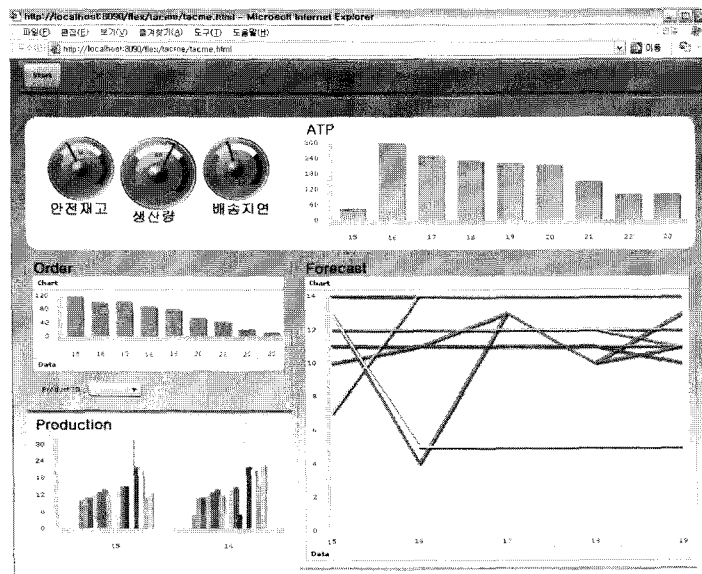
기 위한 컴포넌트로 구성된 프레임워크라 할 수 있다(Tapper, 2007). 이러한 MVC 모델에서 뷰인 사용자 인터페이스 부분을 다시 MVC 계층으로 구성 것이 Cairngorm 프레임워크이다. Cairngorm은 화면을 처리하는 과정을 view, model, value object, control, command, business 6가지로 분류하여 로직을 처리한다(이정식, 2005). 이러한 Cairngorm은 특정 애플리케이션 서비스를 제공하기 위한 라이브러리군 등을 가지지 않으며 처리 골조만을 제공하는 아키텍처 프레임워크이기 때문에 매우 간편한 프레임워크로 화면과 처리로직을 분리하고 있어, 향후 비즈니스 환경 변화에 유연하게 대처하여 쉽게 변경하거나 개선할 수 있는 장점이 있다.

Cairngorm 프레임워크는 서비스지향 방식을 이용하여 컴포넌트 간의 느슨한 결합을 유지하는 특징을 가진다(Tapper, 2007). 또한 어플리케이션을 구축할 때부터 확장성을 고려해 업무가 추가되더라도 아키텍처를 안정적으로 유지할

수 있다(이정식, 2005).

대시보드는 단일 업무 내지는 기업에 한정되어 있던 기존의 대시보드를 공급망 전반에 걸친 정보들을 활용하여 특화된 목적에 따라 사용될 수 있도록 하였다. 이종의 데이터들을 수용하고, 다시 가공하여 보여줄 수 있는 ESB(Enterprise Service Bus) 및 XML 관련 기술들을 사용하여 데이터의 통합 및 표현이 가능하다(옥상훈, 2006). 이를 통하여 다양한 이기종의 어플리케이션 인터페이스들을 공통 환경에서 통합하는 것이 가능하며, 다양한 플랫폼과 디바이스에서 개발을 할 수가 있다. 이는 곧 웹 서비스와 데이터 서비스 사용이 가능하다는 것과, 고성능의 런타임을 제공 할 수 있다는 것을 의미한다.

<그림 7>에서 상단의 게이지와 그래프는 사용자에게 문제 발생 시 경보를 주거나 주요 현황에 대한 데이터를 보여준다. 게이지의 경우 안전재고, 생산량 그리고 배송지연 등의 주요 현안에 대해 사용자가 쉽게 확인할 수 있다. 또한 그



<그림 7> SOMAS 모니터링 구현 화면

<표 2> 30회 게임 결과 평균

		SOMAS	DUMMY
Revenue		14,741,257	58,445,037
Interest		2,274,652	299,106
Costs	Material	87,798,012	48,785,320
	Storage	10,768,305	4,324,043
	Penalty	149,538	334,500
		0%	1%
Margin1		14%	10%
Margin2		16%	10%
Result(Net Profit)		18,300,055	5,300,280

* Margin1은Interest와 Penalty를 제외한 값이며 Margin2는 Interest와 Penalty를 포함한 값임

래프는 일정계획을 바탕으로 생성된 ATP 정보를 사용자에게 보여줌으로써 에이전트들의 활동을 확인 가능하게 한다. 하단의 그래프들은 실제로 주문된 주문량과 생산량에 대한 현황 및 예측 결과를 보여준다. 이러한 내용은 사용자 요구에 따라 리스트 형태 또는 그래프로 쉽게 전환되어 보여지며, 주문 현황의 필요한 내용을 쉽게 검색할 수 있도록 검색창도 함께 제공하고 있다.

IV. 알고리즘 실험

위와 같은 실험을 5개의 Dummy Agent와 함께 30회를 진행하였다. Dummy Agent는 서버측에서 만들어 놓은 것으로 기본적으로 주문제 조방식으로 생산하고 있다. 30회의 게임을 진행

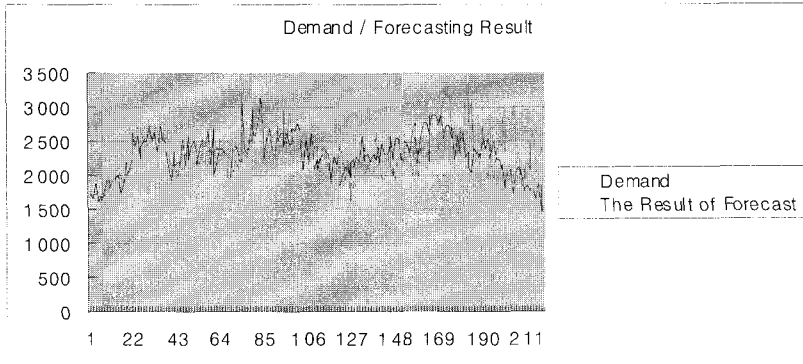
한 평균 결과를 정리하면 <표 2> 와 같다.

<표 2> 에서 보는바와 같이 30회의 게임 동안 순 수익을 보면 SOMAS은 Dummy에 비해 3배 정도의 수익을 보였다. 이는 Dummy Agent가 사용하는 간단한 MTO(Make-To-Order)에 비해서는 본 연구에서 구현한 MTS(Make-To-Stock) 전략이 우수한 것으로 보이며, 각각의 서비스도 제대로 작동한 것으로 분석된다. 다만 ATP를 기반으로 고객에게 계약조건을 제시하기 때문에 배달이 지연되는 경우는 없어야 하는데 실험해서는 지연되는 경우가 0.6% 정도 발생하였다.

그리고 30회의 게임 동안 입찰비율과 낙찰 비율의 평균값은 각각 30%, 71%로 나타났다. 2,000 Cycle/Day로 정해진 공장의 30회 게임 동안의 활용도 평균은 78% 정도를 보였다. 마지막으로 배송은 30회의 게임 중 평균 정시에 배달한 것이 99.4%,

<표 3> 30회 게임의 주요 서비스 평균값

서비스		평균
예측 / 수요		103%
낙찰 비율		71%
입찰 비율		30%
공장 활용		78%
배송	On time	99.4%
	Late	0.3%
	Missed	0.3%



<그림 8> 실수요 대비 수요예측 비율

늦은 것이 0.3%, 납기일보다 5일 이상 지연되어 주문 자체가 취소된 것이 0.3%로 나타났다. 이상을 종합하면 <표 3>과 같다.

또한 주요 서비스 중 수요예측 서비스의 경우 실수요를 제대로 예측하고 있는 것으로 분석되었다. <표 2>에서 보는 바와 같이 30게임 동안의 순수익을 보면 SOMAS은 Dummy에 비해 3배 정도의 수익을 보였다. 이는 Dummy Agent가 사용하는 간단한 MTO(Make-To-Order)에 비해서는 본 연구에서 구현한 MTS(Make-To-Stock) 전략이 우수한 것으로 보이며 각 모듈도 잘 작동한 것으로 분석된다. 다만 ATP를 기반으로 고객에게 Offer하기 때문에 배달이 지연되는 경우는 없어야 하는데 실험해서는 지연되는 경우가 0.6% 정도 발생하였다. 총 30회의 게임 동안 수요예측을 실수요량의 비율로 표시하면 <그림 8>과 같다. 30회 게임의 평균 수요 예측비 값은 약 103%였다.

VI. 결론

본 연구에서는 서비스지향 아키텍처 기반의 멀티 에이전트 시스템을 구현하여 동적 공급망관리

와 RTE를 실현하는 에이전트를 제안하였다. 이러한 멀티 에이전트 시스템은 에이전트의 주요 업무를 서비스화 함으로써 참조와 상속 관계없이 자율성을 보장하였다. 서비스 사용 방식은 결국 에이전트 시스템의 결합도를 낮춤으로서 개발 언어와 플랫폼에 대한 종속을 없애고, 이기종 플랫폼의 상호 운영이 가능하게 할 수 있다. 또한 서비스 버스를 통하여 정의되지 않은 타 개체와도 통신이 가능하게 되어 다른 개체 상호간 영향력을 행사하는 상호작용성을 높일 수 있다. 이는 곧 에이전트와 같은 지능형 기술이 시장과 고객의 요구 변화에 대해 민첩하게 반응하며, 다양하고 복잡한 문제를 신속하게 해결할 수 있다는 것이다. 이러한 에이전트를 활용한다면 개방적이고 동적인 환경에서 보다 효과적으로 수요 변화에 대응하는 공급망관리가 가능할 것으로 예상된다.

본 연구에서는 가상의 주어진 시나리오 내에서 다양한 전략을 구현하기 보다는 시나리오상에서 가장 적합한 전략만을 구현하여 적용하였기 때문에 실제 공급망 환경에서 다양한 전략을 구현하지 못한 것이 한계점이며 향후 연구 과제라 할 수 있다. 향후 연구에서는 각 서비스별로 다양한 방법론들을 시뮬레이션 하여 경험에 의한 최적의 방법론

을 적용한 알고리즘을 설계 및 구현하여 에이전트들이 사용할 수 있는 서비스로 개발할 필요가 있다.

〈참고문헌〉

육상훈, 예제로 배우는 Adobe 플렉스 에이콘, 2006
이명환, 김홍기, 김성희, 박상진, 실시간기업
(RTE), 한국기업문화연구원, 2005.

이정식, "MVC 구조 기반 웹 어플리케이션 프레임
워크 비교", 국민대학교, 2005.

신광선, "글로벌 경영환경에서의 e-SCM 전략",
서강대학교 대학원, 2002.

Caroline, Z., "Will Dashboard Computers Turn Drive
Time Into Work Time?," *HRMagazine*,
Vol.43, No.13, 1998. pp. 18~22.

Carriere, J., "Lightweight Architecture
Alternative Assessment Method (LAA
AM)" [http://blogs.msdn.com/jeromyc/
archive/2005/08/27/457081.aspx](http://blogs.msdn.com/jeromyc/archive/2005/08/27/457081.aspx).

Chen, Y.; Peng, Y.; Finin, T.; Labrou, Y.; and Cost,
S., "A Negotiation-based Multi-Agent
System for Supply Chain Management,"
*Workshop on Agent -Based Decision
Support in Managing the Internet-
Enabled Supply- Chain at Agents*,
[http://www.csee.umbc.edu/~finin
/papers/aiec99.pdf](http://www.csee.umbc.edu/~finin/papers/aiec99.pdf), 1999.

Collins J., Arunachalam R., Sade N., Eriksson J.,
Finne N., and Janson S., *The Supply
Chain Management Game for the 2007
Trading Agent Competition*, [http://
www.sics.se/tac/page.php?id=1](http://www.sics.se/tac/page.php?id=1), 2006.

Erl, T., *Service-Oriented Architecture: Concepts,
Technology, and Design*, Prentice Hall,
2006. pp. 120~122.

Gaussman, B., "Business Activity Monitoring:
Architecture," Gartner Symposium
ITEXPO, [http://findarticles.com/p/arti
cles/mi_m0EIN/is_2003_Jan_15/ai_9645
1750/pg_2/?tag=content;coll](http://findarticles.com/p/articles/mi_m0EIN/is_2003_Jan_15/ai_96451750/pg_2/?tag=content;coll), 2003.

Jennings, N.R., Roda, C., and Mamdani, E.H., "The
Impact of Heterogeneity on Cooperating
Agents," *Proc. AAAI Workshop on
Cooperation among Heterogeneous
Intelligent Systems*, Anaheim, 1991. pp.1~10

Krafzig, D., Banke, D. and Slama, D., *Enterprise
SOA*. Prentice Hall PTR. 2004.

Kumar, K., "Technology for Supporting Supply-
Chain Management," *Communications of
the ACM* Vol.44, No.6, 2001. pp. 58~61

Lambert, S. E., *Fundamental of Logistics
Management*, McGraw-Hill Internation
al Editions, 1998.

Mcgee, K., *Gartner Research Note* DF-22-2973,
2004.

Raskino, M. *Gartner Research Note* COM-21-3312,
2004.

Tapper, J., Talbot, J., *Flex 2*, Adobe Press, 2007.

최형림(Hyugrim, Choi)



서울대학교 경영학과에서
학사를 마쳤으며, KAIST대학
원에서 경영과학으로 석사 및
박사학위를 취득하였다. 현재
동아대학교 경영정보학과에 재
직 중이며, 다수의 항만물류 및
에이전트 관련 프로젝트에 참

여하였다. 주요 연구 관심분야는 에이전트시스템, 의사결정지원시스템, 스케줄링 등이며, 현재 자동화 컨테이너터미널 관련 프로젝트 및 다수의 학술연구 프로젝트를 진행 중이다.

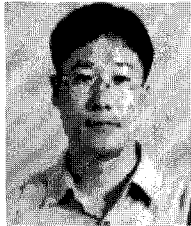
김현수(Hyunsoo, Kim)



서울대학교에서 경영학 학사, 한국과학기술원(KAIST) 경영과학과에서 경영정보학 석사와 박사를 취득하고, 현재 동아대학교 경영정보학과 교수로 재직하고 있다. 주요 관심 분야는 에이전트를 활용한 협

상 방법론의 개발과 이의 항만 및 공급사슬망 구성에의 응용, 지능형 정보시스템에 관한 이론 및 기술 개발이다.

홍순구(Soongoo, Hong)



영남대학교에서 경영학사, 미국 네브라스카 주립대학교에서 석사 및 박사 학위를 취득하였다. 현재 동아대학교 경영정보학과 부교수로 재직중이다.

연구 관심분야는 Data Warehousing, Knowledge Management, e-commerce, IS Evaluation, ERP 등이다.

박영재(Youngjae, Park)

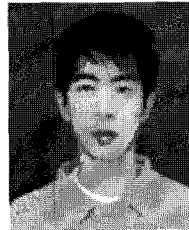


동아대학교에서 경영학사, 동 대학에서 경영정보학 석사와 박사를 취득하고, 현재 동명대학교 전임강사로 재직중이다. 주요 관심분야는 공급망관리, Agent System, 지능

형 정보시스템에 관한 이론 및 기술 개발이다.

조민제(Minje, Cho)

동아대학교 경영학부를 졸업하고, 동 대학 경영정보학 석사를 취득, 현재 동대학원 박사과정에 재학중이다. 주요 관심분야는 SOA, Agent System, 기업정보



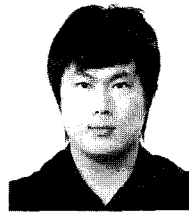
시스템에 관한 이론 및 기술 개발이다.

최성욱(Sungwook, Choi)

동아대학교 산업공학과에서 학사를 마쳤으며, 부경대학교에서 컴퓨터 공학으로 석사학위, 한국해양대학교에서 전자통신 공학 박사학위를 취득하였다. 현재 부산인적자원개발원에 재직 중이며, 다수의 정보시스템 개발 프로젝트에 참여하였다. 주요 연구 관심분야는

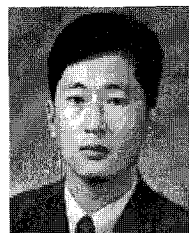
e-러닝시스템이다.

박창현(Changhyun, Park)



동아대학교 경영정보학과에서 학사를 마쳤으며, 현재동아대학교 대학원에서 경영정보학으로 석사학위 과정 중에 있다. 연구 관심 분야는 물류정보시스템 개발이다.

강시협(Sihyeob, Kang)



부산정보통신대학교에서 학사를 마쳤으며, 동아대학교에서 경영정보 석사석사를 취득, 현재 동대학원 박사과정에 재학중이다. 연구 관심 분야는 물류정보시스템 개발이다

안병선(Bungsun, An)

현재 (주)넷솔루트에 이사로 재직중이며, 다수의 정보시스템 개발 프로젝트에 참여하였다. 연구 관심 분야는 건축행정정보시스템 개발이다.

<Abstract>

Development of a SOA-based Distributed Collaborative Agent in RTE Environments

Hyung Rim Choi · Hyun Soo Kim · Soon Goo Hong · Young Jae Park · Min Je Cho · Sung Wook Choi · Chang Hyun Park · Si Hyeob Kang · Bung Sun An

The organizations in the competitive business environment are under pressure of handling internal management processes as well as external supply chain issues. In addition, the paradigm of supply chain management (SCM) has been shifted from static to dynamic problems. For the efficient SCM, organizations should be a RTE (Real Time Enterprises) that views entire supply chains as a single entity. In this paper, the agent system that makes RTE and dynamic supply chains possible is developed. Then, the developed agent system is verified in the TAC SCM(Trading Agent Competition Supply Chain Management) experimental environment and a dashboard is also developed for monitoring management activities of supply chains. The contributions of this paper are as followings. First, unlike the previous agent systems, the suggested agent system in this paper provides autonomy and scalability using SOA. Second, this suggested model for dynamic environment can be applied for the development of software supporting a SCM.

Keywords: Supply Chain Management(SCM), Real-Time Enterprise(RTE), Service Oriented Architecture(SOA), Business Activity Monitoring(BAM), Multi Agent System(MAS)

* 이 논문은 2009년 12월 30일 접수하여 1차 수정을 거쳐 2010년 3월 15일 게재 확정되었습니다.