구축 사례 기반 EAI 표준화 방안

모코코 오이식 · 정중훈
현대정보기술 임정석

1. 서 론

급변하는 기업 경영환경에서 기업의 다양한 플랫폼, 데이터베이스, 네트워크, 패키지 및 레거시 어플리케이션들을 저비용으로 단기간에 효율적으로 통합하는 것이 기업의 경쟁력 강화에 필요하기 때문에 EAI (Enterprise Application Integration)는 급변하는 e-Biz 환경에 신속하게 대응해야 되는 기업의 필수 IT 인프라로서 인식되고 있다. 예를 들어 기업에서 ERP나 CRM 등의 패키지 시스템을 도입할 경우, 각 패키지 자체의 구축 뿐만 아니라 패키지 시스템과 기존 업무 시스템 간의 원활한 인터페이스 연계가 성공적인 패키지 시스템 구축 평가의 한 요소가 되고 있다.

어플리케이션들간의 통합 영역은 한 기업내에서의 데이터 통합에서부터 기업과 기업 간의 프로세스 통합에 이르기까지 다양하다. EAI의 영역은 그림 1과 같이 간단하게 통합 범위에 따라 데이터 통합, B2B 통합, 프로세스 통합 등으로 분류될 수 있다.

EAI 시스템은 특정 업무 시스템에 국한되는 것이 아니라 회사의 모든 시스템을 대상으로 하기 때문에 각 업무 시스템 간의 인터페이스 통합이 전사적으로 일관성 있게 추진되기 위해서는 적절한 전사 EAI 표준화 방안이 필요하게 된다. EAI 표준화를 기반으로 업무 시스템 간의 인터페이스들이 업무 시스템 담당자들 간의 협의하에 정의, 구축, 운영되어야 한다. 모든 시스템의 업무 인터페이스를 일시에 통합하는 것이 어려우므로 EAI 프로젝트는 단계별로 수행되는 것이 일반적이며, 주로 시범 프로젝트 또는 프로젝트 초기 단계에 EAI 표준화 방안을 먼저 수립함으로써 향후 구축되는 EAI 시스템이 일관성 있고 단일화된 통합 시스템으로 유지되도록 하여야 한다.

다시 말하면, EAI 시스템 구축시에는 개발자와 시스템 관리자들이 EAI 표준화를 통하여 설계된 EAI 아키텍처 안에서 각자의 역할을 이해하고 수행할 수 있는 기준을 보유함으로써, 시스템 구축의 방향성을 유지할 수 있어야 한다. 또한, 프로젝트 종료 후 EAI 시스템 운영이나 확산이 필요한 경우, 기존 EAI 시스템의 아키텍처를 유지하면서 업무 인터페이스를 추가, 삭제, 변경할 수 있는 명확한 가이드라인이 제공될 수 있어야 한다.

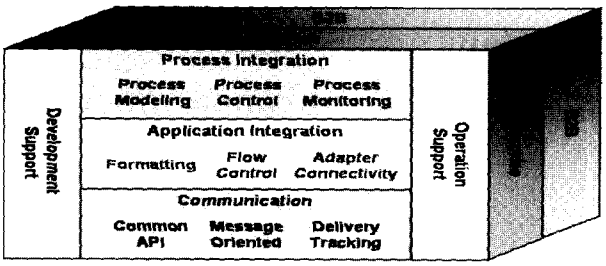


그림 1 EAI 개념도

특히 EAI 확산시, 잘 정의된 EAI 표준안을 적용한 경우, 기존 업무와 유사한 인터페이스 업무의 구축에는 초기 개발시 보다 약 3-4배의 생산성이 증대되었으며, 시스템 규모의 증가에도 기존 시스템 연계보다 적은 유지보수 인력을 운용함으로써 시스템 유지비용을 절감할 수 있었다[5].

본 고에서는 필자들이 다양한 산업 분야에서의 EAI 시스템들을 구축하면서 경험하였던 EAI 표준화의 중요성과 구축에 적용하였던 개발 프로세스 표준화, 인터페이스 표준화, 어뎁터 표준화, 전문 표준화, 자원 표준화, 보안 체계 표준화, 운영 표준화 방안을 살펴봄으로써, 향후 EAI 시스템을 도입할 기업들에게 효율적인 EAI 표준화 방안을 제시하고자 한다.

2. EAI 표준화 범위

EAI 표준화는 넓은 의미에서 EAI 개발과 운영에 관계된 모든 프로세스의 체계적인 수행 방안을 의미한다. EAI 표준화 방안을 수립하기 위해서는 회사의 업무 시스템 현황을 파악하는 것 뿐만 아니라 사용할 EAI 솔루션의 정확한 기능과 기술적 제약 사항들을 바르게 이해하는 것이 구축 목표에 적합한 EAI 표준을 지정하는데 필요하다.

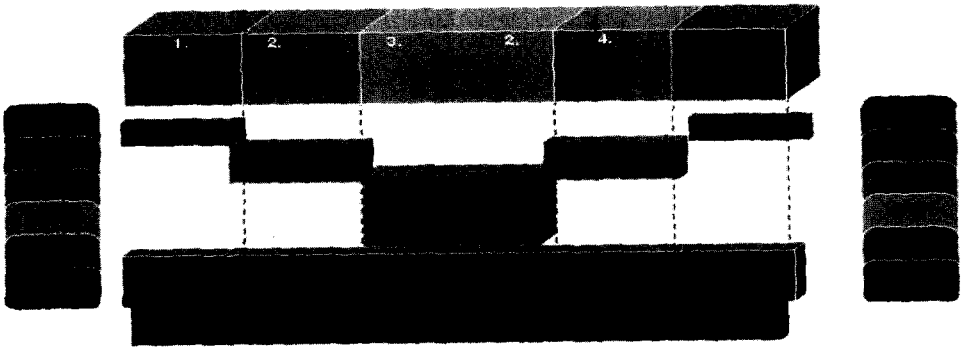


그림 2 EAI 솔루션 구성도

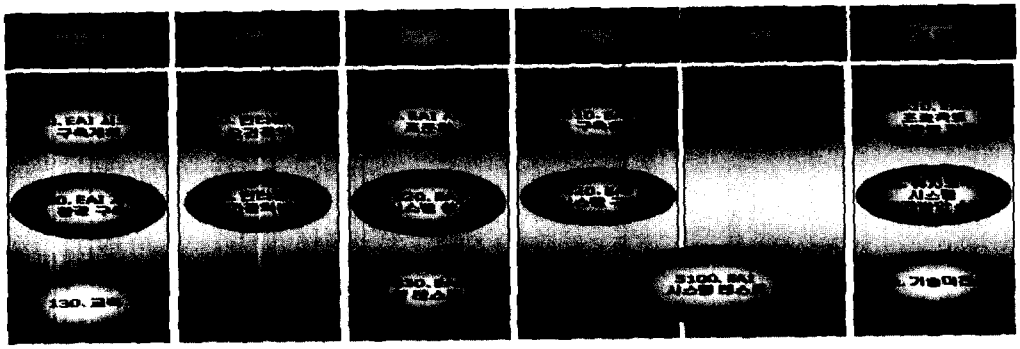


그림 3 EAI 방법론 프로세스

따라서, 우선 EAI 솔루션의 구축에 필요한 구성 요소들을 정의할 필요가 있다. EAI 솔루션의 아키텍처, 지원 어댑터 종류, 미들웨어의 특성, 모니터링 도구의 기능에 따라서 인터페이스의 구축에 관련된 표준화 방안의 세부 사항들이 결정되기 때문이다. 표 1은 일반적인 EAI 솔루션들의 구성 요소인 메시지 브로커, 미들웨어, 어댑터, 관리 및 모니터링 도구들의 기능을 간단히 정리한 것이다.

EAI 아키텍처는 Hub&Spoke, Pub/Sub, P2P 등의 구조로 구성할 수 있다. 그림 2는 HuB&Spoke 구조로 구성할 때의 어댑터, 메시지 브로커, 미들웨어, 관리도구 간의 기능관계를 보여준다.

앞에서 설명한 바와 같이 EAI 표준화는 사용하는 EAI 솔루션 및 EAI 구축 방법론과 깊은 상관관계가 있다. 본 고에서 제시하는 EAI 표준화 방안은 EAI 솔루션으로 IBM WBI[1] (WebSphere Business Integration)[1]와 MTE[2] (*MOCOCO* Tools for EAI)[2], 방법론으로는 MSDM for EAI[3](MOCOCO System Development Methodology for EAI)[4]를 적용한 사례들을 기반으로 하였음을 밝혀둔다.

EAI 시스템 구축 프로젝트와 관련된 표준화는 그림 3과 같이 EAI 방법론[4]의 각 단계에서 수행되는 프로세스들을 포함한다. 각 단계에서 수행되는 태스크은 아래와 같다.

표 1 EAI 솔루션 구성 요소

구성 요소	주요 기능	비고
메시지 브로커 (Massage Broker)	데이터 통합을 위해 전송되는 데이터의 포맷팅과 라우팅 기능 지원	대부분의 EAI 솔루션에서 제공
미들웨어 (Middleware)	시스템 간의 데이터 전송 시 데이터의 전송 보장 지원	EAI 솔루션의 특성에 따라 제공되지 않는 경우도 있음
어댑터 (Adapter)	Package 시스템이나 어플리케이션 데이터 추출 또는 적재 지원	패키지나 어플리케이션들에 따라 다양한 종류 제공
관리/모니터링 도구	EAI 솔루션의 관리 및 인터페이스 데이터의 전송 상태 모니터링 도구	시스템 & 인터페이스 모니터링 도구

- **계획 단계** : EAI 통합 일정과 범위를 고려한 프로젝트의 수행 계획을 세운다.
- **분석 단계** : 업무 담당자가 작성한 요건 정의서를 분석하여, 인터페이스 하고자 하는 대상을 도출하여 구체화한다.

1) IBM의 EAI 솔루션
2) MOCOCO의 EAI 솔루션
3) MOCOCO의 EAI 구축 방법론

- **설계 단계** : 인터페이스 분석을 기준으로 인터페이스 유형별 처리 방안을 설정하고, EAI 컴포넌트를 표준 패턴에 따라 설계한다. 이 단계에서 시범 프로젝트를 수행하여, 설계를 변경할 수도 있다.
- **구현 단계** : EAI 설계를 기반으로 EAI 구성 요소인 어댑터, 미들웨어, 메시지 브로커를 구성하고 개발한다.
- **테스트 단계** : 구축된 인터페이스를 시스템별 단위 테스트와 통합 테스트를 수행한다.
- **전개 단계** : 개발 환경에서 구성된 EAI 시스템이 운영 환경으로 이행되어 가동되면서 안정화 및 유지보수 단계로 전환된다.

표준화 요소는 방법론 각 단계에서 다양한 영역으로 정의가 가능하다. 예를 들어 솔루션의 컴포넌트 표준 설계 가이드 등이 있으나, 본고에서는 전체 EAI 구축 관련 표준화 중에서 그림 4와 같이 어댑터와 관련 있는 인터페이스 패턴 표준화를 중점적으로 고찰하였으며, 그 외에 EAI 자원 및 보안 등에 관련된 몇 가지 주요 표준화에 대해서만 살펴보았다.

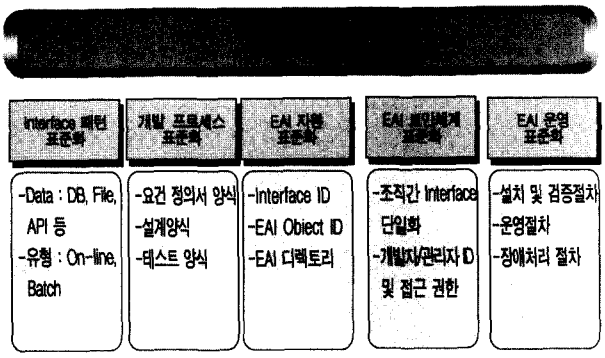


그림 4 EAI 주요 표준화 범위

3. 표준화 방안

3.1 개발 프로세스 표준화

EAI 시스템을 구축하기 위해서는 각 업무 시스템 담당자의 요청에 따라 인터페이스를 분석하여 구현하여야 한다. 전사 시스템의 인터페이스를 EAI로 통합해야 되는 시스템의 특성상 많은 업무팀과의 협의가 필요하므로, 이를 효율적이고 정확하게 진행하기 위해서 개발 프로세스를 미리 정의하여야 한다.

개발 초기의 요청 사항은 업무 시스템 담당자가 작성한 요건 정의서를 통하여 인터페이스 요건을 접수하고 이의 분석 및 설계를 거쳐 개발을 진행한다. 또한 마지막 단계로 정형화된 테스트 단계를 거쳐 개발 과정을 종료한다. 각 단계에서는 요건 정의서, 설계서, 테스트 결과서와 같은 표준화된 양식을 사용하여 개발 프로세스를 수행한다[5]. 이 개발 프로세스가 정립되면 인터페이스의 개발과 유지 보수에 대한 생산성이 개발 초기보다 약 3-4배 증가할 수 있다.

3.1.1 요건 정의서

업무 인터페이스의 특성 및 송수신 데이터 간의 관계(매핑)를 정의하는 양식이다. 전사적으로 사용되는 다양한 인터페이스 요건을 나타낼 수 있어야 하며, EAI를 잘 모르는 업무 시스템 담당자가 쉽게 이해하고 사용할 수 있는 표현을 사용해야 된다. 요건 정의서의 내용이 정확하지 않을 경우 설계부터 시스템 가동까지의 개발 전체 단계의 원만한 진행이 어렵고, 때로는 재설계 등의 추가 작업이 요구되므로 인터페이스 요건을 정확하게 충분히 기술할 수 있는 전사적인 요건 정의서의 표준화가 중요하다. 요건 정의서는 인터페이스 패턴을 파악할 수 있는 전송 유형, 전송 주기, 송수신의 데이터 발생 유형 및 송수신 데이터의 매핑서를 포함한다. 그림 5는 인터페이스 전송 유형, 데이터의 발생 유형 및 주기 등의 요건들을 정의하는 인터페이스 요건 정의서의 예를 보여준다.

NO	질 문	답 변
1	인터페이스의 전송 유형은 무엇입니까?	□ One-Way · Update □ Request-reply □ Request-Reply(Update · Confirm) □ 기타(……)
2	인터페이스 데이터의 발생 유형은?	□ Real-time □ Batch □ 기타(……)
3	인터페이스 데이터의 발생 주기는?	□ 수시 □ 정기적(일, 주 ,월) □ 기타(……)
3.1	주기당 최대 · 데이터 · 전송 회수는?	□ →…… (회)
3.2	한번에 전송하는 데이터 크기는?	□ →…… (KB)
4	송신, 데이터의 발생 유형은?	□ →파일

그림 5 인터페이스 요건 정의서의 예

3.1.2 설계서

표준화된 요건 정의서의 정확한 작성이 업무 시스템 담당자의 역할이라면, 이 요건을 정확히 분석하여 EAI 시스템에 적합하게 구현되도록 설계하는 것은 EAI 개발자의 역할이다. EAI 시스템은 여러 EAI 컴포넌트 개발자 및 전문가가 각각 맡은 부분을 설계하여 개발한 다음 통합하여 구축하므로, 이 설계서를 표준화하여 개발팀 내부의 인터페이스 구현을 일관성 있게 유지하는 것이 중요하다. 표준 설계서에는 각 인터페이스를 구현하기

위해 필요한 어댑터, 미들웨어, 메시지 브로커별 컴포넌트 설계를 포함한다. 그림 6은 메시지 브로커의 공정재고 인터페이스 관련 메시지 흐름(Message Flow)을 설계하는 인터페이스 설계서의 한 예이다.

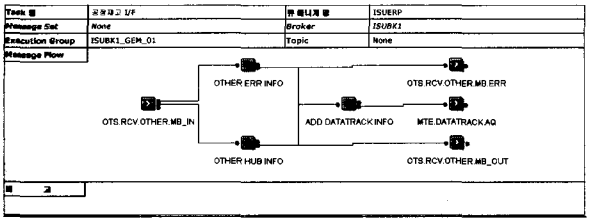
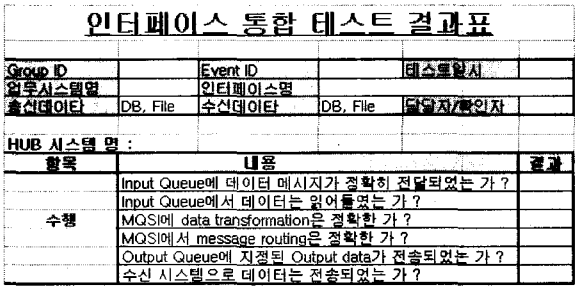


그림 6 메시지 흐름 설계서의 예

3.1.3 테스트 결과서

EAI 테스트 단계에서는 단위 테스트, 연계 테스트와 통합 테스트를 수행한다. EAI 개발자는 정형화된 각 테스트 단계에서 미리 정의된 테스트 항목을 기준으로 인터페이스가 정확히 구현 되었는가를 검증함으로써, 인터페이스 구현의 품질을 유지한다. 이 단계에서 사용되는 테스트 결과표는 인터페이스와 관계된 각 모듈들-어댑터, 미들웨어, 메시지 브로커-들의 구현 상태를 확인할 수 있도록 자세하게 작성되어야 한다. 그림 7은 각 시스템에서 데이터의 추출 및 적재 여부, 미들웨어 및 브로커의 요건 구현 여부들을 검증하기 위한 인터페이스 테스트 결과표의 예이다.



인터페이스 통합 테스트 결과표

Group ID		Event ID		테스트일시	
업무시스템명		인터페이스명			
송신데이타	DB, File	수신데이타	DB, File	담당자/확인자	

HUB 시스템 명 :

항목	내용	결과
수행	Input Queue에 데이터 메시지가 정확히 전달되었는 가 ?	
	Input Queue에서 데이터는 읽어들였는 가 ?	
	MQSI에 data transformation은 정확한 가 ?	
	MQSI에서 message routing은 정확한 가 ?	
	Output Queue에 지정된 Output data가 전송되었는 가 ?	
	수신 시스템으로 데이터는 전송되었는 가 ?	

그림 7 인터페이스 테스트 결과표의 예

표 2 인터페이스 패턴 요소

항 목	종 류
발생 유형	Online, Batch
데이터의 송수신 유형	DB, 파일, 소켓, ERP, TP 모니터 등
데이터 전송 유형	단방향 전송, 요청/응답 (Request/Reply), 병렬처리(Aggregation) 등

3.2 인터페이스 패턴 표준화

인터페이스 패턴이란 송신과 수신 시스템 간의 데이터 연계 방식을 정의하는 것으로서 인터페이스 패턴 표준화는 EAI 표준화의 핵심이라고 할 수 있다. 각 업무 시스템마다 고려하여야 할 많은 인터페이스들이 있지만, 각각의 인터페이스들의 특성을 분석해 특정 유형들의 묶음으로 정리할 수 있다. 인터페이스 패턴들을 표준화 할 때 고려하는 요소들로는 인터페이스 발생 유형, 인터페이스 데이터 송수신 유형, 인터페이스 데이터 전송 유형 등이며, 이들 요소들의 성격에 따라 인터페이스들의 유형을 표 2와 같이 분류할 수 있다.

송신/수신 유형을 포함한 다양한 기준으로 분류하여 전사적인 EAI 인터페이스 패턴들이 정의가 되면[3], 이 표준화는 EAI 프로젝트에 적용하여 검증을 하여야 한다. 검증이 완료된 후에는 EAI의 확산이나 신규 시스템의 도입시 추가 또는 변경되면서 전사 인터페이스 패턴 표준안으로 지속적으로 관리되어야 한다. 그림 8은 대고객 마케팅 데이터를 파일로 주고 받는 파일 인터페이스 송신 유형 표준화의 예를 보여준다.

인터페이스 패턴이 표준화되면 EAI 시스템에서 사용할 어댑터의 종류를 결정할 수 있다. 어댑터는 상용화된 어댑터를 사용하거나, EAI 솔루션에서 제공하는 어댑터 API를 사용하여 자체 개발을 할 수 있다.

소켓(Socket)이나 메인프레임용 어댑터들은 프로토콜이나 시스템 특성들의 이유로 자체 개발하는 경우가 많다. 인터페이스 데이터의 송신과 수신 유형에 따라서 송신과 수신 시스템에 각각 다른 어댑터가 사용될 수 있으므로, 각 인터페이스 패턴은 송신과 수신 시스템을 모두 포함하기 보다는 송신과 수신 인터페이스 패턴으로 각각 분리하여 정의하는 것이 편리하다.

일반적으로 많이 고려하는 어댑터 종류와 사용 유형들은 표 3에서 보는 바와 같다. DB 어댑터는 DB 데이터를 추출하거나 적재하는 어댑터로서, 오라클, DB2, 사이베이스 등 다양한 DBMS를 지원한다. 소켓 어댑터는 소켓 방식으로 전달되는 데이터를 EAI를 통하여 재전송하는 어댑터이다. 또한, ERP, CRM, TP 모니터는 특정 패키지를 지원하는 어댑터를 사용해야 된다. 메인프레임 어댑터는 IBM 호스트의 CICS 등을 지원하는 어댑터를 의미한다.

표 3 어댑터 종류

종 류	주요 사용 인터페이스 유형
DB 어댑터	DB 데이터의 송신/수신 인터페이스
파일 어댑터	파일 데이터의 송신/수신 인터페이스
소켓 어댑터	소켓 통신을 하는 어플리케이션의 데이터 송신/수신 인터페이스
ERP 어댑터	ERP 패키지의 데이터 송신/수신 인터페이스
CRM 어댑터	CRM 패키지의 데이터 송신/수신 인터페이스
TP 모니터 어댑터	TP 모니터와 연계되는 데이터의 송신/수신 인터페이스
메인프레임 어댑터	메인프레임 시스템의 데이터 송신/수신 인터페이스

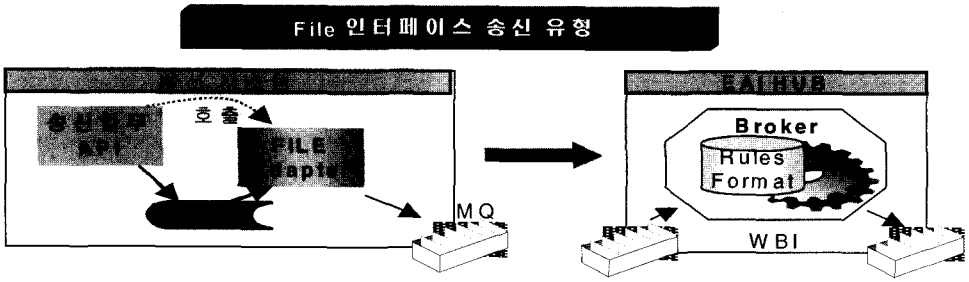


그림 8 인터페이스 패턴-파일 인터페이스 송신 유형

3.3 자원 표준화

EAI를 사용하면서 새로운 개념들을 많이 접하게 된다. 즉 메시지 큐명, 채널, 인터페이스 명, 시스템 명 등 EAI에 연관된 많은 용어들이 필요하게 된다. 메시지 큐에도 로컬 큐, 리모트 큐, 알리아스 큐, 클러스트 큐 등의 다양한 큐들이 존재하기 때문에 표준화된 자원 명의 규칙이 필요하다. 시스템 명 및 허브가 처리하는 EAI 관련 실행 그룹과 메시지 플로우 명, 그리고 인터페이스 명도 전부 표준화된 명명 규칙에 따라 생성하여야 한다. 또한, 사용하는 라이브러리나 심지어는 환경 변수들까지도 표준화를 해야 한다. 명명 규칙은 차후에 수백대의 시스템을 연계할 때에도 무리가 없도록 고안되어야 하기 때문에 표준화의 한가지로 중요성을 가지게 되었다.

3.3.1 명명 규칙 표준화

EAI 관련 항목들에 대한 명명 규칙을 표준화함으로써, EAI 개발시의 일관성을 확산 또는 운영 단계까지 유지할 수 있다. 명명 규칙은 EAI 개발팀 뿐만 아니라 업무 시스템 개발자나 관리자도 이해해야 되므로, 회사의 기존 명명 규칙에 대한 가이드라인을 참조해서 지정해야 향후 개발 및 운영에서의 EAI 적용에 필요한 시간을 절감할 수 있다. 명명 규칙 표준화에는 인터페이스 ID 및 EAI 오브젝트 이름의 표준화 등이 포함된다.

가. 인터페이스 ID : 많은 업무 인터페이스를 체계적으로 관리하기 위해 각 인터페이스마다 부여하는 이름으로, 업무 시스템 명과 어플리케이션 이름 등을 조합하여 규칙을 세운다. 단순히 일련번호를 사용하는 것은 초기 개발 단계에는 간편하여 사용하기 쉬우나, 추후에 인터페이스 종수가 늘어날 경우 일련번호와 인터페이스 간의 연관성을 찾기 어려워서 운영에 어려움을 초래할 수 있으므로 사용을 제한하는 것이 좋다. 그림 9는 명명 규칙을 적용한 한 사례를 보여준다[3].

나. EAI 오브젝트 이름 : EAI 시스템을 구축하면서 많은 EAI 오브젝트(Object)들을 생성해서 사용하고 관리해야 되므로, 각 오브젝트들을 위한 명명 규칙을 지정하여 사용하는 것이 편리하다. 이 오브젝트들은 인터페이스 ID와 관련되어 이름이 정해지는 것이 일반적인데, 해당 인터페이스들이 공통 오브젝트들을 사용할 경우가 많으므로, 이러한 특수 경우까지 고려하여 이름을 정해야 된다. 예를 들어 영업 시스템(SALES)의 배치용 큐를 SALES.BATCH.LQ로 표시할 수 있다.

3.3.2 디렉토리(Directory) 표준화

각 업무 시스템이나 EAI 전용 시스템에 EAI 모듈이 설치되거나 구성되므로 이의 관리를 위해 EAI 디렉토리 구조를 정의하여 사용하는 것이 편리하다. 설치 및 구축 시에도 디렉토리 구조에 따라 실행 파일이나 데이터를 체계적으로 분류하여 관리할 수 있으며, 시스템 이관 작업이나 백업 시에도 EAI 관련 파일 시스템의 작업을 신속하고 정확히 할 수 있는 장점이 있다. 실행 파일 디렉토리로 BIN, 라이브러리 디렉토리로 LIB, 데이터 디렉토리로 DATA 등의 이름을 사용하여 EAI 표준 디렉토리 구조를 정의하는 것이 편리하다.

3.4 보안 체계 표준화 및 운영 표준화

3.4.1 보안 체계 표준화

EAI 시스템은 각 업무 시스템의 자원을 사용함으로 이의 보안관리를 철저히 함으로써, 회사의 보안 체계를 안정적으로 유지하도록 하여야 한다. EAI 및 일반 시스템 자원의 액세스를 위한 사용자(User) ID를 관리하고 필요한 경우에 데이터 암호화를 사용하여 EAI 보안 체계를 수립하여야 한다.

가. 개발자/관리자의 사용자 ID 관리 : EAI 시스템의 개발자 및 관리자의 사용자 ID를 통합하여 관리하고, 각 업무 범위에 따라 접근 제어를 실시하는 방안이 허가되지 않은 사용자가 업무 시스템이나 EAI 시스템에서 EAI 자원을 임의로 변경, 삭제하는 위험을 미연에 방지할 수 있다.

인터페이스 명명 규칙

수신 업무(2)+수신 업무 시스템(2)+송신 업무(2)+Reply 수신 여부(1)+수신 업무 형태(1)+sequence(2)

구 분	표 현	비 고
Reply 수신 여부	R	Request에 대해 Reply가 있는 형태
	D	Request에 대해 Reply 가 없는 형태 (File/DB 인 경우 포함)
	B	수신 Bridge에서 결과값을 확인하지 않는 형태
수신 업무 형태	N	계정계 Unisys Adapter(IT 단일 시스템)
	S	Unix R/T Adapter
	T	Tmax R/T Adapter
	P	Panda R/T Adapter
	B	Real Time 이 아닌 File/DB Adapter
	M	수신 형태가 여러 가지 일 경우
	W	Java API 송신 어뎁터

그림 9 인터페이스 명명 규칙의 예

나. 인터페이스 자원 통합 제어 : EAI 개발자나 관리자가 EAI 구축을 위해 사용하는 업무 시스템의 DB, 파일이나 패키지의 자원을 안전하게 이용할 수 있도록 적절한 접근 제어를 표준화해서 실시하여야 한다.

다. 인터페이스 데이터 암호화 : 외부 시스템과의 연계를 위해 EAI 시스템을 활용시에는 필요에 따라 데이터들을 암호화하여 사용함으로써 인터페이스의 누출을 보호하는 방안을 수립할 수 있다.

3.4.2 운영 표준화

EAI 시스템의 구축이 완료된 후에는 EAI 시스템의 안정적인 운영을 위해서 EAI 시스템의 관리 절차를 표준화하여 사용하여야 한다. 이 운영 절차에는 설치, 구성, 백업, 가동, 실시간 모니터링에 대한 운영 프로세스를 포함한다.

가. EAI 솔루션의 설치 및 구성 절차 : EAI 솔루션을 설치하게 될 경우, 사전 준비 사항부터 업무 협조 사항, 설치 및 검증 절차를 수립하여 솔루션 설치 및 구성상의 안전성을 확보하여야 한다. 대부분의 시스템이 운영 시스템일 경우에는 신규 소프트웨어 설치 및 운영에 대한 검증이 필수적이며, 이에 대한 표준화는 실제 운영 관리팀에게 있어서 중요한 자료가 된다.

나. EAI 시스템의 장애 처리 절차 : EAI 시스템의 장애시 응급조치 방법 및 처리 절차를 정의하여, 시스템 관리팀에서 EAI 장애를 신속하게 판단해서 정확하게 처리할 수 있도록 해야 한다.

다. EAI 시스템의 백업 절차 : 시스템 자원의 백업시 EAI 주요 자원도 백업할 수 있도록 보존할 로그나 실행 파일들을 정의하여야 한다.

라. EAI 시스템의 가동/중지 절차 : 시스템의 가동 및 중지 시에 EAI 시스템도 정상적으로 중지 및 가동되도록 시스템의 가동/중지 프로세스에 EAI 자원도 포함을 시켜서 타 자원과 함께 운영되도록 하여야 한다.

마. 실시간 모니터링 : EAI 시스템의 규모가 커지고 업무 범위가 확대되면서 EAI 시스템 자원과 인터페이스 데이터의 전송 현황을 실시간으로 파악하여 각종 상황에 신속히 대처하는 것이 필요하다. 적은 유지보수 인력으로는 많은 EAI 자원을 수동으로 모니터링하고 관리하기가 어려우므로 EAI 전용 모니터링 도구를 사용하여, EAI 자원의 관리 및 각종 EAI 장애 처리에 이용하는 것이 좋다. 실시간 EAI 모니터링은 EAI 표준 운영 절차에서 초기 장애 탐지에 중요한 역을 수행한다[3]. 그림 10은 EAI 자원 모니터링과 실시간 처리 현황 모니터링을 보여준다. 자원 모니터링은 EAI에 통합되어 있는 시스템의 이상 유무를 판단해서 보여주며, 실시간 처리 현황 모니터링은 EAI를 거쳐가는 모든 메시지 큐에 대해서 실시간적으로 전송되는 메시지 수 및 전송 현황 등을 보여준다. 만약 메시지가 전송이 안 되고 적체가 되는 경우에는 장애가 발생한 것으로 판단됨으로 적절한 조치가 필요함을 알 수 있다.

4. 결 론

효율적인 EAI 표준화 방안은 성공적인 EAI 시스템의 구축과 안정적인 운영을 보장하며, 개발 생산성과 유지보수 비용을 줄일 수 있는 기반이 되므로, 기존 구축 사례를 충분히 검토하여 각 회사의 환경에 적합한 EAI

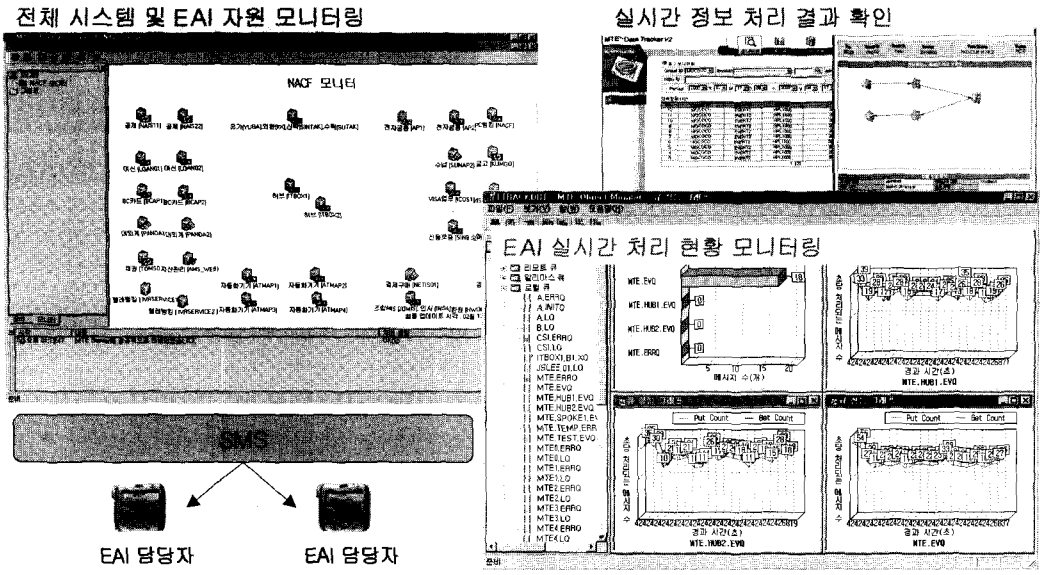


그림 10 실시간 모니터링의 예

표준화 방안을 도출하고 꾸준히 변경 관리하는 노력이 필요하다.

4.1 회사의 IT 환경에 적합한 표준 수립

EAI 표준화는 기존의 구축 사례가 많은 도움이 되지만, 회사마다 IT 환경과 업무 시스템의 구축 기준이 다르므로, 기존 표준안들을 그대로 받아들이는 것이 아니라, 회사의 IT 표준화 가이드라인에 적합하게 변경 적용되어야 한다. 일례로 명명 규칙은 EAI 기준을 참조하여 회사의 기존 명명 규칙 기준에 맞춰 사용하는 것이 업무 시스템 관리자들의 이해를 돕고, 시스템 운영에 효율성을 기할 수 있다.

4.2솔루션에 적합한 인터페이스 패턴 표준화

도입되는 EAI 솔루션의 아키텍처나 지원 사양에 따라 인터페이스 패턴의 적용 방법이 변경될 수 있다. 인터페이스 패턴은 어댑터와 밀접한 관련이 있으므로 EAI 솔루션의 어댑터 지원 여부를 확인하는 것이 필요하다. 또한 인터페이스 패턴이 모든 회사의 업무 인터페이스를 지원하는 것이 바람직하지만, 경우에 따라 발생할 수 있는 예외 사항을 고려해야 한다. EAI로 구현이 가능하더라도 EAI 솔루션으로 특정 인터페이스를 구현하는 것이 비효율적일 수 있으므로, EAI 적용 인터페이스와 비적용 인터페이스를 사례와 함께 분류해서 업무 시스템 관리자들과 공유함으로써, 초기 요건 정의 단계부터 업무 개발 영역의 범위를 정하는 것이 효율적이다.

4.3 시범 사업을 통한 표준화 검증

EAI 프로젝트에서 상세 설계를 수행하기 전에, 주요 인터페이스를 대상으로 EAI 솔루션을 적용한 시범 프로젝트를 수행하면, 솔루션을 검증할 수 있을 뿐만 아니라 EAI 표준안이 적합하게 수립되었는지도 확인할 수 있는 좋은 기회가 된다. EAI 표준안은 실제 업무 인터페이스나 시스템에 적용되어야 되므로, 사용 가능하면서 다른 표준안들과 충돌되지 않는지를 살펴야 한다.

4.4 장기적 안목 기반의 표준화 수립

EAI는 단계적 구축 전략 하에서 수행되므로, 초기에 소규모로 수행되더라도, 시간이 흐름에 따라 EAI 시스템 규모가 계속 증가하게 되는 특성이 있다. 처음 표준화 방안을 수립할 때, 현재의 IT 환경과 규모만을 고려하여 표준화 방안을 수립할 경우, 오래지 않아 표준화 방안이 수정되어야 하는 문제점이 발생할 수 있다. EAI는 회사의 기간 인프라로 자리를 잡을 것이기 때문에, 이런 표준화 방안의 잦은 변화는 EAI 시스템의 통일성과 일관성을 위협하고 많은 변경 관리로 인한 추가 비용과 재구축이 요구될 수 있으므로, EAI 구축 경험이 많은 아키텍트들이 장기적으로 사용할 수 있는 안정적인 표준화 방안을 수립하는 것이 필요하다.

참고문헌

[1] IBM WBI, http://www.ibm.com

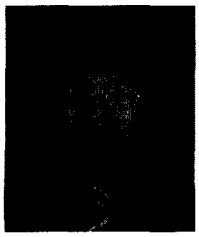
[2] MTE, http://www.mococo.co.kr

[3] 농협중앙회, "EAI 프로젝트 추진 배경 및 EAI 진행 방향", 농협 EAI 구축 완료 세미나, 2004

[4] 모코코, "MSDM for EAI (MOCOCO System Development Methodology for EAI)," 2001

[5] 유영민, "LG전자 EAI 사례 소개", IBM CIO FORUM 2003, 2003

오 이 식



1987 서울대학교 계산통계학과(학사)
1989 서울대학교 계산통계학과(석사)
1991. 9~1994. 9 한국휴렛팩커드 컴퓨터 개발부 연구원
1995. 7~현재 (주)모코코 e-Biz 사업부 이사
관심분야 : EAI, BPM, Middleware
E-mail : ysoh@mococo.com

정 중 훈

1996 동아대학교 금속공학과(학사)
2004 한양대학교 컴퓨터공학과(석사)
1995. 1~1997. 4 (주)비젼플러스 사원
1997. 4~1999. 9 (주)현대정보기술 연구소 연구원
1999. 10~2000. 12 (주)위즈네트 팀장
2001. 1~2003. 9 (주)CMT 대표이사
2003. 10~현재 (주)모코코 e-Biz 사업부 책임컨설턴트
관심분야 : EAI, BPM
E-Mail : junghun@mococo.com

임 정 석

1985 서울대학교 계산통계학과(학사)
1987 서울대학교 계산통계학과(석사)
1999 The University of Connecticut, Dept. of Computer Science and Engineering(박사)
1987. 2~1998. 7 현대전자 소프트웨어 개발부
1998. 8~현재 현대정보기술 IT Consulting 팀장
관심분야 : EAI, BPM, Web Services
E-mail : Ljs2000@hit.co.kr