

컴퓨터에서의 CTI의 구성 및 요소

김 태 식
한국 디지털 주식회사

이제 우리는 전화 없는 사회 활동을 생각할 수 없을 만큼 전화와 밀접한 관계를 맺으며 살아가고 있다. 전화는 음성을 이용한 개인간의 통신 뿐만이 아니라 FAX 및 Modem등을 이용한 정보의 교환 수단으로도 이용되고 있으며 또한 VMS (Voice Mail System) 및 ARS (Audio Response System) 등의 부가 장비들을 이용한 한 차원 높은 정보의 전달 등에도 이용되고 있다. 또한 근래 들어 각광을 받고 있는 Call Center와 같은 형태의 대량 호처리의 기능 등을 활용한 보다 고 부가적인 활동들이 모색되어 지고 있다. 각 기업은 이러한 Call Center를 활용하여 대고객 서비스의 질을 높여 매출의 증대를 기대하고 있으며 이러한 전화 및 관련 부가 장비들을 활용한 텔레마케팅 시스템을 구축하여 영업 비용의 감소와 매출의 증대라고 하는 두 마리의 토끼를 잡고자 노력하고 있다.

또한 현재의 사회는 컴퓨터 없이는 아무 일도 할 수 없다고 할 만큼 컴퓨터를 활용한 정보의 전달과 이용이 극에 달하고 있다. 대부분의 회사는 자체 업무를 처리하기 위하여 컴퓨터 시스템을 활용하고 있으며 이러한 컴퓨터 시스템은 고객 및 회사와 관련한 방대한 양의 정보를 보유하고 있다.

이에 컴퓨터와 전화 시스템을 연동하여 정보의 이용 효율을 극대화 시키고자 하는 필요성이 대두되어 이를 위한 기술적인 면에서의 접근이 이루어지고 있다. 컴퓨터-전화 시스템 연동 (CTI-Computer Telephone Integration)은 컴퓨터와 전화 시스템을 연결 시키는 모든 형태의 접속을 칭한다. 그러나 대부분의 경우 컴퓨터 전문가는 전화 시스템에 대한 이해가 부족하며 전화 시스템에 대한 전문가는 컴퓨터에 대한 이해가 부족한 실정으로써 이 두 시스템간의 연동을 통한 정보 이용의 극적인 효율화를 꾀하는데 걸림돌이 되고 있다. 이에 본 문에서는 컴퓨터 측면에서의 CTI에 대한 기술과 그 구성에 대하여 설명함으로써 사용자들이 현장에서 겪게 되는 어려움을 해소하는데 일조를 하고자 한다.

CTI란

서론에서 설명하였듯이 CTI란 컴퓨터 시스템과 전화 시스템이 연동 운영되도록 하는 모든 것을 뜻한다. 두 시스템간의 물리적 연결은 필수적이며 이에 따라 컴퓨터에서의 어플리케이션은 전화 관련 서비스를 요청할 수 있고 전화 관련 시스템을 통한 호의 추적 등이 가능하다. 이에 아래와 같은 두 가지 형태의 물리적 연결이 가능하다.

- Desktop 연결 : 한 사용자의 PC와 해당 사용자의 전화 Line간의 연결
- 서버 연결 : 서버 시스템과 전화 시스템간에 LAN을 통한 연결

각각의 구성에 대한 자세한 내용은 본문에서 알아 보기로 하고 여기서는 CTI의 기본 개념과 그 필요성에 대해서 알아보도록 한다.

컴퓨터-전화 시스템 연동을 통하여 1차적으로 얻을 수 있는 이득으로써는 안정적인 호의 운영과 컴퓨터를 이용한 정확하고 신속한 정보의 제공 및 업무 처리 효율의 극대화라고 할 수 있으며 이를 통하여 결국은 통화 시간의 단축 및 보다 향상된 대고객 서비스를 제공할 수 있게 됨으로써 비용 감소의 효과와 매출 증대의 효과를 꾀할 수 있다.

기능적인 측면에서 CTI는 어플리케이션 소프트웨어와 전화 시스템을 연결함으로써 아래와 같은 개선된 기능을 제공한다.

- 자동 전화 걸기
- 호의 처리(Call Routing)
- 호 인입시 정보 화면의 자동 Pop-up
- 호 전환시 관련 정보의 자동 전환(Call/Data Transfer)
- 인입 호의 번호 인식 기능(ANI-Automatic Number Identification)

이에 더하여 컴퓨터 기술을 도입함으로써 아래와 같은 부가적인 기능도 제공할 수 있다.

- 음성 저장 및 복원
- 화자 인식
- 음성 인식
- FAX 연동

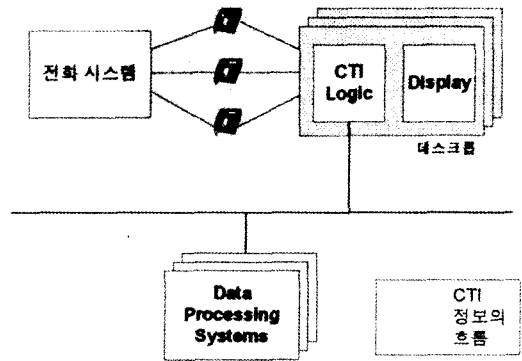
CTI의 구성 및 요소

그럼 CTI를 활용한 시스템의 구조는 어떠한지 그리고 어떠한 요소들로 구성이 되는지 살펴보자.

CTI의 구성 (CTI Link 측면)

CTI의 구조는 앞에서 언급하였듯이 데스크톱 연결 방식과 서버 연결 방식의 두가지가 있으며 그 구성 형태는 아래와 같다.

1. 데스크톱 연결 방식

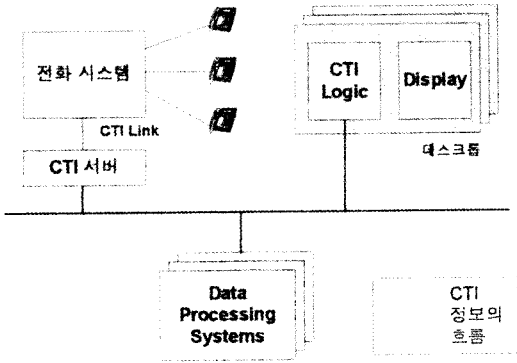


<그림 1> 데스크톱 연결 방식의 CTI

개별적인 데스크톱 연결 방식은 통상 소규모의 사용자와 단일 사용자를 지원하는 간단한 형태의 어플리케이션에 적합하며 이에 전화 Dialer같은 것을 예로 들 수 있다. 통상 10내지 15 사용자 이상의 그룹에 데스크톱 연결 방식을 채택하는 것은 비용 측면에서 불리하다. 이것은 각 데스크에 특정 컴퓨터와 전화의 물리적인 설치가 필요하기 때문이며 이에 따라 자재 및 장비 등에서의 문제 발생 소지를 보유하게 된다.

2. 서버 연결 방식

CTI 서버의 채택은 중 규모 이상 그룹의 경우 효과적이다. 또한 전체 그룹에 대한 호 추적 및 전환등의 서비스가 필요한, 보다 복잡한 형태의 어플



<그림 2> 서버 연결 방식의 CTI

리케이션에 효과적이다. 하나의 서버를 설치함으로써 네트워크에 연결된 모든 해당 어플리케이션 사용자들에게 서비스를 제공할 수 있다.

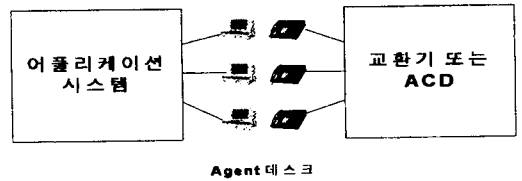
사용자의 환경이 CTI 서버를 채택하여야 하는 경우 이의 채택은 신중하게 이루어 져야 한다. 대부분의 경우에 있어 사용자는 자신이 보유하고 있는 전화 시스템에 대하여 단지 하나의 CTI 서버를 채택하기를 바라고 되고 이러한 전화 시스템은 다수의 사용자나 부서 및 업무 조직들을 지원하게 되는데 사용자가 채택하게 되는 CTI 서버가 LAN을 통한 전화 시스템과의 유일한 통로임을 고려하면, 이러한 서버는 사용자의 현재뿐만 아니라 미래의 요구 사항까지도 지원할 수 있어야 하기 때문이다.

데이터베이스나 프린트 서버와 같이, CTI 서버도 일반적인 컴퓨터 시스템에서 운용된다. 사실상 대부분의 CTI 서버 제품들은 여타 서버 기능이나 어플리케이션 등과 컴퓨터 시스템을 공유하게 된다. 따라서 시스템 구성에서의 유연성을 확보하기 위하여서는 Data Processing System이 운영되고 있는 환경과 동일한 환경에서 운영되는 CTI 서버를 채택하는 것이 일반적이다. 이렇게 함으로써 운영상 발생할 수 있는 문제들에 대한 대책수립 및 기존 운영 요원들의 재교육 문제 등과 같은 여러 문제들을 극복할 수 있다. 즉, 컴퓨터 하드웨어와 소프트웨어 및 네트워크 프로토콜 등에 대한 일관된 방향과 정책이 필요한 것이다.

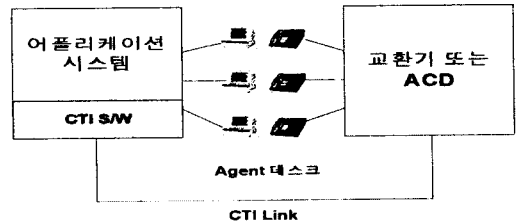
CTI의 구성 (어플리케이션 측면)

CTI가 출현하기 이전의 시스템은 전화 시스템과 컴퓨터 시스템이 각자 독자적으로 운영되는 환경이었으나 (그림 3 참조) 이후 전화 시스템과 컴퓨터 시스템의 연동의 필요성에 의하여 교환기에 CTI Link가 개발되어 어플리케이션 시스템의 CTI 소프트웨어와 연동 운영되는 형태 (그림 4 참조)를 취하게 된다. 최근에는 클라이언트/서버의 개념이 소개되면서 대부분의 CTI 구조 (Architecture)는 통상 API를 사용하는 클라이언트/서버의 형태 (그림 5 참조)로 구현되고 있다.

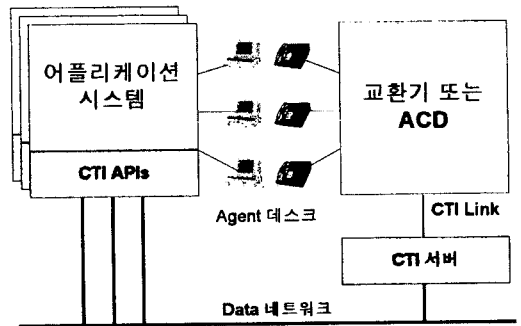
CTI 이전의 환경은 전화 시스템과 컴퓨터 시스



<그림 3> CTI의 구현이 안된 상태

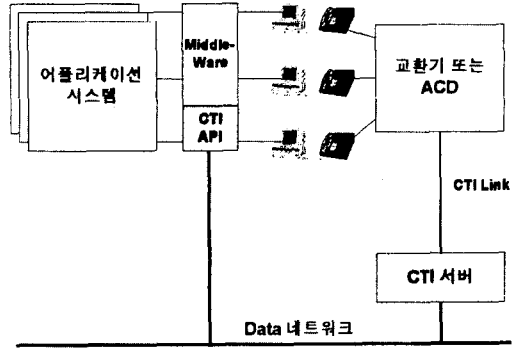


<그림 4> CTI를 구현한 상태(클라이언트/서버 형태가 아닌)

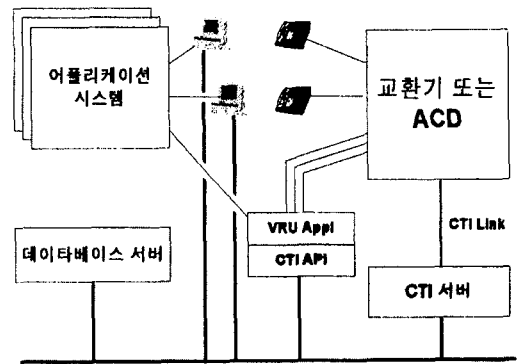


<그림 5> 클라이언트/서버 형태의 CTI 구현

템간에 어떠한 연계 체계도 구축되어 있지 않음으로써 호의 흐름과 정보의 흐름은 전혀 별개로 운영되며 따라서 호의 흐름에 따른 컴퓨터로부터의 적절한 정보 제공이 불가능할 뿐만 아니라 부가적인 서비스의 제공 또한 불가능한 실정이었다. 사실 아직 이러한 형태의 Call Center를 운영하고 있는 곳도 상당 수 존재하는 것이 현실이다. 그러나 이러한 형태의 Call Center로는 업무량의 증가나 고객의 서비스에 대한 욕구가 증가하는 데 따른 대응을 전혀 기대할 수가 없다. 반면에 교환기의 CTI Link를 이용하여 컴퓨터 시스템과 연동 운영되는 형태의 Call Center에서는 다양한 교환기의 기능들을 활용함으로써 여러 가지 부가적인 기능의 구현이 가능 하며 고객의 요구에 유연히 대응할 수 있다는 장점을 갖게 된다. 그러나 클라이언트/서버의 아키텍처가 소개되기 이전의 CTI 시스템에서는 여전히 업무량의 증가와 이에 따른 Agent의 증가에 적절히 대응할 수 없었다. CTI의 구현에 클라이언트/서버의 개념이 도입됨으로써 CTI 시스템은 Middleware를 사용하여 그 확장성 및 안정성 확보에 획기적인 전기를 맞게 된다. 클라이언트/서버 형태의 CTI 시스템을 구현하는데 있어 반드시 고려하여야 하는 것이 API (Application Programming Interface) 이다. API에 대하여는 뒤에서 자세히 알아보기로 한다. 그림 6에서는 Middleware를 이용한 보다 복잡하면서 최근에 통상적으로 많이 구축하는 형태의 CTI 구성을 보여주고 있으며 그림 7에서는 VRU를 도입한 형태의 CTI 구성을 보여 주고 있다.



〈그림 6〉 Middleware를 이용한 CTI 구성



〈그림 7〉 VRU를 도입한 일반적인 CTI 구현

Middleware를 이용한 CTI의 구성의 장점은 아래와 같다.

- Agent의 업무 처리 정보의 취득이 가능
- 어플리케이션 및 시스템의 분리 용이
- Desktop 시스템에서 구현 가능
- 현재의 어플리케이션에 대한 수정 없이 화면 Pop-up 및 호와 정보의 전달이 가능

이러한 CTI의 구성에 자동 음성 응답 장치 (VRU - Voice Response Unit) 및 FRS (FAX Response System), VMS (Voice Mail System) 등의 부가 장비를 추가하여 보다 고급의 서비스를

제공하게 된다. 물론 보다 복잡하고 다양한 서비스의 제공은 해당 처리 업무의 수준과 연계되어 결정될 사항이며 이에 따라 시스템의 구성 또한 보다 복잡하게 된다. 그러나 그 기본 구성은 교환기, CTI Link, CTI API, CTI 서버, 어플리케이션, Middleware 및 Database 서버등이 되며 이를 업무 요건에 맞게 적절히 배치 하여 구성하게 된다. 여기까지 CTI 구성에 대하여 알아보았다. 결국 CTI의 구성은 그 용도와 목적에 따라 다양하게 구축할 수 있으나 근래의 추세는 안정성, 확장성, 유연성 등을 고려하여 Middleware를 활용한 클라이언트/서버 모델의 구축이라고 할 수 있다.

CTI 서버의 선정

CTI 서버를 구성하는 요소로는 아래와 같은 것이 있다.

- CTI Link Protocol 해석기 - 교환기와 정보를 교환
- CTI 서버 Core 모듈 - 모든 클라이언트 어플리케이션으로부터의 요청을 처리하며 호 정보를 해당하는 클라이언트에 전달
- 클라이언트/서버 Transport - 서버와 클라이언트간의 정보 전달
- 클라이언트 어플리케이션 Interface - 각 어플리케이션에 맞는 형태로 정보를 포맷팅하여 이를 전달

CTI 서버를 공급하는 업체는 많이 있으나 근래에 많이 사용되는 것은 마이크로소프트사의 TAPI와 노벨사의 TSAPI 및 ECMA의 CSTA (Computer Supported Telephony Applications) 규약을 준수하는 독립된 업체로부터의 CTI 서버들이다.

아래의 표는 각 사의 CTI 서버에 대한 위의 각 요소들에 대한 비교표이다.

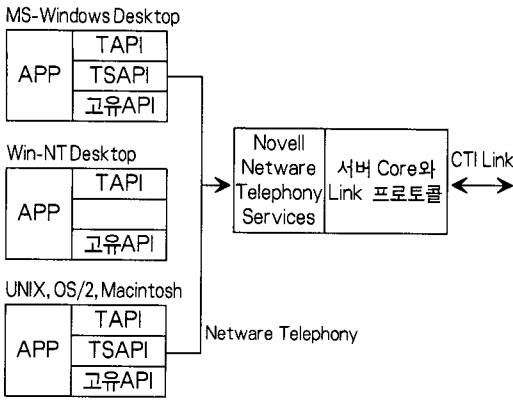
CTI API

API는 프로그램간에 의사 소통을 하기 위한 규

약과 같은 것이다. CTI에서의 API는 어플리케이션 프로그램이 전화 관련 서비스를 요청하기 위한 하나의 언어라고 생각하면 된다. 사람과 사람간의 의사 전달에서의 마찬가지로 정해진 문법에 맞게 자신이 원하는 바를 요청하여 그 결과를 받게 된다.

현재 가장 일반적으로 사용되는 API로는 Microsoft사의 TAPI (Telephony API)와 Novell사의 TSAPI (Telephony Services API)가 있으며, 또한 이러한 업체와 독립적인 업체에서 제공되는 고유 API등이 있다. 그러나 어떠한 API도 기능적으로 완벽한 것은 없으며 컴퓨터와 네트워크, 전화와 관련된 모든 영역을 지원하는 것도 없다. 따라서 어플리케이션을 개발하는 사람의 입장에서는 어느 것도 완벽한 것은 없으며 어플리케이션 개발의 목적과 용도에 맞는 최적의 API를 선택하여 사용할 수 밖에 없다. 이러한 상황은 가까운 시일 안에 바뀔 것으로 보이지는 않는다. 위의 업체에서 제공하는 API는 각각 자신의 고유 CTI 아키텍처를 가지고 있으며 따라서 한 API를 선정하게 되면 타 API와의 호환이 안되는 문제를 야기할 수 있기 때문에 자신의 업무 목표와 향후의 업무 환경 변경 가능성 등을 면밀히 검토하여 선정하여야 한다.

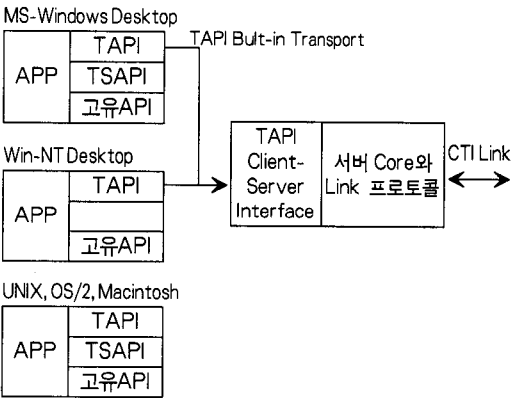
	노벨	마이크로소프트	A사	B사
Protocol Interpreter	X		CSTA, 교환기 specific	교환기 specific
Server Core 기능	X		포함	포함
Client / Server Transport	Netware 고유	Microsoft 고유	DCE	CORBA, OLE, DDE, OCX, COM
지원되는 Client O/S	Windows, Win NT, OS/2, Mac, UNIX	Windows, Windows NT	Windows, Win NT, OS/2, UNIX, VMS	Windows, Win NT, OS/2, Mac, UNIX
지원되는 API	TSAPI only	TAPI only	TAPI, TSAPI, 고유 API	TAPI 고유 API



〈그림 8〉 TSAPI Architecture

그 업체 독자의 고유 기술에 근거하고 있으며 위의 두 API를 지원하지 않는 경우가 많다. 따라서 CTI 서버를 선정할 때는 위의 두개의 API를 모두 지원하는지를 확인하는 것이 좋다.

고유 API를 제공하는 독립적인 업체는 해당 교환기 제조 업체에서 제공하는 CTI Link에 근거한 API 구현 방식을 취하고 있다. 그러나 이러한 제조 업체마다 다른 CTI Link의 문제점을 해소하고자 이에 대한 국제 표준을 만들려는 노력으로 ECMA에서 CSTA라고 하는 CTI Link에 대한 표준을 발표하여 이를 따르는 업체가 점점 증가하는 추세에 있다.



〈그림 9〉 TAPI Architecture

대부분의 교환기 제조 업체는 마이크로소프트 및 노벨 양 사의 전략을 공히 따르고 있으며 각각의 API를 지원하는 모듈을 공급하고 있으나 하나의 모듈에서 두개의 API를 공히 지원하지는 못하고 있는 실정이다. 결국 사용자는 복수 개의 API 중 하나를 선택하여야만 하는 것이다. 만약 TSAPI를 선택하였다면 TAPI에 의거한 어플리케이션은 사용할 수 없으며, TAPI를 선택하였다면 TSAPI에 의거한 어플리케이션은 사용할 수 없으며 또한 Windows가 아닌 다른 운영 환경을 가진 클라이언트는 지원이 안된다.

따라서 CTI 서버를 선정 시에는 Vendor로부터의 제품 보다는 이러한 Vendor와 무관한 독립된 업체의 제품을 선정하는 것이 바람직하다 하겠다. 그러나 많은 독립된 업체로부터의 CTI 서버 또한

CTI Link

CTI 서버는 교환기와 CTI Link를 통하여 접속된다. 이러한 CTI Link는 교환기 업체로부터 별도로 구입하여야 한다. 예를 들면 CTI Link는 각 교환기 업체로부터 CallBridge, Meridian Link, CallVisor/ASAI 등과 같은 이름으로 판매되고 있다.

또한 CTI Link 프로토콜에 대한 국제 표준으로는 ECMA(European Computer Manufacturer Association)의 CSTA(Computer Supported Telephony Applications)가 있으며 이러한 표준은 Siemens Rolm이나 Alcatel, Ericsson등을 포함한 많은 교환기 제조 업체에 의하여 받아들여지고 있다. 그러나, AT & T나 Northern Telecom등과 같은 일부 업체는 아직 자신의 고유 프로토콜을 고집하고 있는 곳들도 있다.

사실 교환기가 어떠한 프로토콜을 사용하는지는 최종 사용자의 입장에서는 중요한 것은 아니다. Link 프로토콜을 해석하여 어플리케이션 프로그램에게 API를 통하여 제공하는 것은 CTI 서버의 역할이기 때문이다. 그러나 독자의 프로토콜을 사용하는 교환기의 경우 사용자는 자신의 어플리케이션이 CTI Link와 Interface되어 작업을 할 수 있도록 해당 교환기의 Link에 맞는 Module을 개발 내지는 구매하여 사용하여야 하며 교환기의 교체 요구가 발생시 이러한 부분들을 새로 개발 내지 교체 구매를 하여야 한다. 이러한 불편한 점을 없

이기 위하여 CTI Link의 표준화가 필요한 것이다. 표준화된 CTI Link를 사용함으로써 사용자는 선택의 폭을 넓힐 수 있으며 어플리케이션을 개발하는 개발자의 입장에서는 교환기의 종류에 구애받지 않고 개발할 수 있으며 또한 각 개발자들은 이러한 표준화된 환경에서 자신의 어플리케이션의 성능을 향상시키기 위한 노력을 함으로써 전반적인 수준 향상을 기대할 수 있다. 현재의 환경에서는 일부 교환기는 CSTA Link 표준을 지원하는 반면 일부는 자신의 독자 Link를 사용하고 있기 때문에 최종 사용자의 입장에서는 현재 사용하고 있거나 조만간 구매하게 될 교환기를 지원할 수 있는 CTI 서버를 선정하는 것이 보다 중요하다. 따라서 교환기 구입시에는 교환기 제조 업체에 자신이 필요로 하는 기능들을 해당 교환기가 CTI Link를 통하여 지원하는지 확인하여야 한다. 이것은 전화 사용자에게 제공되는 기능들이 모두 CTI Link를 통하여 어플리케이션 프로그램에 제공되는 것은 아니기 때문이다.

결론

어떠한 형태의 CTI를 구축할 것인지 이에 따라 어떠한 CTI 서버가 적절한 지를 결정하는 것은 CTI 프로젝트에서의 가장 중요한 성공 요소 중 하나이다. CTI 구축 형태에 의하여 그리고 거기에 사용되는 CTI 서버에 의하여 향후의 변화에 대한 유연성과 넓은 선택의 폭을 갖을 것인가 아닌가 하는 것이 결정될 수 있기 때문이다. 모든 사용자 그룹과 어플리케이션을 지원할 수도 있으나 하나의 솔루션에 국한되거나 타 솔루션과의 연계가 안될 수도 있다.

대부분의 기술적인 결정을 함에 있어 가장 좋은 방법은 우선 사업 요구 사항들을 먼저 추출하는 것이다. 이에 따라 CTI의 수준이 단일 사용자의 형태인지, 사용자 그룹의 형태인지, 또는 부서나 회사 전체 수준의 형태인지를 결정할 수 있다. 어플리케이션의 구축은 클라이언트/서버의 어느 수

준을 유지할 것인지, Middleware를 사용할 것인지, 그리고 개방형 또는 유연성의 중요성이 어느 정도인가를 결정하여야 하며 현재 안고 있는 문제를 해결 할 수 있는 솔루션으로는 어떠한 것이 있는가도 결정하여야 한다.

CTI는 컴퓨터와 전화 시스템간의 조화다. 따라서 이 두 분야에 모두 익숙한 기술자가 있다면 가장 좋겠지만 대부분의 경우 이러한 기술자를 찾는 것은 대단히 어려운 일이다. 따라서 CTI의 구축은 컴퓨터 전문가와 전화 시스템의 전문가가 밀접한 협조 체제하에 구축하는 것이 바람직하다. 그러나 이러한 이질적인 기술 집단에 속해 있는 사람들이 함께 작업하는 것은 쉬운 일은 아니다. 동일 용어의 의미가 서로 다르며 기술 분야가 서로 다름으로 인하여 두 집단간의 의사 소통이 쉽지 않기 때문이다. 따라서 CTI 구축 프로젝트 팀의 구성은 이러한 이질 기술 집단의 인원들이 어떻게 의견을 교환하고 어떻게 문제를 해결해 가며 어떻게 업무를 구분 추진할 것인지를 세부적으로 신경써야 하는 것이다.

CTI의 구축 및 관련 제품의 선정은 단순히 기술적인 면에서의 검토만으로 이루어 지지 않는다. 업체로부터 직접 구입을 하든, SI 업체와 같이 공동으로 작업을 하든 CTI 솔루션이 업무의 전반적인 요구 사항들에 대하여 적절한지의 여부를 항상 면밀히 검토하여야 한다. 이러한 사항을 항상 염두에 두므로써 보다 명확하고 책임있는 선택을 할 수 있게 될 것이다.

저 자 소 개



金 泰 式

1958年 10月 19日生

1973年 2月 용문중학교 졸업

1977年 2月 경동고등학교 졸업

1984年 2月 고려대학교

1983年 12月~1988年 7月 두산컴퓨터

1988年 8月~1997年 6月 한국디지털

주관심 분야 : Telecom 분야 Operation Swpport Solution