

INMARSAT-C를 통한 인터넷 기반의 메시지 전송 에이전트 구현에 관한 연구

이 태 오¹⁾, 임 재 홍²⁾

A Study on a Development of Internet-based Message Transfer Agent via INMARSAT-C

Tae-Oh Lee , Jae-Hong Yim

Abstract

Sailing ship can communicate efficiently as result of the development of maritime satellite communication. The GMDSS (Global Maritime Distress and Safety System) has been developed to provide mariners with a global communications, the elements of which are capable of being operated by an individual with minimum communications knowledge and yet enable alerting and SAR(Search And Rescue) service to be reliably coordinated.

In the land, the Internet using the TCP/IP(Transmission Control Protocol /Internet Protocol) has effected extension of distributive processor, exchange and share of information via related connection of computers

But, in case of INMARSAT-C satellite communication in korea, it has been occurred a problem. Namely, the transmission message has not been connected from shipboard to land in transmitted message. Also, the data of transmission and receiving has not been managed.

Therefore, this paper describes that the development of Internet-based message transfer agent via INMARSAT-C between the land and ship. And this paper built database for the purpose of managing data of transmission and receivment.

1) 한국해양대학교 전자통신공학과 석사과정 전자·전산 전공

2) 한국해양대학교 전자통신공학과 교수

This paper is composed of 4 chapters. Chapter 1 describes the historical background and goals of this work, outline of remaining chapters. Chapter 2 describes the development of agent system and each modules of it. Chapter 3 describes the experimental result of agent system and consideration to the result. Chapter 4 summarizes all results obtained in chapters 2, 3 and includes the further research topics and the work to be supplemented.

1. 서 론

1.1 연구의 배경

최근 들어 고도로 발달한 전자·통신분야의 기술과 인공위성을 이용하여 해상에서의 발전된 통신을 행할 수 있게 되었고, 해난사고 발생 시 부근의 선박과 인명을 구조할 수 있는 수색구조(SAR : Search And Rescue)를 가능하게 하며 부수적으로 안전에 관계된 항해 및 기상 정보를 포함한 해사정보의 서비스를 제공하는 전세계 해상 조난 안전 제도(GMDSS : Global Maritime Distress and Safety System)가 1988년 국제 해사 기구(IMO : International Maritime Organization)에서 도입이 가결되어 1999년 2월 1일부터 일정한 선박에 적용하게 되었다[1, 2].

기존의 해상 조난 안전 제도에서는 선박의 총 톤 수에 따라 탑재설비를 정한데 비해 GMDSS 제도에서는 선박의 항해구역에 따라 탑재설비를 규정하고 있다. 이 중에서도 국제 해사 위성 기구(INMARSAT : International Maritime Satellite)-C는 양방향의 메시지 통신을 위하여 설계된 소형, 경량의 장비로서 무선 전화 통신은 할 수 없지만 국제 텔렉스(telex) 및 전자우편(E-mail), 컴퓨터 데이터베이스(database)를 구축할 수 있다.

INMARSAT-C 장비는 가격이 저렴하고 선박 장비의 이중화라는 GMDSS 요구사항을 만족할 수 있으므로 많은 선주들이 선호하고 있으며, 이외에도 위치 및 데이터 보고, 고기능 그룹 호출(EGC : Enhanced Group Call) 기능을 이용한 기상예보, 전자해도 교정, 데이터 수집 및 배포 등으로도 이용되고 있다[3].

이러한 INMARSAT-C 시스템을 사용하기 위해서는 이동지구국(MES : Mobile Earth Station)은 물론 육상지구국(LES : Land Earth Station) 등이 잘 운용되어야만 된다. 그러나 우리나라의 경우를 살펴보면, 육상지구국의 업무를 담당하고 있는 금산 지구국에서 이러한 기능 등을 충분히 만족시키지 못하고 있다. 즉, 선박에서 육상으로의 데이터 통신이 잘 이루어지지 못하고 있는 실정이다. 또한 육상에서는

항상 온라인(on-line) 상태가 되지 못하고, 전송한 데이터의 관리가 제대로 이루어지지 않고 있는 상황이다. 그러므로 본 연구에서는 이러한 결점을 보완하기 위하여 INMARSAT-C를 통한 인터넷 기반의 메시지 전송 에이전트(agent) 시스템을 구현하여, 보다 효율적이고 저렴한 통신 수단을 제공하고자 한다.

1.2 연구의 필요성

전자·통신분야의 기술과 인공위성 통신의 발달은 시시각각으로 변하는 정보의 신속한 전달과 공유라는 측면에서 정보시대라 불리는 현대사회의 요구에 따라 선진국에서 급진적으로 발전해 왔으며, '90년대에 들어서면서 다운사이징(down sizing)과 분산처리 경향이 뚜렷해지면서 개방형 시스템의 표준 운영 체제인 유닉스(UNIX)가 서버/클라이언트 컴퓨팅 환경의 중심 요소로 자리를 잡아 왔다[4, 5].

네트워크에서 분산처리 환경의 확산과 함께 다수 컴퓨터들 간의 상호접속을 통한 업무의 분산처리 및 종합화, 파일의 공유, 전자우편을 통한 정보교환의 필요성이 더욱 커지면서 이에 따른 컴퓨터 통신 시스템의 역할에 대한 중요성이 증대하게 된다. 특히, 이기종 통합 네트워크 구축에 필요한 다중 네트워크 어플리케이션(application)을 지원하며, 연결 지향이고 종단간에 신뢰성이 있는 통신 서비스를 제공하는 TCP/IP(Transmission Control Protocol/Internet Protocol) 프로토콜 중심의 네트워크 구성이 주를 이루고 있다.

따라서, 본 논문에서는 현재 우리나라에서 해사위성을 이용하여 통신을 하는 경우, 선박지구국과 육상간의 통신 데이터의 원활한 전송과 더불어, 육상지구국의 업무를 담당하고 있는 급산 지구국의 업무 효율을 증대하고, 전송 데이터의 효율적인 관리를 위하여 육상에서의 인터넷 기반의 네트워크를 연결한 메시지 전송을 위한 시스템을 구현하는데 있다.

2. 메시지 전송 에이전트 설계 및 구현

2.1 메시지 전송 에이전트 설계

본 절에서는 선박과 육상간에 메시지 전송을 위한 전체적인 에이전트 시스템을 설계하고, 또한, 에이전트 시스템의 각 모듈을 상세히 구현한다.

2.1.1 전체적인 에이전트 시스템 구성

에이전트 시스템을 구현하는데 있어서 운영체제는 MS Windows NT Server, 데이터 관리를 위한 데이터베이스 서버는 MS SQL Server, E-mail 서버는 MS Exchange Server, 웹(WWW : World Wide Web) 서버는 MS IIS(Internet Information Server, 개발 툴은 MS Visual Basic, VB Script, Java Script를 이용하여 구현하였다.

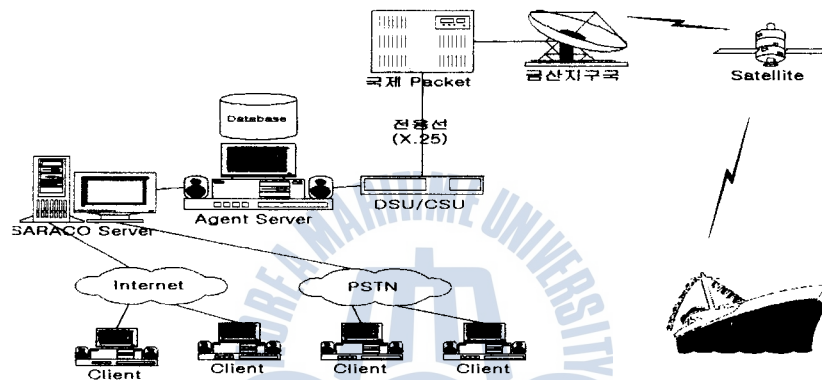


그림 1. 전체 시스템 구성

Fig 1. Overall system configuration

그림 1은 전체 시스템 구성을 나타낸다. 통신망을 보면 선박에서 금산 지구국까지는 기존의 INMARSAT-C 통신망을 이용하며, 금산 지구국에서 에이전트 서버까지는 X.25의 데이터 전용회선을 이용하고, 에이전트 서버에서 사용자까지는 두 가지의 방법을 이용하여 통신회선을 이루고 있다. 한 가지는 회사 내에 전산망이 잘 되어있어 인터넷을 이용하여 접속하는 방법과 다른 하나는 전화접속을 할 수 있는 공중전화망(PSTN : Public Switched Telephone Network)을 이용하여 접속하는 방법이다. 이러한 두 가지 방법 모두 전국 어디에서나 E-mail를 전송할 수 있다는 점이다.

2.1.2 데이터베이스 설계

메시지를 송수신 하는데 있어서 <User_ID>, <Passwd> 등 인증 절차, 관리자 연락처, 삐삐호출 등에 필요한 각종 데이터를 요구하게 된다. 이를 좀더 효율적으로 관리하기 위하여 데이터베이스를 설계한다.

표 1. 가입자 정보 테이블

Table 1. Member information table

가 입 자				
Key	ID	Data Type	Null	Format
P.K	user_ID	char(8)		
	user_name	char(20)		
	passwd	char(9)		***** *
	telephone	varchar(20)		
	fax	varchar(20)	√	
	manager	varchar(20)		
	m_telephone	varchar(20)		
	m_e-mail	varchar(40)		
	m_pager	varchar(13)		
	address	varchar(50)		-우편번호

표 2. 선박 정보 테이블

Table 2. Ship information table

선 박				
Key	ID	Data Type	Null	Format
P.K	ship_ID	char(9)		inmarsat number
	ship_name	char(20)		
F.K	user_ID	char(8)		

표 1, 2는 사용자와 선박이동국 인증을 위한 데이터가 들어가는 테이블이다. 중요부분을 설명하면 다음과 같다. <user_ID>, <passwd> 필드는 메시지 송신 시 가입자 인증을 위한 필드, <m_e-mail> 필드는 성공적인 메시지 수신에 대한 관리자에게 통보를 위한 필드, <m_pager> 필드는 수신 메시지 통보 시 관리자에게 삐삐호출을 위한 필드, <ship_ID>는 메시지 수신 시 인증을 위한 필드이다.

표 3. 송신 메시지 테이블

Table 3. Transmission message table

송신 메시지				
Key	ID	Data Type	Null	Format
P.K	send_msg_id	int		9801*****
	LES_num	int		금산,외국
	date_time	datetime		
	msg_length	smallint		최대32K
	msg_ref	int		6자리
	ship_ID	char(9)		
F.K	user_ID	char(8)		
	ocean_code	char(3)		
	send_text	text		

표 4. 수신 메시지 테이블

Table 4. Receiving message table

수신 메시지				
Key	ID	Data Type	Null	Format
P.K	get_msg_id	int		9801*****
	LES_num	int		금산,외국
	date_time	datetime		
	msg_length	smallint		최대32K
	msg_ref	int		6자리
	ship_ID	char(9)		
F.K	user_ID	char(8)		
	CC	varchar		(150)
	get_text	text		

표 3은 메시지를 송신하는 경우 데이터를 효율적으로 관리하기 위한 테이블이다. 중요부분을 설명하면 다음과 같다.

<send_msg_id> 필드는 육상 user가 메시지를 송신하는 경우 에이전트 시스템에 저장할 때의 일련 번호 필드, <LES_num> 필드는 중계 역할을 담당하는 해안지구국 번호 필드, <date_time> 필드는 송신할 때의 국제 표준시각 필드,

<msg_length> 필드는 송신되는 메시지의 길이(byte) 필드, <msg_ref> 필드는 금산 지구국으로부터 전송 받는 메시지 실행(reference) 번호 필드, <ship_ID> 필드는 수신 측의 선박 ID 필드, <corp_ID> 필드는 가입자 회사 ID 필드, <Ocean_code> 필드는 수신 측의 대양 코드 필드, <send_text> 필드는 메시지 송신 시 저장되는 실제 메시지 필드이다.

표 4는 메시지를 수신하는 경우 이를 데이터를 효율적으로 관리하기 위한 테이블이다. 중요부분을 설명하면 다음과 같다.

<get_msg_id> 필드는 선박 user가 육상으로 메시지를 송신하는 경우 에이전트 시스템에 저장할 때의 일련 번호 필드, <LES_num> 필드는 중계 역할을 담당하는 해안지구국 번호 필드, <date_time> 필드는 선박 user가 송신할 때의 국제 표준시각 필드, <msg_length> 필드는 선박에서 송신되는 메시지의 길이(byte) 필드, <msg_ref> 필드는 금산 지구국으로부터 전송 받는 메시지 실행 번호 필드, <ship_ID> 필드는 송신 측의 선박 ID 인증 필드, <corp_ID> 필드는 수신측 가입자 회사 ID 필드, <CC> 필드는 부가적인 수신측의 E-mail 저장 필드, <get_text> 필드는 육상 User가 수신하게 될 실제 메시지 저장 필드이다.

2.1.3 E-mail 메시지 포맷 설계

본 절에서는 사용자가 E-mail를 송신할 때 이를 에이전트 서버가 POP 3(Post Office Protocol)을 이용하여 메시지를 받아들일 때, 일정한 형식의 E-mail 포맷과, 필요한 부분을 본문에서 추출하는 메시지 parsing에 대해서 설명한다.

에이전트 서버에서 메시지를 parsing 하기 위해서는 E-mail나 메시지를 전송할 때는 일정한 형식에 맞추어야 parsing을 정확히 할 수가 있다.

육상에서 선박으로 메시지를 전송할 경우를 살펴보면 다음과 같다. E-mail를 보낼 때 수신처는 <inmarsat@saraco.co.kr>이 되어야만 에이전트 서버에 E-mail이 정확히 도착한다. 실제적인 <Body>부분이 매우 중요하다. 첫 번째 줄에는 <UserID>를 기입하고 구분 점인 <:>을 찍고, 옆에 실제적인 사용자 ID인 8자리를 기입하고 엔터. 두 번째 줄에는 사용자 ID와 함께 부여되는 <Passwd>를 기입하고 구분 점인 <:>을 찍고, 옆에 패스워드 9자리를 기입하고 엔터. 세 번째 줄에는 메시지를 받아들일 수 있는 수신처를 기입하는데 태평양인지 인도양인지 구분할 수 있는 <대양코드> 3자리를 기입하고 옆에 구분 점인 <,>을 찍고, 옆에 <INMARS-AT-C 단말기의 고유 ID> 9자리를 기입하고 엔터, 네 번째 줄부터는 동보 전송을 위한 것으로 세 번째 줄과 같은 형식으로 기입한다. 그 다음에 실제적인 메시지 내

그림 2. 메시지 포맷

Fig 2. Message format

용을 기입한다. 동·서 대서양의 경우는 다른 해안지구국을 이용해야만 한다.

선박에서 육상으로 메시지를 전송할 경우를 살펴보면 다음과 같다. 선박에서 메시지를 전송할 경우에는 받는 쪽의 주소를 기입한다. 즉 <대양코드> 3자리와 데이터 통신을 위한 <4500> 그 다음에 데이터통신을 위해 부여받은 <고유의 ID>를 기입한다. <Body> 부분에 들어가서 첫 번째 줄에는 <UserID>를 기입하고 구분점인 <>을 찍고, 옆에 실제적인 사용자 ID인 8자리를 기입하고 엔터. 두 번째 줄에는 <Email>을 기입하고 구분 점인 <>을 찍고, 옆에 전자우편 수신처를 기입하고, 동보 전송을 원 할 때는 구분 점인 <>을 정확히 찍는다. 그리고 다음 줄에는 실제적인 메시지 내용을 기입하는데 <MSG>을 정확히 기록하고 구분 점 <>을 찍고 엔터, 그 다음 줄부터 실제적인 내용을 기록한다.

2.1.4 에이전트 서버 설계

에이전트 서버를 구현하는데 있어서 고려되어야 할 사항이 몇 가지 있다. 먼저 사용자가 어떠한 통신망을 이용하여 E-mail를 전송하든지 이를 처리할 수 있어야 한다. 즉 모뎀을 이용하여 PSTN망에 접속하여 E-mail을 전송할 경우와 인터넷망을 이용하여 E-mail를 전송하는 경우 모두 처리할 수 있어야 한다는 것이다. 더불어서 전송 메시지의 포맷 준수, 한글코드의 지원, 메시지의 동보전송 지원, 송수신 메시지 및 결과를 데이터베이스에 저장, 메시지 전송에 관련된 기록(log)의 실시간 인쇄 등이 고려되어야 할 중요사항 이다. 에이전트 서버의 동작은 두 가지의 경우로 나누어 볼 수 있다. 즉, 육상의 사용자가 선박이동국에 메시지를 전송하는 경우, 선박이동국의 사용자가 육상의 사용자에게 메시지를 전송하는 경우로 나눌 수가

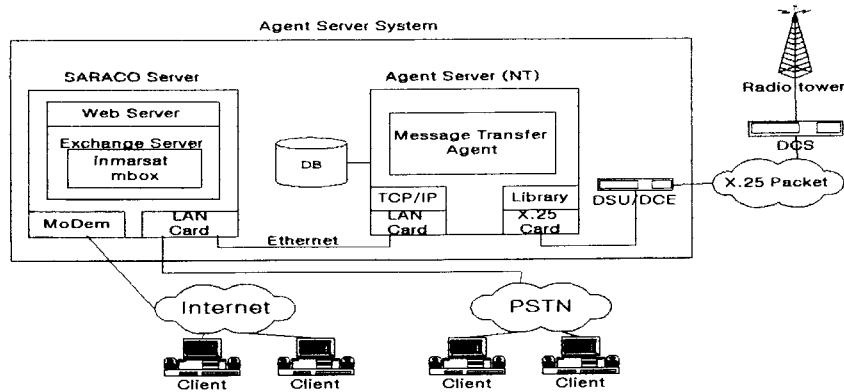


그림 3. 에이전트 서버 시스템

Fig 3. Agent server system

있다. 또한, 이러한 동작은 각각 E-mail parsing 모듈, 데이터베이스 관리 모듈, X.25 인터페이스 모듈 등의 단계를 거쳐서 이루어진다.

2.2 메시지 전송 에이전트 구현

2.2.1 데이터베이스 관리 GUI 구현

본 절에서는 3.2.2절에서 설계한 데이터베이스를 GUI(Graphical User Interface) 형태로 에이전트 관리자가 보다 편리하게 운용하기 위해서 구현하였다.

가입자를 등록할 경우 입력되는 내용을 스트링 형태로 저장을 한다. 이 경우, <회사이름>, <패스워드>, <전화번호>, <관리자 이름>, <관리자 전화번호>, <관리자 뼈번호>, <관리자 E-mail> 등은 반드시 입력되어야 할 내용들이다. 기입해야 할 칸에 내용을 기입하지 않으면 경고 메시지가 화면에 출력된다.

그림 4는 MS SQL 서버로 설계한 표 1의 테이블을 관리자가 편리하게 관리할 수 있도록 GUI 형태로 구현한 것이다. 즉 가입자 관리 GUI에서는 가입 신청을 받아 관리자가 빈칸을 작성하여 저장하면 MS SQL 서버에 저장되고, 언제나 필요한 정보를 관리할 수 있다. GUI 형태이기 때문에 등록, 수정, 삭제 등이 편리하다.

선박 관리 GUI는 표 2를 구현한 것으로서 선박 관리 GUI이다. 즉, 선박이동국의 가입 신청을 받아 관리자가 빈칸을 작성하여 저장하면 MS SQL 서버에 저장되고, 언제나 필요한 정보를 관리할 수 있다. GUI 형태이기 때문에 등록, 수정, 삭제 등이 편리하다.

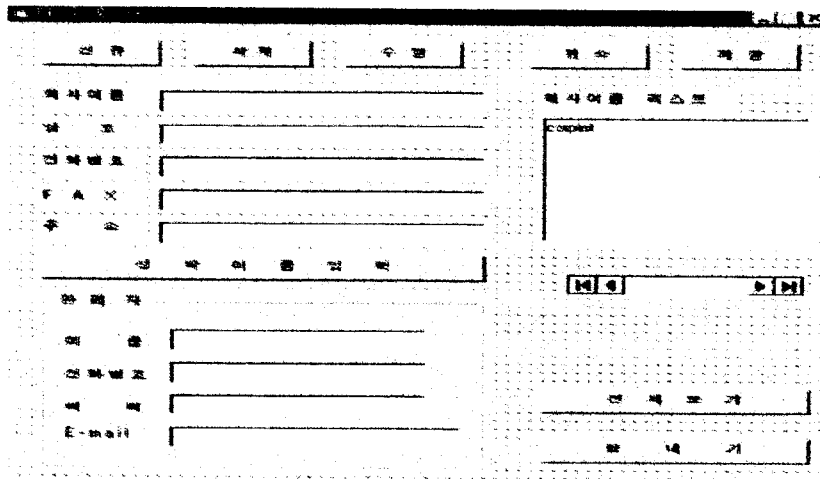


그림 4. 가입자 관리 GUI
Fig 4. Member management GUI

표 3과 표 4는 테이블을 관리할 GUI 형태의 관리의 필요가 없다. 단지 메시지를 송수신 하는데 있어서 데이터를 액세스(access)하는데 그 목적이 있고, MS SQL 서버 내부에서 자동적으로 실행되기 때문이다.

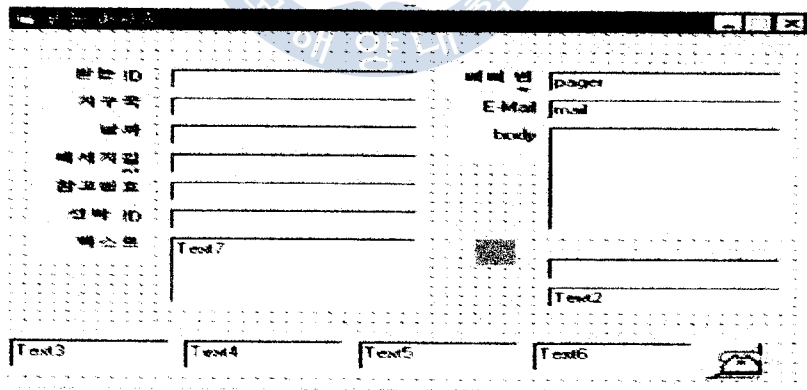


그림 5. 데이터 저장 및 삐삐호출 GUI
Fig 5. Data store and pager call GUI

그림 5는 선박이동국에서 육상 이용자에게 메시지를 전송할 때의 인터페이스를 나타내고 있다. 즉, 급산 지구국으로부터 전송되어온 메시지를 분석하여, <User_ID>, <Ship_ID>, <중계 지구국>, <수신측 E-mail 주소>, <날짜>, <삐삐호출> 기능 등을 처리하는 수신용 GUI 모듈을 나타내고 있다.

2.2.2 E-mail Parsing GUI 구현

그림 2와 같은 포맷으로 사용자가 메시지를 전송하였을 경우, 수신된 메시지를 Parsing하게 되는데, 먼저 <User_ID>, <Passwd>를 확인하고, <대양코드>, <선박이동국 번호>, <텍스트>등으로 Parsing하여 데이터베이스에 저장을 한다.

그림 6은 E-mail parsing GUI를 보여주고 있다. 에이전트를 구동하면 대기 상태에서 POP3 프로토콜을 이용하여 E-mail을 수신하기 위해서 일정 시간을 주기로 계속적으로 E-mail 수신 여부를 체크한다. E-mail을 수신하면 먼저 <사용자 ID>를 체크하여 가입 여부를 확인한다. 가입하지 않은 사용자인 경우에는 메시지를 처리하지 않고, E-mail을 송신한 사용자에게 미리 준비된 E-mail를 전송하여 가입된 사용자가 아님을 알린다. <사용자 ID>와 <Passwd>을 확인하여 인증을 통과할 경우에는 본문에서 수신처의 <대양코드>, <단말기 ID> 등을 추출하여 데이터베이스에 저장하고, X.25 인터페이스 모듈로 parsing한 부분을 넘겨준다.

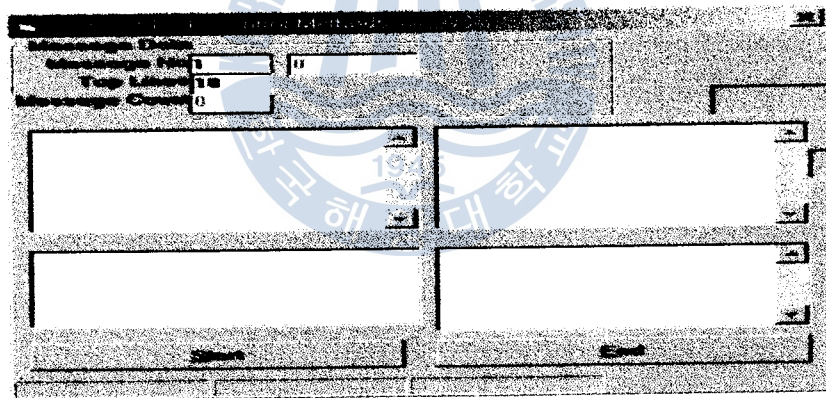


그림 6. 메시지 파싱 GUI

Fig 6. Message parsing GUI

2.2.3 X.25 인터페이스 구현

육상에서 선박지구국으로 메시지를 전송할 경우의 X.25 인터페이스 모듈은 E-mail Parsing 모듈에 포함되어 있다. 그림 22는 선박지구국에서 육상으로 메시지를 전송할 때, 금산 지구국으로부터 메시지를 받는 GUI 모듈이다. 즉, 에이전트가 구동하면 금산 지구국으로부터 메시지가 전송되어 오는지를 계속적으로 체크하고, 이벤트(event)에 의해 이를 처리하고, 이 다음에 그림 5에서 처리된다.

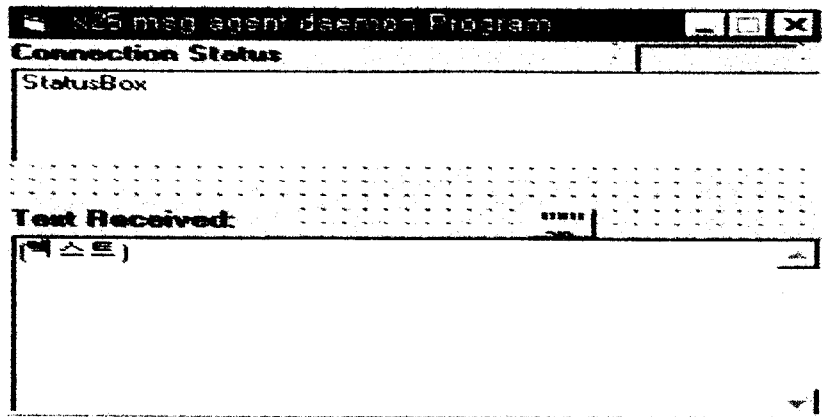


그림 7. X.25 인터페이스 GUI
Fig 7. X.25 interface GUI

2.2.4 웹 기반의 메시지 전송

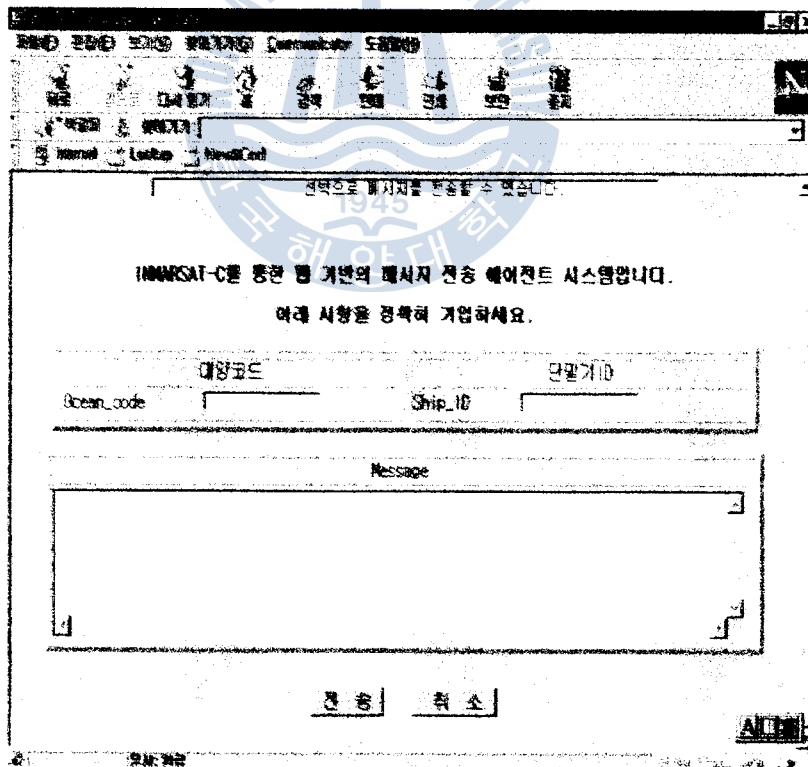


그림 8. 웹 기반의 메시지 전송
Fig 8. Web-based message transfer

그림 8은 E-mail이 아닌 웹 문서 상에서 처리를 한다. <User_ID>와 <Password> 인증 절차를 통과한 후에 그림 8의 빈 칸을 기록한 후, 전송 버튼을 누르면 메시지가 전송된다.

3. 실험 및 고찰

3.1 실험결과

X.25 인터페이스 모듈은 수신된 메시지를 분석하여, 데이터베이스에 저장하고, E-mail parsing 모듈에서 수신자에게 E-mail을 사용하여 메시지를 전송하게 된다.

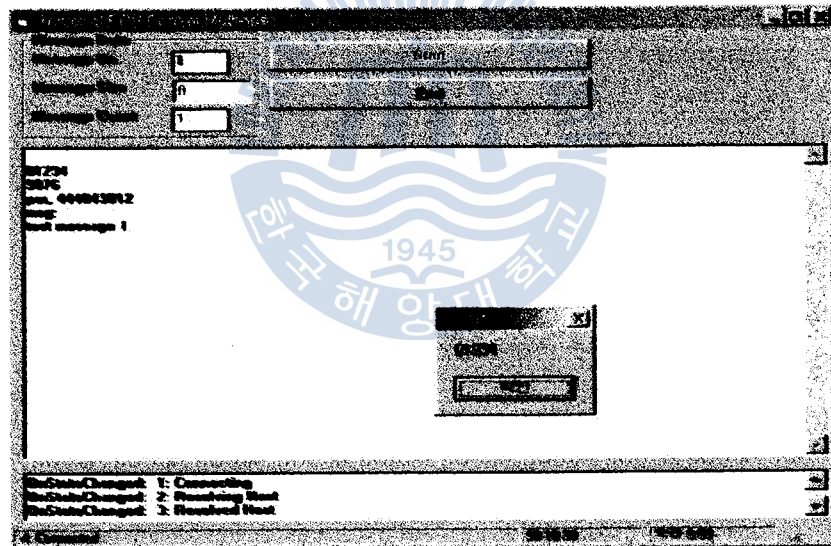


그림 9. 파싱 결과

Fig 9. Parsing result

그림 25는 육상에서 선박으로 메시지를 전송할 경우 에이전트가 E-mail을 수신한 다음에 <User_ID>, <Passwd>, <대양코드>, <메시지 내용> 등을 parsing한 결과를 나타내고 있다.

3.2 결과고찰

본 논문에서는 동작 시험을 위하여 육상 사용자 측면에서는 인터넷망과 PSTN

망을 지원하는 통신망을 이용하고, 메시지 전송과 데이터베이스 관리를 위한 에이전트 서버와 MS SQL 서버를 탑재한 데이터베이스 서버를 구성하였다. Windows 95 운영체제를 사용하는 클라이언트에서는 인터넷과 전화 접속 네트워크를 이용하여 에이전트 서버에 E-mail을 전송함으로써 육상과 선박간에 메시지 전송을 하였다.

외국의 경우는 국제 해사 위성 기구에서 INMARSAT-C를 통해 E-mail 서비스를 제공하고 있다. 이 경우는 <Thrane & Thrane Capst>, <Trimble Galaxy>, <JRC> 장비를 지원하고 있다[16]. 이 경우에도 육상과 선박지구국과 인터넷 E-mail를 전송하기 위해서 데이터 통신에는 X.25를 이용하고, 일정한 E-mail 주소 포맷을 가지고 있다는 점에서 공통점이 있다. 그러나 이 시스템은 7bit를 제공하기 때문에 영문자만을 지원하고, 한글은 지원되지 않는다.

본 논문의 결과, 선박내 자동화 시스템을 웹에서 응용 프로그램을 구동시킬 수 있는 자바(java) 언어로 구현한다면 웹 브라우저를 이용한 선박 자동화 시스템의 제어가 가능할 것이다. 더불어서 선박과 육상지구국간의 E-mail 서비스를 위한 국제적인 일정한 표준이 마련되어야 할 것이다.

4. 결 론

산업이 발달하면서 해상 물동량이 증가하므로 해상에서의 선박 사고도 늘어나게 되었다. 국제 해사 기구에서는 선박 사고에 대비한 더욱 효율적인 조난 안전 체도를 갖추고자 오랫동안 검토하여 전세계 해상 조난 안전 체도라는 새로운 해상 이동 통신 방식을 도입하게 되었다. 또한 육상에서는 정보통신의 발달로 TCP/IP 프로토콜을 기반으로 한 인터넷망의 기하급수적인 증가와 더불어 분산 컴퓨팅과 데이터의 공유 및 효율적인 관리가 대두되고 있다.

우리나라는 세계 조선 수주량 1, 2위의 위치를 점유하고 있으면서도, 해양 장비의 개발과 효율적인 사용이 이루어지지 못하고 있는 실정이다.

본 논문에서는 육상과 선박간 INMARSAT-C를 통한 인터넷 기반의 메시지 전송 에이전트를 구현하는데 있다. 이를 위하여 선박과 육상간 통신을 위한 전체적인 시스템 구성 및 데이터베이스 연동과 X.25 데이터 통신을 위한 인터페이스 모델을 제시하였다. 아울러 본 논문의 타당성 검토를 위하여 마이크로소프트 Windows NT 서버와 Windows 95 클라이언트의 연결을 이용자에게 보다 편리하고 저렴한 사용환경을 제공하기 위해서 인터넷망과 PSTN망을 이용할 수 있는 모델을 제시하고, 데이터 관리를 위한 데이터베이스 설계 및 구축, 인터넷

E-mail 기반의 메시지 전송 에이전트 개발을 통하여 송수신 메시지를 관리자에게 효율적으로 통보할 수 있다. 또한, 국내 INMARSAT-C 단말기 개발, 생산, 판매 촉진과 더불어 국내 금산 지구국의 사용 증대를 실현할 수 있게 하였다. 이 결과를 토대로 선박 내 데이터베이스 접근과 더불어 웹 브라우저를 이용한 선박 자동화 시스템의 구현 가능성을 논하게 되었다.

추후 연구로는 선박과 육상간 통신 기능과 더불어 선박 자동화 기능을 제공할 수 있는 플랫폼의 개발과 더불어 웹 브라우저를 이용하여 원격으로 선박 자동화 시스템을 제어할 수 있는 방법이 추진되어야 할 것이다.

참 고 문 헌

- [1]. 배정철, 선박용 구난 통신 시스템 개발에 관한 연구, 박사학위논문, 해양대, 1996.
- [2]. Graham D. Lees and William G. Williamson, Handbook for Marine Radio Communication, pp 37 - 41, 1996.
- [3]. Laurie Telteley & David Calcutt, Understanding GMDSS, pp 178-189, 1994.
- [4]. Stephen S. Wilson., TCP/IP NFS Internetworking in a UNIX Environment, Michael Santifaller, pp 20 - 46, 1991. 4.
- [5]. Douglas E. Comer., Internetworking with TCP/IP Principles, Protocols and Architecture, Prentice-Hall International, Inc., pp49 - 150, 1988.
- [6]. 한국통신 마케팅본부, 인말세트 통신 서비스 업무처리 지침, 1997. 10.
- [7]. 한국통신 금산위성지구국, 해사통신안내, 1996. 6.
- [8]. Werner Feibel, 첨단편집부 옮김, 네트워크 백과사전, 성안당, 1997. 9
- [9]. 최창선, 정보통신과 네트워크, 한국생산성 본부, 1993.
- [10]. 한국생산성 본부, 정보통신전문가(II), 1993.
- [11]. "컴퓨터통신". 한국통신학회지. 1987. 6.
- [12]. 통상산업부, 종합항법장치 기술에 관한 연구, 1996. 6
- [13] Microsoft, Windows NT Technical Support, Microsoft Press, 1997.
- [14] 박석(편역), 데이터베이스 시스템, 홍릉과학출판사, 1994.
- [15] 김범수, "SQL은 정보시스템과 DB의 핵", 마이크로소프트웨어, 2월호, 1997.
- [16] <http://www.inmarsat.org/c-guide>, INMARSAT-C를 통한 E-mail 서비스.