

선박에서 데이터의 통합 관리를 위한 데이터베이스 설계 및 구현

서정민¹ · 황훈규¹ · 윤진식¹ · 이성대² · 박휴찬[†] · 이장세³ · 장길웅⁴

(원고접수일 : 2010년 10월 11일, 원고수정일 : 2010년 11월 5일, 심사완료일 : 2010년 11월 12일)

A Design and Implementation of Database for Integrated Data Management on Shipboard

Jeong-Min Seo¹ · Hun-Gyu Hwang¹ · Jin-Sik Yoon¹ · Seong-Dae Lee² · Hyu-Chan Park[†] · Jang-Se Lee³ · Kil-Woong Jang⁴

요약 : 선박에서는 수많은 장비로부터 다양한 형태의 데이터가 발생되고 있다. 이러한 데이터는 주로 선박의 안전한 운항을 위하여 실시간으로 이용되고 있지만, 이를 데이터베이스화 한다면 선박의 각종 이력 정보를 검색할 수 있는 등 그 효용성이 증대될 것이다. 하지만, 이러한 데이터는 통일된 포맷이 아니라 장비의 종류에 따라 상이한 포맷을 가지기 때문에 통합된 데이터베이스의 개발이 요구되고 있다. 본 논문에서는 이러한 데이터베이스의 설계 및 구현 결과를 제시한다.

주제어 : 선박, 데이터베이스, MiTS, IEC 61162-4, NMEA 0183

Abstract: On shipboard, several kinds of data are generated from many equipments. Although these data are mainly used for the safe navigation of ship, their usability may be enhanced if they are managed on a database. However, because these data have different formats according to the kinds of equipments, an integrated database is required. This paper presents a design and implementation of such database.

Key words: Ship, Database, MiTS, IEC61162-4, NMEA 0183

1. 서 론

해상에서는 선박이 안전한 운항을 하기 위해서 수심, 수온, 풍향, 풍속, 위경도 등의 데이터를 필요로 한다. 운항을 위한 데이터는 선박의 여러 장비를 통하여 실시간으로 얻는데, 과거에는 이러한 데이터를 관리하기 위하여 직접 일지로 작성하는 방법을 사용해 왔다. 그러나 최근에는 선박 내에 발생하는 전자 데이터를 통해 선박 제어, 자율 운항, 상황 발생 시 정보 제공 등의 기능을 할 수 있는 디지털 선박이 등장하였다[1]. 또한 선박에 장

착되는 장비의 수가 증가함에 따라 발생하는 데이터의 양도 함께 증가하였다. 따라서 직접 기록하는 방법으로는 많은 양의 전자 데이터를 관리하는 데 한계가 있다.

이러한 대량의 전자 데이터를 관리하기 위하여 최근 여러 방법이 연구되고 있으며, 특히 이 중에서 데이터베이스를 활용하는 방법이 대표적이다 [2,3]. 또한 2006년 5월 국제해사기구(IMO) 해사안전위원회(MSC)의 81차 회의에서 제안된 e-Navigation에서도 이와 관련된 연구가 진행되

[†] 교신저자(한국해양대학교 IT공학부, E-mail : hcpark@hhu.ac.kr, Tel : (051)410-4573)

1 한국해양대학교 대학원 컴퓨터공학과

2 한국해양대학교 진과공학과

3 한국해양대학교 IT공학부

4 한국해양대학교 데이터정보학과

고 있다[4].

e-Navigation에서는 사용자의 의사 결정 및 작업 부하 관리를 쉽게 하고 안전 및 보안이 확보된 운항을 하기 위하여 전자 데이터의 통합 관리가 필요하다고 결정하였다[5]. 이를 위해 MiTS (Maritime information Technology Standard)에 기반하여 선박에서 발생하는 다양한 형태의 데이터를 통합 관리하고 상호 교환할 수 있는 방법이 연구되어 왔다[6]. 본 논문에서는 이러한 MiTS 시스템에서 IEC 61162-4 Series 및 NMEA 0183[7]의 Sentence 형식으로 변환된 데이터를 통합 관리하는 데이터베이스를 설계하고, 이를 바탕으로 구현 및 시험한다.

본 논문의 구성은 다음과 같다. 2장에서 MiTS와 NMEA 0183 Sentence에 대한 관련 연구를 다루고, 3장에서는 이를 기반으로 데이터베이스 및 데이터 처리 모듈을 설계한다. 또한 4장에서 구현과 시험 과정에 관하여 다룬 후, 5장의 결론 및 향후 과제로 글을 맺는다.

2. 관련 연구

선박의 수많은 장비로부터 발생하는 다양한 형태의 데이터를 통합 저장하고, 저장된 데이터로부터 체계적으로 조회하기 위해서는 데이터베이스를 활용하는 것이 효율적이다. 본 장에서는 이러한 데이터베이스의 설계와 관련된 연구로서 MiTS와 NMEA 0183 Sentence에 대하여 살펴본다.

2.1 MiTS 시스템

MiTS 시스템은 선박에서 발생하는 다양한 정보를 통합하여 관리하고 상호 교환할 수 있도록 한 것으로서, IEC 61162-4 Series와 NMEA 0183 Version 4.0를 기반으로 하고 있다. **그림 1**은 이러한 MiTS 시스템의 구성을 보여주고 있다.

MiTS 시스템은 크게 게이트웨이(G/W), 응용시스템(App), 데이터베이스 서버(DB Server)의 세 부분으로 나누어진다. 게이트웨이가 각 장비에서 발생하는 데이터를 LAN을 통하여 실시간으로 전송하면, 응용시스템은 이 데이터를 받아서 처리하고, 데이터베이스 서버는 이를 데이터베이스에

저장한다. 이 중에서 게이트웨이와 응용시스템은 여러 개가 사용될 수도 있다.

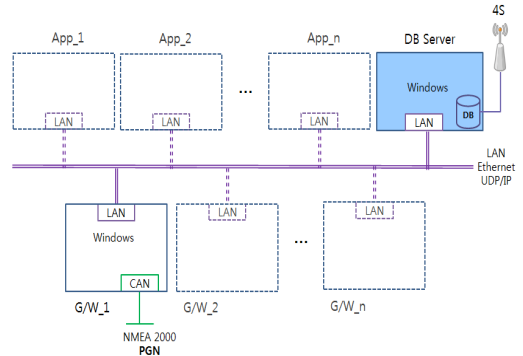


그림 1: MiTS 시스템 구성

이렇게 저장된 데이터베이스는 타 시스템에서 다양한 용도로 활용될 수 있다. 예를 들면, 선박-선박 또는 선박-육상 간의 통신을 담당하는 4S (Ship to Ship, Ship to Shore) 시스템에서 선박의 주요 정보를 보고하는 Noon Report 작성시 활용된다[8].

본 논문에서는 이러한 MiTS 시스템의 구성요소 중에서 데이터의 통합 관리를 담당하는 데이터베이스 서버에 대하여 논하고자 한다.

2.2 NMEA 0183 Sentence

NMEA 0183에서는 선박에서 발생하는 여러 데이터를 Sentence로 주고받는다. 가장 최근 버전인 Version 4.0에서는 이전 버전에서 사용되었던 것을 포함하여 약 100여 개가 넘는 Formatter를 제시하고 있다. 이러한 Formatter는 각각 다른 형식의 데이터를 전송할 수 있도록 해준다. **그림 2**

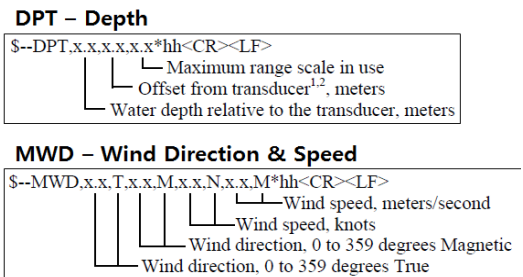


그림 2: DPT 및 MWD Formatter 형식

는 본 논문에서 사용하는 Formatter 중에서 DPT(수심)와 MWD(풍향-풍속)의 형식을 보여주고 있다.

3. 데이터베이스 서버 설계

3.1 데이터베이스 테이블 설계

그림 3은 NMEA 0183의 Sentence 형식으로 전송되는 데이터를 저장하기 위한 데이터베이스의 E-R Diagram이고, 그림 4는 장비로부터 발생하는 데이터 중 DPT 데이터가 저장되는 예를 나타낸 것이다.

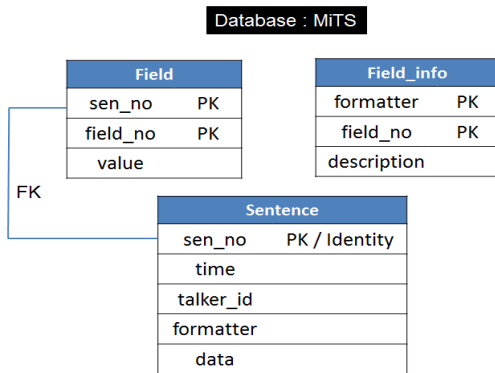


그림 3: 데이터베이스의 E-R Diagram

	sen_no	time	talker_id	formatter	data
Sentence	1	2010-09-17 13:42:29.817	SD	DPT	15.211,111.613,44.756

	sen_no	field_no	value
Field	1	1	15.211
	1	2	111.613
	1	3	44.756

	formatter	field_no	description
Field_Info	DPT	1	Water Depth relative to the transducer, meters
	DPT	2	Offset from transducer, meters
	DPT	3	Maximum range scale in use

그림 4: 데이터베이스에 저장되는 데이터의 예

Sentence 테이블은 각 장비에서 발생하는 데이터를 순차적으로 저장하기 위한 테이블이다. 이 테이블의 data 컬럼은 Sentence 형식에서 DPT와 MWD에 해당하는 Formatter 부분과 *hh(Checksum) 사이의 데이터 부분을 저장한 것이다. 하지만,

data 컬럼에 저장된 데이터는 응용프로그램에서 원하는 데이터를 바로 조회하기가 쉽지 않기 때문에 이 문제를 해결하고자 Field 테이블을 사용한다.

Field 테이블은 Sentence 테이블의 data 컬럼에 저장된 데이터를 콤마(comma) 단위로 분리하여 필드별로 저장함으로써 응용프로그램에서 좀 더 쉽게 이용할 수 있도록 하였다.

추가적으로, Field_info 테이블은 NMEA 0183 Sentence Formatter의 각 필드에 대한 상세 정보를 등록하여 응용프로그램에서 활용할 수 있도록 하였다.

표 1, 표 2, 표 3은 각 테이블의 컬럼이 의미하는 바를 정리한 것이다.

표 1: Sentence 테이블의 컬럼 및 의미

컬럼명	설명
sen_no	저장된 Sentence의 일련번호
time	Sentence가 저장된 시각
talker_id	Sentence가 발생한 장비의 ID
formatter	NMEA 0183 Sentence의 Formatter
data	장비에서 발생한 데이터

표 2: Field 테이블의 컬럼 및 의미

컬럼명	설명
sen_no	저장된 Sentence의 일련번호
field_no	Formatter에서의 필드 번호
value	필드의 실제 값

표 3: Field_Info 테이블의 컬럼 및 의미

컬럼명	설명
formatter	NMEA 0183 Sentence의 Formatter
field_no	Formatter에서의 필드 번호
description	필드에 대한 상세한 설명

3.2 데이터베이스 모듈 설계

데이터베이스 서버의 기능은 크게 저장과 검색의 두 부분으로 나누어질 수 있고, 그림 5가 그 구성을 보여주고 있다. 데이터 저장 모듈은 게이트웨이로부터 데이터를 수신하여 데이터베이스에 저장하는 역할을 하고, 데이터 검색 모듈은 특정 조건의 데이터를 데이터베이스로부터 검색하는 역할을 한다.

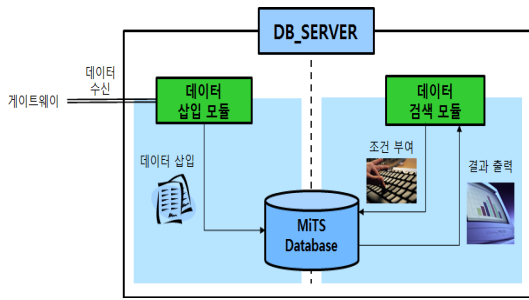


그림 5: 데이터베이스 서버의 구성

3.2.1 데이터 저장 모듈 설계

그림 6은 데이터 저장 모듈의 흐름을 나타낸 것이다. 각 장비들로부터 게이트웨이를 통하여 전송되어 온 데이터는 데이터베이스에 저장되기 위하여 우선 데이터베이스에 접속한다. 접속이 완료되면 소켓을 초기화하고 데이터 수신을 준비한다.

데이터 수신 준비가 완료되면 데이터를 수신하여 Sentence 및 Field 구조체의 각 변수에 맞게 저장하고, Insert_Sentence 및 Insert_Field 함수에 전달한다. Sentence 테이블과 Field 테이블 사이에 설정된 외래키(FK: Foreign Key) 관계에 따라 먼저 Insert_Sentence 함수가 Sentence 테이블에 저장하고, 후에 Insert_Field 함수가 Field 테이블에 저장한다. 위의 동작은 데이터가 수신되는 동안 계속 반복된다.

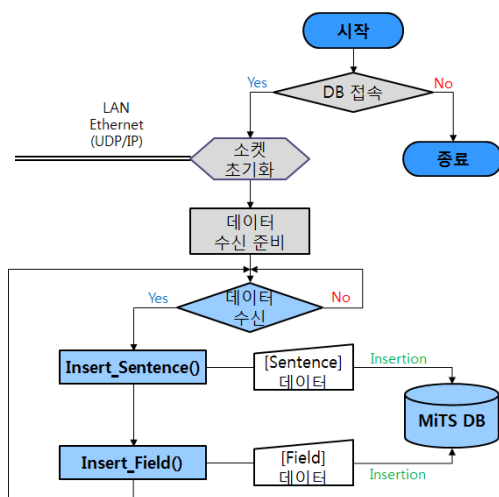


그림 6: 데이터 저장 모듈의 흐름

3.2.2 데이터 검색 모듈 설계

그림 7의 데이터 검색 모듈은 사용자가 부여한 데이터 타입, Formatter, 조회 기간, Interval 등의 특정 조건에 해당하는 데이터를 검색하여 결과를 출력한다.

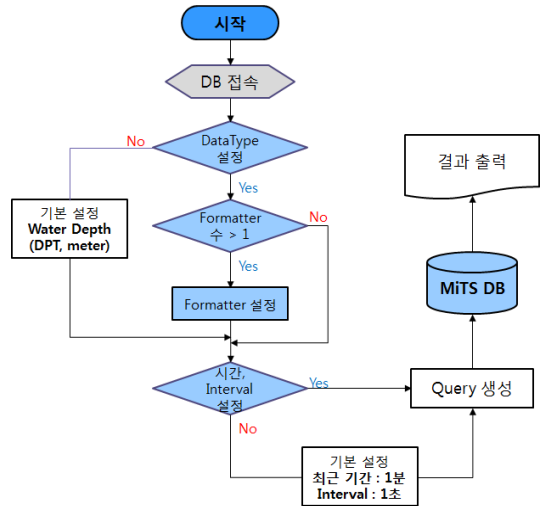


그림 7: 데이터 검색 모듈의 흐름

데이터 검색 모듈의 User Interface에서는 수심, 수온 등의 데이터 타입 및 단위, Formatter, 조회 기간, Interval 등의 조건을 선택 또는 입력

표 4: 특정 기간에 대한 초 단위별 수심 데이터를 조회하기 위한 SQL Query

```

select time, value as [water depth]
from sentence, field
where sentence.sen_no = field.sen_no
and formatter = 'DPT' and field_no = 1 and time in
(select min(time)
from sentence, field
where (formatter = 'DPT' and field_no = 1
and sentence.sen_no = field.sen_no)
and convert(char(19),time,120) in
(select distinct substrng(convert(char(19),time,120),1,19)
from sentence, field
where (time between '2010-09-17 15:10:16.267'
and '2010-09-17 16:24:50.746')
and formatter = 'DPT' and field_no = 1
and sentence.sen_no = field.sen_no)
group by convert(char(19),time,120))
order by time;
    
```

하도록 한다. 만약 선택 또는 입력하지 않으면 기본으로 설정된 조건으로 검색한다. 또한, 특정 Formatter의 데이터를 조회했을 때 같은 시간대에 2개 이상의 데이터가 존재한다면 먼저 발생한 데이터만 검색된다.

표 4는 특정 조건을 조회하기 위한 SQL Query 를 나타낸 것이다.

4. 구현 및 시험

4.1 개발 환경

구현에 사용된 운영체제는 Windows 7이며, 개발 도구와 언어는 Microsoft Visual Studio 2008의 C# 및 C++을 이용하였다. 또한 Chart FX 7 .NET 및 Chart FX 7 Gauges 컴포넌트를 사용하였다. DBMS(Database Management System)는 Microsoft SQL Server 2005를 사용하였다. 시험은 VGN-Z45LD 노트북과 PowerEdge 2950 서버를 이용하여 운영체제 Windows 7과 Windows Server 2008 상에서 수행하였다.

4.2 데이터 저장 모듈 구현 및 시험

그림 8은 임의로 발생시킨 데이터를 전송하여 이를 데이터베이스에 저장하는 과정을 나타낸 것이다. 데이터는 LAN을 통하여 전달되어 데이터베이스에 저장된다. 그림 9에서 그 저장 결과를 확인할 수 있으며 그림의 위쪽은 Sentence 테이블, 아래쪽은 Field 테이블을 의미한다.

```

MSQLConnect(>) Succeeded.
2010-09-17 15:28:49.471 [DPT] - Insertion.
2010-09-17 15:28:49.487 [MTW] - Insertion.
2010-09-17 15:28:49.502 [MWD] - Insertion.
2010-09-17 15:28:49.518 [RMC] - Insertion.
2010-09-17 15:28:49.533 [UHW] - Insertion.
2010-09-17 15:28:49.549 [VTG] - Insertion.
2010-09-17 15:28:49.611 [XDR] - Insertion.
2010-09-17 15:28:49.643 [DPT] - Insertion.
2010-09-17 15:28:49.689 [MTW] - Insertion.
2010-09-17 15:28:49.705 [MWD] - Insertion.
2010-09-17 15:28:49.767 [RMC] - Insertion.
2010-09-17 15:28:49.830 [UHW] - Insertion.
2010-09-17 15:28:49.877 [VTG] - Insertion.
2010-09-17 15:28:49.923 [XDR] - Insertion.
2010-09-17 15:28:49.939 [DPT] - Insertion.
2010-09-17 15:28:49.955 [MTW] - Insertion.
2010-09-17 15:28:49.970 [MWD] - Insertion.
2010-09-17 15:28:50.001 [RMC] - Insertion.
2010-09-17 15:28:50.033 [UHW] - Insertion.
2010-09-17 15:28:50.157 [VTG] - Insertion.
2010-09-17 15:28:50.360 [XDR] - Insertion.
2010-09-17 15:28:50.407 [DPT] - Insertion.
2010-09-17 15:28:50.438 [MTW] - Insertion.
    
```

그림 8: 데이터베이스 저장 과정

두 그림에서 표시된 시간을 비교해 볼 때 몇몇 데이터에 대해서 0.001초의 차이가 발생하는 것을 확인할 수 있는데 이는 데이터가 발생되고 저장되기까지 지연되는 시간을 의미한다. 이 지연시간은 데이터 검색 모듈에서의 최소 조회 간격이 초 단위로 설계되어 있기 때문에 검색 결과에 영향을 미치지 않는다.

sen_no	time	talker_id	formatter	data
3...	3865	2010-09-17 15:28:49.470	SD	DPT 15,211,111,613,44,756
3...	3866	2010-09-17 15:28:49.487	YX	MTW 61.5.C
3...	3867	2010-09-17 15:28:49.503	YX	MWD 62.699.T,173.036.M,64.052.N,261.698.M
3...	3868	2010-09-17 15:28:49.517	GP	RMC 123036.A,318.422.S,107.184.W,94.47,105.85,201007...
3...	3869	2010-09-17 15:28:49.533	HC	VHW 306.817.T,229.679.M,65.784.N,727.345.K
3...	3870	2010-09-17 15:28:49.550	WM	VTG 119.462.T,79.587.M,90.182.N,485.384.K.S
3...	3871	2010-09-17 15:28:49.610	YX	XDR H,127.65.P,Humidity
3...	3872	2010-09-17 15:28:49.643	SD	DPT 129.85,1.434,13.534
3...	3873	2010-09-17 15:28:49.690	YX	MTW 80.934.C
3...	3874	2010-09-17 15:28:49.707	YX	MWD 261.184.T,222.27.M,130.918.N,223.686.M
3...	3875	2010-09-17 15:28:49.767	GF	RMC 123036.A,228.644.S,223.146.W,57.82,78.11,2010072...
3...	3876	2010-09-17 15:28:49.830	HC	VHW 47.117.T,111.613.M,97.552.N,652.783.K
3...	3877	2010-09-17 15:28:49.877	WM	VTG 179.935.T,222.27.M,45.962.N,382.313.K.S
3...	3878	2010-09-17 15:28:49.923	YX	XDR H,154.56.P,Humidity
3...	3879	2010-09-17 15:28:49.940	SD	DPT 74.2,109.462,82.812
3...	3880	2010-09-17 15:28:49.957	YX	MTW 6.765.C

sen_no	field_no	value
2...	3865	1 15,211
2...	3865	2 111,613
2...	3865	3 44,756
2...	3866	1 61.5
2...	3866	2 C
2...	3867	1 62.699
2...	3867	2 T
2...	3867	3 173.036
2...	3867	4 M
2...	3867	5 64.052
2...	3867	6 N
2...	3867	7 261.698
2...	3867	8 M

그림 9: 데이터베이스 저장 결과

임의로 발생시킨 데이터뿐만 아니라, 실제로 선박에서 사용하는 장비인 수심-수온 측정 장비와 풍향-풍속 측정 장비를 네트워크에 연결한 상태에서도 시험하였다. 이러한 장비로부터 발생하는 실시간 데이터도 데이터베이스에 정상적으로 저장되는 것을 확인할 수 있었다.

4.3 데이터 검색 모듈 구현 및 시험

데이터베이스 검색 모듈은 사용자가 부여한 특정 조건에 따라 데이터를 검색하고 그 결과를 출력한다. 그림 10은 데이터 검색 모듈의 User Interface를 나타내며, 사용자는 자신이 검색하고자 하는 데이터 타입, 조회 기간, 조회 간격 등에 대하여 적절하게 조건을 부여할 수 있다. 그림 11은 '전체 기간에 대한 초 단위별 수심 데이터'라는 특정 조건을 부여하여 검색한 결과를 나타낸 것이다.

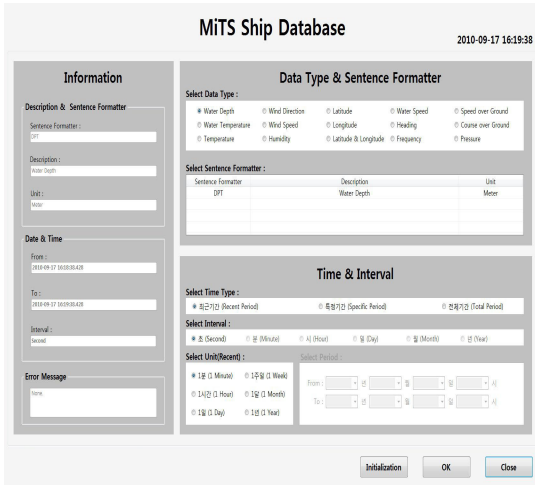


그림 10: 검색용 User Interface

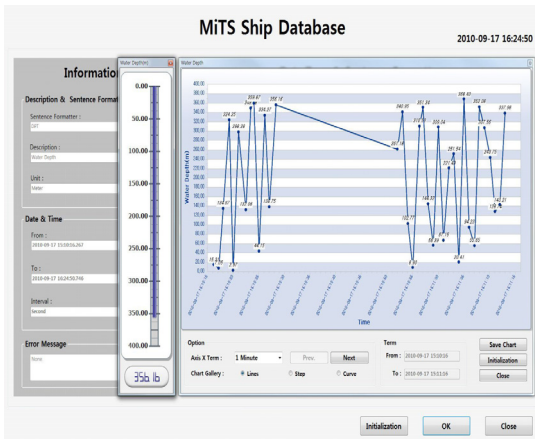


그림 11: 수심 데이터 검색 결과

특정 조건을 주었을 때 검색된 데이터의 시간적 변화를 그래프로 확인할 수 있었으며, 컴퓨터의 사양에 따라 다르지만 대체로 큰 지연 없이 그려지는 것을 확인할 수 있었다. 하지만, 한 번에 그릴 수 있는 데이터의 양이 컴퓨터의 사양에 따라 평균적으로 1~2만 개로 제한되는 것을 확인할 수도 있었다.

5. 결론 및 향후 과제

본 논문에서는 선박의 각 장비로부터 실시간으로 측정되어 전송되는 NMEA 0183 Sentence 형식의 데이터를 통합 관리할 수 있는 데이터베이스를

설계하였다. 이를 바탕으로 데이터 저장 및 검색 모듈을 구현하였으며, 시험을 통하여 데이터가 정상적으로 저장되고 사용자가 원하는 정보를 검색할 수 있음을 확인하였다.

향후 연구과제로, 데이터 발생량이 과다하여 데이터베이스 용량을 초과하는 경우에 오래된 데이터를 샘플링하거나 요약하여 2차 데이터베이스에 저장하는 기능을 추가할 예정이다.

후 기

본 연구는 지식경제부 및 한국산업기술평가관리원의 IT핵심기술개발사업의 일환으로 수행하였음. [2008-F-046-01, E-Navigation 대응 IT-선박 융합 핵심기술 개발]

참고문헌

- [1] 송병호, 박경우, 이진석, 이경효, 정민아, 이성로, “디지털 선박 내 다차원 센서 스트림 데이터의 효율적인 처리”, 한국통신학회논문지, 제 35권, 제5호, pp. 794-800, 2010.
- [2] 송병호, 최명수, 권장우, 이성로, “상황 인식 기반 해양 디지털 선박 상황 진단 시스템 구현 및 설계”, 한국통신학회논문지, 제35권, 제6호, pp. 859-866, 2010.
- [3] 김재동, 박수한, 김형진, 고성위, 정해중, “실습조사선의 종합정보통신망시스템 구축에 관한 연구”, 한국해양공학회지, 제18권, 제6호, pp. 44-50, 2004
- [4] 황훈규, 김태중, 윤진식, 서정민, 박휴찬, 장길웅, 이장세, “선박 내 통합 정보 서비스를 위한 미들웨어 서버의 서비스 모듈 설계 및 구현”, 한국마린엔지니어링학회지, 제34권, 제1호, pp. 141-146, 2010.
- [5] 해양수산부, 국제해사기구(IMO) 제53차 항해안전전문위원회(NAV) 회의 결과, 2007.
- [6] 이장세, 박휴찬, 장길웅, 이주형, 장남주, 이주영, 이부형, “선박 내 정보의 통합관리를 위한 정보 아키텍처”, 2009년도 공동학술대회논문집, 한국마린엔지니어링학회, 2009.
- [7] National Marine Electronics Association, NMEA 0183 Verion 4.0 : Standard for

Interfacing Marine Electronic Devices, 2008.

- [8] 정성훈, 김병찬, 양규식, “해상환경에서 IEEE 802.16e의 RSSI 및 CINR 측정 분석”, 한국항해학회논문지, 제13권, 제6호, 2009.

저 자 소 개



서정민(徐正民)

2009년 한국해양대학교 IT공학부 컴퓨터정보공학전공(공학사), 2009년-현재 한국해양대학교 대학원 컴퓨터공학과 석사과정. 관심분야: 데이터베이스, 데이터마이닝



황훈규(黃勳圭)

2009년 한국해양대학교 IT공학부 컴퓨터정보공학전공(공학사), 2009년-현재 한국해양대학교 대학원 컴퓨터공학과 석사과정. 관심분야: 정보보안, 네트워크, 시뮬레이션, 해양정보시스템



윤진식(尹珍植)

2009년 한국해양대학교 IT공학부 컴퓨터정보공학전공(공학사), 2009년-현재 한국해양대학교 대학원 컴퓨터공학과 석사과정. 관심분야: 정보보안, 네트워크, 포렌식



이성대(李聖大)

1999년 한국해양대학교 컴퓨터공학과(공학사), 2001년 한국해양대학교 대학원 컴퓨터공학과(공학석사), 2007년 한국해양대학교 대학원 컴퓨터공학과(공학박사), 2007년 - 2009년 한국해양대학교 산학협력단 전임연구원, 2009년 - 현재 한국해양대학교 전파공학과 연구교수. 관심분야: 데이터마이닝, 해양정보시스템



박휴찬(朴休讚)

1985년 서울대학교 전자공학과(공학사), 1987년 한국과학기술원 전기및전자공학과(공학석사), 1995년 한국과학기술원 전기및전자공학과(공학박사), 1987년 - 1990년 금성반도체, 1997년 - 현재 한국해양대학교 IT공학부(교수). 관심분야: 데이터베이스, 데이터마이닝, 해양정보시스템



이장세(李章世)

1997년 한국항공대학교 컴퓨터공학과(공학사), 1999년 한국항공대학교 컴퓨터공학과(공학석사), 2003년 한국항공대학교 컴퓨터공학과(공학박사), 2004년 - 현재 한국해양대학교 IT공학부(조교수). 관심분야: 컴퓨터보안, 지능시스템, 모델링 및 시뮬레이션



장길웅(張吉雄)

1997년 경북대학교 컴퓨터공학과 (공학사), 1999년 경북대학교 컴퓨터공학과(공학석사), 2002년 경북대학교 컴퓨터공학과(공학박사), 2003년 - 현재 한국해양대학교 데이터정보학과(부교수). 관심분야: 네트워크 프로토콜, 유비쿼터스 네트워킹