

---

# 데이터 샘플링을 이용한 선박용 다단 데이터베이스 설계 및 구현

서정민\* · 황훈규\* · 이성대\*\* · 이장세\*\* · 장길웅\*\* · 박휴찬\*\*\*

Design and Implementation of Multistage Database for Shipboard  
by using Data Sampling

Jeong-Min Seo\* · Hun-Gyu Hwang\* · Seong-Dae Lee\*\* · Jang-Se Lee\*\* · Kil-Woong Jang\*\* · Hyu-Chan Park\*\*\*

## 요 약

선박에 장착된 장비들을 통하여 수집된 다양한 데이터는 선박의 안전한 운항을 위하여 사용된다. 하지만 이러한 다양성으로 인해 데이터를 각각 관리하기에는 어려움이 있으므로 이를 통합 관리하기 위하여 데이터베이스를 구축하는 것이 효율적이다. 그러나 선박에서의 데이터베이스는 통상의 데이터베이스에 비해 급격한 저장 빈도 때문에 용량 초과 문제가 자주 발생한다. 본 논문에서는 이를 해결하고자 데이터의 샘플링과 삭제를 통하여 데이터베이스의 용량을 확보하면서 통합 관리가 가능한 다단 데이터베이스를 설계하고 구현한 후, 그 효율성을 시험한다.

## ABSTRACT

Several data from equipments on shipboard have been used for the safe navigation of ships. Due to the varieties of data, however, there may be difficulties to manage those data separately. Therefore, it can be effective to develop database for the integrated management of ship data. However, the database may become full soon because huge amount of data are saved continuously. To cope with this problem, this paper proposes multistage database which can always keep the database under full by means of sampling mechanism. To verify the functionality of the methodology, a prototype system has been implemented and tested.

## 키워드

다단 데이터베이스, 데이터 샘플링, 선박, IEC 61162-450, NMEA 0183

## Key word

Multistage Database, Data Sampling, Ship, IEC 61162-450, NMEA 0183

---

\* 준회원 : 한국해양대학교 대학원  
\*\* 정회원 : 한국해양대학교  
\*\*\* 정회원 : 한국해양대학교 (교신저자, hcpark@hhu.ac.kr)

접수일자 : 2011. 01. 11  
심사완료일자 : 2011. 02. 22

## I. 서 론

오늘날 선박은 점차 디지털화되고 있다. 이에 따라 선박 내에 있는 여러 장비로부터 측정된 전자적 데이터는 모니터링 시스템, 항행 시스템, 상황 예측 시스템 등에서 안전 운항을 위해 활용되고 있다[1,2,3]. 이와 같이 여러 시스템에서 효율적으로 활용되기 위해서는 측정된 데이터의 통합 관리를 위한 데이터베이스가 필요하다[4,5].

하지만 선박에서 데이터의 통합 관리를 위해 사용되는 데이터베이스는 통상의 데이터베이스에 비하여 데이터의 저장 빈도가 높기 때문에 용량이 초과하는 문제가 쉽게 발생할 수 있다. 하지만 일반적으로 선박에는 데이터베이스 관리자가 승선하지 않기 때문에 용량 초과 문제가 발생하더라도 즉시 해결할 수 없는 실정이다. 따라서 자동으로 과거의 데이터를 샘플링하고 삭제하는 기능을 갖는 데이터베이스 서버를 구축한다면 용량 초과 문제를 해결할 수 있으며 관리 비용도 줄일 수 있다.

본 논문은 선박에서 IEC 61162-450 표준[6]을 따르는 전자적 데이터의 통합 관리를 위한 다단 데이터베이스를 설계한다. 또한 데이터베이스 용량 초과 문제를 해결하기 위하여 데이터 샘플링 및 삭제 모듈을 구현한다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구에 대하여 기술한 후, 3장에서는 데이터의 통합 관리를 위한 다단 데이터베이스와 각 모듈에 대하여 설계한다. 4장에서는 각 모듈을 구현하고 다단 데이터베이스의 효율성을 시험하며, 5장에서 결론 및 향후 과제로 끝을 맺는다.

## II. 관련 연구

### 2.1. 선박 네트워크

선박에서 통합 네트워크에 사용될 수 있는 표준으로 IEC 61162-450이 있다. 이 표준은 선박 내의 여러 장비로부터 측정된 데이터의 상호 교환 및 통합 관리를 목적으로 제안된 국제적인 표준이다. 이 표준에서 채택한 데이터 표현형식은 NMEA 0183의 Sentence[7]를 준용한다. 이러한 표준에 따라 구성된 선박 네트워크에는 다양한 용도의 시스템이 존재할 수 있다.

본 논문에서는 이러한 선박 네트워크를 구성하는 여러 시스템 중 전자적 데이터를 효율적으로 관리하고 다른 시스템에서 활용할 수 있도록 지원하는 데이터베이스 서버에 관한 내용을 다룬다.

### 2.2. 데이터 샘플링

선박에서는 다양한 장비로부터 수많은 데이터가 지속적으로 발생된다. 이러한 데이터를 저장하기 위하여 데이터베이스를 사용하더라도 그 용량이 쉽게 한계에 도달하는 문제가 발생한다.

한계에 도달한 데이터베이스의 용량을 추가 확보하는 방법으로 두 가지를 고려해볼 수 있다. 첫째는 데이터베이스 용량을 증가시키는 방법이며, 둘째는 저장된 데이터의 일부를 삭제하는 방법이다. 데이터베이스 용량을 증가시키는 방법은 지속적인 추가 비용이 초래되고 데이터베이스 관리자의 배치가 필요하다는 단점이 있다. 데이터를 삭제하는 방법 또한 데이터가 소실되는 단점이 발생한다.

본 논문에서는 데이터베이스의 용량을 확보하기 위하여 데이터 삭제 방법을 사용하되, 데이터의 소실을 최소화하기 위하여 샘플링 기법을 도입한다. 데이터 샘플링 기법은 크게 세 종류가 존재하는데 그림 1은 그 예를 나열한 것이다.

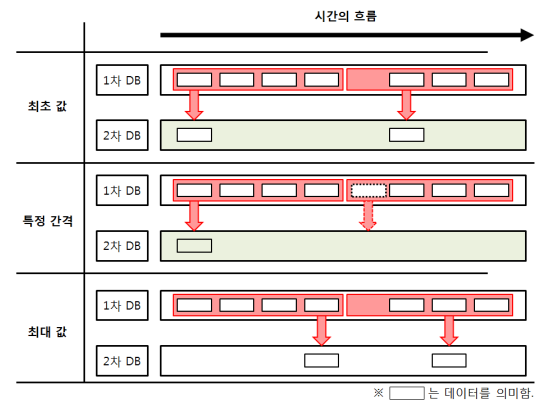


그림 1. 데이터 샘플링 기법의 예  
Fig. 1 Example of data sampling techniques

첫째, **최초 값 샘플링 기법**은 일정 간격 동안 저장된 데이터 중 가장 최초로 발생한 데이터를 샘플링한다. 둘째, **특정 간격 샘플링 기법**은 일정 간격마다 해당 시간에

발생한 데이터를 샘플링 한다. 셋째, **최대 값 샘플링 기법**은 일정 간격 내에 발생한 데이터 중 최대 값을 샘플링 한다.

하지만, 특정 **간격 샘플링 기법**은 해당 시간에 데이터가 발생하지 않으면 샘플링이 되지 않을 수 있으며, **최대 값 샘플링 기법**은 수치 데이터만 가능하기 때문에 본 논문에서는 **최초 값 샘플링 기법**을 사용한다.

### III. 다단 데이터베이스 및 모듈 설계

#### 3.1. 다단 데이터베이스의 구성

본 논문에서 제안하는 데이터 통합 관리 목적의 다단 데이터베이스 및 각 기능을 위한 모듈의 구성은 그림 2와 같다.

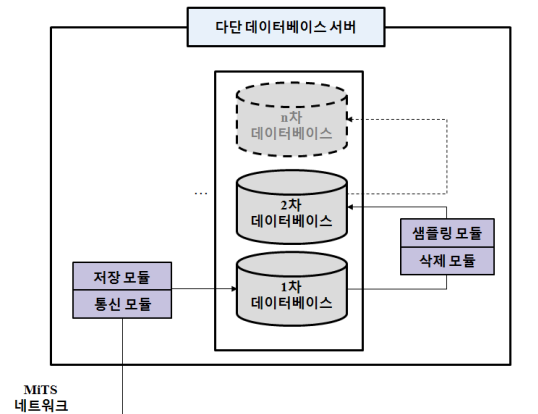


그림 2. 다단 데이터베이스의 구성  
Fig. 2 Architecture of multistage database

다단 데이터베이스는 선박 내의 여러 장비에서 측정된 데이터를 저장하는 1차 데이터베이스와 샘플링된 데이터를 백업하기 위한 2차 혹은 그 이상의 데이터베이스로 구성된다.

다단 데이터베이스를 구성하는 각 모듈 중 저장 모듈은 1차 데이터베이스에 데이터를 저장한다. 또한 샘플링 모듈 및 삭제 모듈은 n-1차 데이터베이스의 데이터를 샘플링 하여 n차 데이터베이스에 저장하거나, n-1차 데이터베이스의 용량을 확인하면서 데이터를 삭제한다. 본 논문에서는 2차 데이터베이스까지 설계하였으며, 따라

서 1차 데이터베이스만 샘플링 및 삭제의 대상이 된다.

#### 3.2. 다단 데이터베이스의 설계

그림 3은 NMEA 0183의 **Sentence** 형식으로 표현된 데이터를 관리하기 위한 다단 데이터베이스의 E-R 다이어그램을 나타낸 것이다.

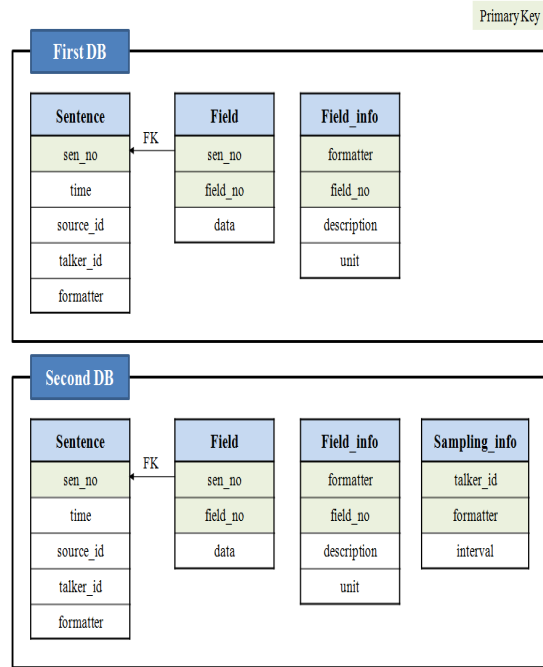


그림 3. 다단 데이터베이스의 E-R 다이어그램  
Fig. 3 E-R diagram of multistage database

선박 네트워크를 통하여 전달된 데이터는 1차 데이터베이스의 **Sentence** 테이블과 **Field** 테이블에 저장된다. **Sentence** 테이블은 각 데이터의 **Sentence**에 대한 정보를 저장하고, **Field** 테이블은 각 데이터의 세부 필드별 값을 저장한다. **Field\_info** 테이블은 NMEA 0183에 정의되어 있는 각 세부 필드에 대한 기본 정보를 가지고 있다.

샘플링된 데이터를 저장하는 2차 데이터베이스의 구조는 1차 데이터베이스와 유사하지만, 샘플링과 관련된 정보를 가지고 있는 **Sampling\_info** 테이블이 추가된다. 이러한 데이터베이스 테이블의 각 칼럼에 대한 설명은 표 1과 같다.

표 1. 데이터베이스 테이블의 칼럼과 그 의미  
Table 1. Column and description of database tables

칼럼명	설명
sen_no	저장된 Sentence의 번호. 자동 증가
time	Sentence가 저장된 시간
source_id	Sentence를 전송한 서버의 ID
talker_id	Sentence가 발생된 장비의 ID
formatter	NMEA 0183 Sentence 형식지정자
field_no	Sentence에서의 필드 번호
value	필드의 값
description	필드에 대한 상세한 설명
unit	필드 값의 단위
interval	샘플링 간격

### 3.3. 데이터 샘플링 모듈

데이터 샘플링 모듈은 데이터가 저장됨과 동시에 동작한다. 이는 데이터 삭제 시에 동작하면 한 번에 많은 데이터를 샘플링해야 하므로 부하가 클 수 있기 때문이다. 본 논문에서 1차 데이터베이스에 저장된 데이터를 샘플링하기 위한 모듈의 동작 과정은 그림 4와 같다.

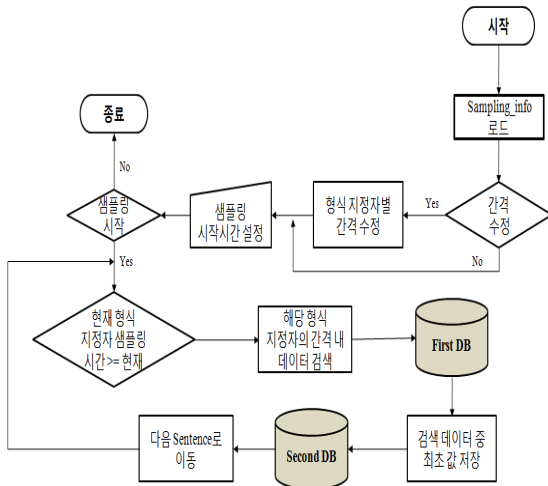


그림 4. 데이터 샘플링 모듈의 동작 과정  
Fig. 4 Process of data sampling module

샘플링 모듈이 실행되면 샘플링에 관한 정보를 *Sampling\_info* 테이블에서 읽어온다. 이 때, 저장되어 있던 각 형식지정자(formatter)별 샘플링 간격에 대하여 사용자가 원할 경우 수정이 가능하다. 간격 수정 및 샘플링 시작시간을 설정하고 샘플링을 시작하면 모듈은 각 형식지정자에 대하여 샘플링 시간을 확인한다. 해당 형식지정자에 대하여 샘플링이 가능한 시간이라면 간격 내에서 최초의 데이터를 1차 데이터베이스에서 추출하여 2차 데이터베이스에 저장한다. 샘플링이 불가능한 시간이라면 샘플링을 하지 않은 채 그 다음 형식지정자를 확인한다. 마지막 형식지정자의 확인이 끝나면 처음부터 다시 반복 실행한다.

### 3.4. 데이터 삭제 모듈

데이터베이스에 데이터가 계속 저장될 경우 용량 초과 문제가 발생하게 되는데 이 문제는 데이터를 삭제하여 해결할 수 있다. 이 모듈의 동작 과정은 그림 5와 같다.

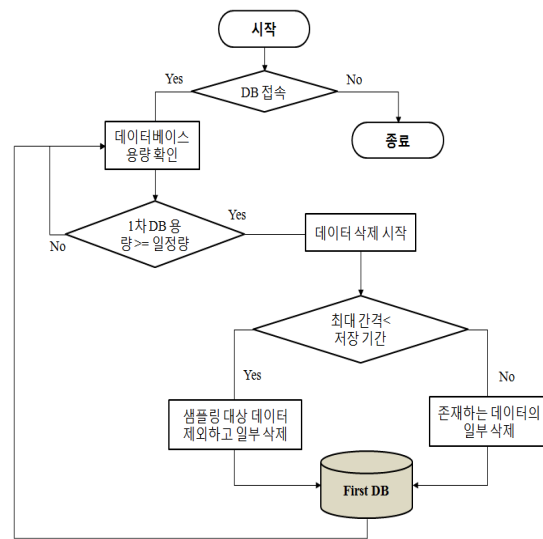


그림 5. 데이터 삭제 모듈의 동작 과정  
Fig. 5 Process of data deletion module

먼저 1차 데이터베이스의 데이터 저장량이 일정량을 초과하면 데이터 삭제를 시작한다. 본 논문에서는 데이터 저장량이 데이터베이스 크기의 90%를 초과하면 삭제를 시작한다. 이 비율이 작으면 불필요하게 빈번한 삭

제 동작이 일어나고, 크면 데이터베이스 용량이 초과될 가능성이 높아지게 된다.

데이터를 삭제할 때, 데이터를 얼마나 삭제하는 것이 적정인가에 대한 문제가 발생한다. 샘플링 모듈을 통해 데이터를 백업하지만 아직 샘플링이 되지 않은 데이터가 존재할 가능성이 적지 않다. 따라서 데이터 삭제 명령이 발생할 때, 샘플링 최대 간격만큼의 최근 데이터를 제외한 나머지 기간의 데이터의 일부를 삭제 대상으로 설정한다. 그러나 데이터베이스의 용량이 매우 작거나 샘플링 최대 간격이 매우 길 경우, 데이터가 저장된 기간보다 샘플링 최대 간격이 더 길 수 있는데 이 경우에는 저장된 전체 데이터의 일부를 삭제 대상으로 설정한다.

#### IV. 다단 데이터베이스의 구현 및 시험

##### 4.1. 구현 환경

다단 데이터베이스 및 각 모듈의 구현은 Windows 7에서 Visual Studio 2008 C#을 사용하였다. DBMS(Database Management System)로는 Microsoft SQL Server(MSSQL) 2005를 사용하였다.

##### 4.2. 데이터베이스 시험

그림 6은 NMEA 0183 Sentence가 저장된 1차 데이터베이스를 보여주고 있으며, Sentence 테이블과 Field 테이블을 조인한 결과이다.

sen_no	time	source_id	talker_id	formatter	field_no	value
1	2011-01-04 16:25:52.617	SD0001	SD	DPT	1	353.192
1	2011-01-04 16:25:52.617	SD0001	SD	DPT	2	228.484
1	2011-01-04 16:25:52.617	SD0001	SD	DPT	3	19.296
2	2011-01-04 16:25:52.803	YX0002	YX	MTW	1	28.948
2	2011-01-04 16:25:52.803	YX0002	YX	MTW	2	C
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	1	278.992
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	2	T
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	3	26.29
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	4	M
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	5	83.348
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	6	N
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	7	102.34
3	2011-01-04 16:25:54.190	YX0003	YX	MWD	8	M

그림 6. 데이터베이스의 데이터 저장 결과  
Fig. 6 Data storage result of database

그림에서 알 수 있듯이 Sentence 번호(*sen\_no*)가 같은 데이터에 대하여 Sentence 테이블은 Sentence에 대한 정보를, Field 테이블은 필드별 데이터를 저장하였음을 확인할 수 있다. 또한 여러 형식지정자가 동시에 저장되는 것을 확인함으로써 데이터가 통합 관리되는 것을 확인할 수 있다.

##### 4.3. 데이터 샘플링 모듈 시험

그림 7은 1차 데이터베이스에서 데이터를 샘플링하는 동작을 보여주고 있다.

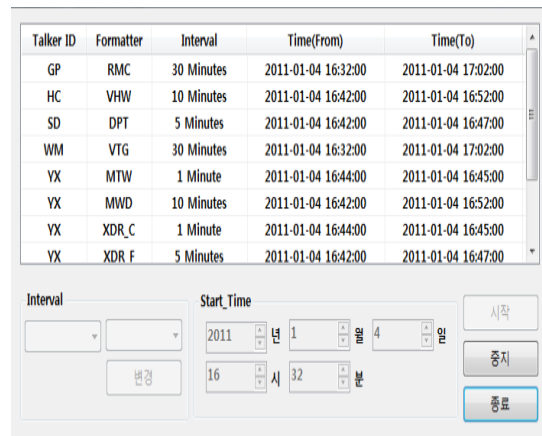


그림 7. 데이터 샘플링 모듈의 동작  
Fig. 7 Operation of data sampling module

sen_no	time	source_id	talker_id	formatter	field_no	value
1	2011-01-04 16:32:05.953	YX0002	YX	MTW	1	29.126
1	2011-01-04 16:32:05.953	YX0002	YX	MTW	2	C
2	2011-01-04 16:32:07.300	YX0008	YX	XDR_C	1	C
2	2011-01-04 16:32:07.300	YX0008	YX	XDR_C	2	15.374
2	2011-01-04 16:32:07.300	YX0008	YX	XDR_C	3	C
2	2011-01-04 16:32:07.300	YX0008	YX	XDR_C	4	Temperature
3	2011-01-04 16:33:00.197	YX0002	YX	MTW	1	28.964
3	2011-01-04 16:33:00.197	YX0002	YX	MTW	2	C
4	2011-01-04 16:33:01.693	YX0008	YX	XDR_C	1	C
4	2011-01-04 16:33:01.693	YX0008	YX	XDR_C	2	16.127
4	2011-01-04 16:33:01.693	YX0008	YX	XDR_C	3	C
4	2011-01-04 16:33:01.693	YX0008	YX	XDR_C	4	Temperature

그림 8. 데이터 샘플링 결과  
Fig. 8 Data sampling result

그림 8은 샘플링 결과의 일부를 나타낸 것으로서, 그림 7의 샘플링 과정을 통하여 1차 데이터베이스에서 샘플링 된 데이터는 2차 데이터베이스에 저장된다. 그림에서 알 수 있듯이, 샘플링 시작 시간부터 샘플링이 시작되며, 1분 간격으로 MTW 및 XDR\_C 데이터가 샘플링 된 것을 확인 할 수 있다.

#### 4.4. 데이터 삭제 모듈 시험

데이터 삭제 모듈은 데이터가 일정량 이상 저장되면 삭제를 시도한다. 본 논문에서는 데이터베이스의 용량을 200MB으로 설정한 후 삭제 모듈의 동작을 시험하였다. 그림 9는 삭제 대상 기간을 계산하는 과정을 보여주 고 있다.

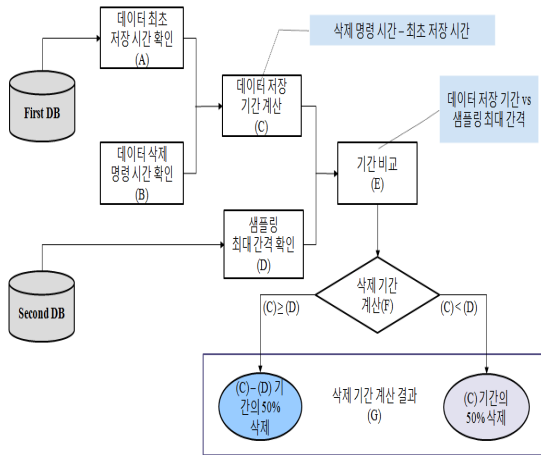


그림 9. 삭제 기간 계산의 흐름도  
Fig. 9 Calculation of deletion period

그림 10은 6시간 동안 데이터베이스의 용량 변화를 나타낸 그래프로서, 한 번 데이터가 삭제 될 때 평균 26% 정도의 용량을 확보하는 것을 알 수 있다. 이 때 데이터베이스의 용량이 크거나 삭제 대상 기간내 데이터의 양이 많으면 확보할 수 있는 데이터베이스의 용량도 커진다.

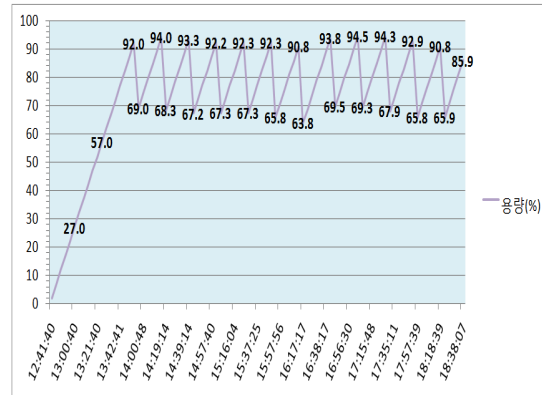


그림 10. 시간에 따른 용량 변화  
Fig. 10 Capacity change with time

## V. 결론 및 향후 과제

본 논문에서는 선박에 장착된 각 장비로부터 수집된 데이터를 통합 관리하기 위한 다단 데이터베이스를 설계하고 구현하였다. 또한 선박 데이터베이스의 특성상 자주 발생하는 용량 초과 문제를 해결하기 위하여 데이터 샘플링 모듈 및 삭제 모듈을 구현하였다. 임의의 데이터 및 실제 데이터를 이용하여 시험한 결과, 샘플링 및 삭제를 통하여 용량 초과 문제를 해결하는 것을 확인하였다.

결과적으로, 다단 데이터베이스 및 각 모듈은 선박에서 안정적이고 지속적인 데이터 통합 관리를 가능하게 하고 데이터베이스의 관리 비용을 절감할 수 있을 것이다.

향후 과제로는 데이터 샘플링뿐만 아니라 데이터 요약 기법에 대한 연구가 필요하다. 즉, 사용자가 데이터베이스에 저장된 실제 데이터 외에 온도의 월별 평균값과 같은 요약 데이터를 원할 수도 있다. 따라서 이러한 요약 데이터를 데이터베이스화 한다면 사용자는 직접 계산을 하지 않고도 쉽게 원하는 요약 데이터를 얻을 수 있을 것이다.

참고문헌

- [1] 임용근, “IT 기반 디지털 선박 국내 개발 본격화”, 해양한국, pp. 104-107, 2002.
- [2] 김도연, 조대운, 이미라, 박계각, “정보융합 기술 기반의 지능형 항행안전정보 시스템”, 한국지능시스템학회 2009년도 추계학술대회 학술발표논문집, 한국지능시스템학회, 제19권, 제2호, pp. 206-209, 2009.
- [3] 박정호, 진광자, 김재명, 유대승, 오문균, 임동선, “조선 IT 현황과 전망”, 전자통신동향분석, 제25권, 제4호, 한국전자통신연구원, pp. 19-26, 2010.
- [4] 박휴찬, 이장세, 장길웅, 이정우, 정희섭, 박중현, 강순열, “선박에서의 통합 정보처리를 위한 시스템 아키텍처”, 2009년도 한국마린엔지니어링학회 공동 학술대회논문집, 한국마린엔지니어링학회, pp. 207-208, 2009.
- [5] 서정민, 황훈규, 윤진식, 이성대, 박휴찬, 이장세, 장길웅, “선박에서 데이터의 통합 관리를 위한 데이터베이스의 구현 및 설계”, 한국마린엔지니어링학회지, 제34권, 제8호, pp. 1188-1194, 2010.
- [6] IEC 61162-450 Ed.1 : Maritime Navigation and Radio-communication Equipment and Systems - Digital Interfaces - Part 450: Multiple Talkers and Multiple Listeners - Light-Weight Ship Systems Interconnection, 2010.
- [7] National Marine Electronics Association, NMEA 0183 Version 4.0 : Standard for Interfacing Marine Electronic Devices, 2008.

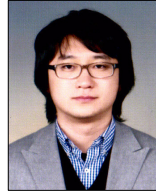
저자소개



서정민(Jung-Min Seo)

2009년 한국해양대학교 IT공학부 (공학사)  
 2011년 한국해양대학교 대학원 컴퓨터공학과(공학석사)

2011년~현재 삼원FA(주) 연구원  
 ※ 관심분야: 데이터베이스, 해양정보시스템



황훈규(Hun-Gyu Hwang)

2009년 한국해양대학교 IT공학부 (공학사)  
 2011년 한국해양대학교 대학원 컴퓨터공학과(공학석사)

2011년~현재 한국해양대학교 대학원 컴퓨터공학과 박사과정  
 ※ 관심분야: 정보보안, 시뮬레이션, 네트워크, 해양정보시스템



이성대(Seong-Dae Lee)

1999년 2월 한국해양대학교 컴퓨터공학과(공학사)  
 2001년 2월 한국해양대학교 대학원 컴퓨터공학과(공학석사)

2007년 한국해양대학교 대학원 컴퓨터공학과 (공학박사)  
 2007년~2009년 한국해양대학교 산학협력단 전임연구원  
 2009년~현재 한국해양대학교 전과공학과(연구교수)  
 ※ 관심분야: 데이터마이닝, 해양정보시스템



이장세(Jang-Se Lee)

1997년 한국항공대학교 컴퓨터공학과(공학사)  
 1999년 한국항공대학교 컴퓨터공학과(공학석사)

2003년 한국항공대학교 컴퓨터공학과(공학박사)  
 2004년~현재 한국해양대학교 조교수  
 ※ 관심분야: 컴퓨터보안, 지능시스템, 모델링 및 시뮬레이션



장길웅(Kil-Woong Jang)

1997년 경북대학교 컴퓨터공학과 (공학사)  
 1999년 경북대학교 컴퓨터공학과 (공학석사)

2002년 경북대학교 컴퓨터공학과(공학박사)  
 2003년~현재 한국해양대학교 부교수  
 ※ 관심분야: 네트워크 프로토콜, 유비쿼터스 네트워크





**박휴찬(Hyu-Chan Park)**

1985년 서울대학교 전자공학과  
(공학사)

1987년 한국과학기술원 전기 및  
전자공학과(공학석사)

1995년 한국과학기술원 전기및전자공학과(공학박사)

1987년~1990 금성반도체

1997년~현재 한국해양대학교 교수

※ 관심분야: 데이터베이스, 데이터마이닝, 해양정보  
시스템