

# A Research on Development of Bills of Material Using Web Grid for Product Lifecycle Management

Ji-Hyun Yoo\*

## Abstract

PLM(Product Lifecycle Management) is an information management system that can integrate data, processes, business systems and human resources throughout the enterprise. BOM(Bills Of Material) is key data for designing, purchasing materials, manufacturing planning and management, which is basic for product development throughout the product life cycle.

In this paper, we propose the efficient system to increase the data loading speed and the processing speed when using such large BOM data. We present the performance and usability of IMDG (In Memory Data Grid) for data processing when loading large amounts of data. In the UI, using the pure web grid of JavaScript instead of the existing data loading method can be improve performance of data managing.

▶ Keyword: BOM, PLM, Web Grid

## I. Introduction

기업의 경쟁력에 있어서 다양한 고객의 요구를 받아들여 제품을 개발하는 것과 개발 프로세스의 향상에 있어서 PLM(Product Lifecycle Management)의 활용은 단순한 제품 개발 뿐만 아니라, 미래 가치 창출에도 중요한 역할을 하고 있다.

PLM은 제품 수명주기 관리를 의미하며 기업 전반에 걸쳐서 데이터, 프로세스, 비즈니스 시스템뿐만 아니라 인력자원까지 모두 통합할 수 있는 정보 관리 시스템이다. PLM을 통해 설계에서 개발 및 제조, 폐기에 이르기까지 아이디어를 포함한 제품의 전반적인 정보를 효율적으로 관리할 수 있다. 여기에서 BOM(Bills Of Material)은 제품 수명주기 전체에 걸쳐서 제품 개발의 기준이 되는 정보로 설계, 재료 구매, 제조 계획 및 관리 등의 핵심데이터이다. BOM은 제품의 부품 명세와 생산, 개발, 판매 시에 소요되는 자재, 원재료 등을 정리한 리스트로, 부품들 간의 관계를 정의하는 데이터이기도 하여 PLM 시스템 구축에 있어서 중요한 역할을 한다.[2]

본 논문에서는 이러한 대용량의 BOM 데이터 활용 시에 데이터 로딩 속도 및 처리 속도를 향상시킬 수 있는 방안에 대해 제시한다.

대용량 데이터를 로딩할 때 IMDG(In Memory Data Grid)를 이용해 데이터를 처리하는 방안에 대해 성능 및 활용 용이성을 제시한다. UI 부분에서는 자바스크립트의 순수 웹 그리드를 사용하여 기존의 데이터 로딩과 비교하여 빠른 속도를 제안하였다. 이를 토대로 특정 분야에서의 PLM 뿐만 아니라 다양한 분야의 PLM에서도 활용될 수 있는 BOM을 적용할 수 있다.

## II. Industrial PLM(Product Lifecycle Management)

많은 기업들이 PLM을 도입하여 활용하고 있고, 산업별 특성과 요구사항은 매우 다양하다.

---

• First Author: Ji-Hyun Yoo, Corresponding Author: Ji-Hyun Yoo  
\*Ji-Hyun Yoo (jihyun\_yoo@jangan.ac.kr), Dept. of Internet Communication, Jangan University  
• Received: 2017. 11. 13, Revised: 2017. 11. 27, Accepted: 2017. 12. 11.  
• This work was supported by Jangan University.

### 1. Plant

기계, 전기 전자, 조선 등은 부품을 조립하는 방식으로 제품을 생산하는 반면에 플랜트 산업은 물질을 가공하는 방식으로 생산한다. 설계 및 제작, 발주처가 지역적으로 떨어져 있고, 부가가치가 높은 설계나 프로젝트 관리 등에 대한 부분은 선진국에서 담당하는 경우가 많다 보니 협업이 매우 중요하다. 설계가 완료되기 전에 시공 및 제작에 착수하다 보니 잦은 설계변경에도 정확한 정보가 관리될 수 있도록 설계와 시공 정보를 공유하는 관리 시스템이 되어야 한다.

프로젝트 준비기간, 전체 시공 기간을 예측하고 관리하는 일정관리가 중요하고, 자재 정보관리, 주계약자와 부계약자들의 통합적인 관리 시스템이 필요하다.

### 2. Shipbuilding, Aviation Industry

선박이나 항공 산업은 설계 요구사항부터 자재의 구입, 제작 및 조립하는 형태로 이루어진다. 설계 단계부터 제품 사양이 반영되어야만 하는 특징 때문에 미리 예측하여 생산하기 어렵고, 납품에 소요되는 시간이 타 산업에 비해 매우 길다.[7] 기업 간 협업이 모든 프로세스 상에서 발생하며 최종 생산 제품의 수가격적 이를 기준으로 예측과 계획을 세워야 한다.

### 3. Electronics Industry

전자제품은 항공이나 조선에 비해서는 비교적 간단하지만 제품 설계기간과 수명주기가 짧다. 그리고 하나의 제품에서 소비자 환경과 지리적인 환경에 따라 매우 다양한 제품의 모델이 가능하다. 이런 이유로 제품 사양에 따라서 자동적으로 구성되는 변경부분에 대한 관리가 필요하고, 빠른 설계변경을 위해서 전체 일괄변경이나 일부 일괄변경의 기능이 제공되어야 한다.

전자제품에서는 소프트웨어가 차지하는 비중이 크고, 이러한 소프트웨어도 하나의 부품으로 인식하여 프로그래밍 소스 코드 및 실행 버전의 관리의 중요성도 커지고 있다.

### 4. Consumer Goods Industry

소비재 산업에는 식음료를 비롯해 각종 세제류, 화장품 등이 포함되는데, 다른 산업에 비하여 시장 환경과 고객의 취향에 민감하기 때문에 변화에 신속하게 대응할 수 있어야 한다. 소비재 산업은 제품의 포물라나 레시피 관리, 매출에 큰 영향을 미치는 포장 및 패키지 관리 등이 차지하는 비중이 크고, 제품에 대한 아이디어 관리가 중요하다.[6]

탁기, 컴퓨터, 자동차 등은 모두 수많은 부품의 조립을 통해 만들어진 것이고, 예를 들어 안경 같은 아주 단순한 제품일지라도 십여 개의 부품으로 구성되어 있음을 알 수 있다.

BOM(Bill of Material)은 PLM 구축에 있어서 매우 중요하고, 특정 제품이 어떠한 부품으로 구성되는지에 대한 정보를 담고 있으므로 제품 개발의 기준 정보가 된다. BOM은 생산, 개발, 판매 시에 소요되는 자재, 원재료 등을 정리한 리스트로, 부품들 간의 관계를 정의하는 데이터라고 할 수 있다. BOM을 데이터베이스에 저장하기 위해서는 BOM 데이터베이스의 개념 구조 설계를 해야 하는데, 기본적으로 필요한 데이터는 부모 아이템, 자식 아이템, 컴포넌트 수로 구성되어야 할 것이다. 이렇게 저장한다면, 어떤 제품이 어떠한 부품들로 구성되는 지를 파악할 수 있다.

제품의 형상 또는 제품 구조를 의미하는 BOM은 제품 수명 주기의 각 단계마다 다른 형상과 구조를 가지게 된다. 개념 단계에서는 개념형상(As-Concept Configuration) 또는 개념 BOM(Concept BOM), 개발 단계에서는 개발형상(As-Designed Configuration), 개발 BOM 또는 기술 BOM(Engineering BOM)이라고 한다. 생산단계에서는 생산형상(As-Built Configuration) 또는 MBOM (Manufacturing BOM), 그리고 운영 단계에서는 운영형상(As-Maintained Configuration) 또는 서비스 BOM이라고 한다. 전사적으로 공동 활용할 수 있는 기업 전체의 정보 관리는 BOM을 EBOM(Enterprise BOM)이라고 한다.

정보의 신뢰성 확보와 공유, 공장별 정보와 실물의 일치를 통하여, 업무 운영 효율 증대 및 최적화된 재고 관리 유지가 BOM 정보관리 및 개선의 주요 목표이다.

BOM은 제품 또는 부품을 만드는데 소요되는 모든 부품의 주종 관계, 소요량, 공정 및 작업 방법 등의 정보를 가지고 있으며, 다음 표1과 같이 용도 및 구성 방법에 따라 다양한 형태로 표현된다.

Table 1. Types of BOM

Purpose	Planning BOM	• use demand and production planning
	Engineering BOM	• use products planing of functions • connected MBOM
	Manufacturing BOM	• on product production consider types of supply and process • on productivity • connected ERP
	Process BOM	• use method of construction and process planning
	Service BOM	• use service as A/S
Configu-ration	Static BOM	• directly specify the relation of parent and child • no conditional branch
	Configurable (Dynamic) BOM	• create dynamic structure on a variety of specification • verified by constraints of specification

## III. The Way to Improve of BOM

우리가 접하는 대부분의 제품들은 모두 여러 가지 부품의 조립품이라고 볼 수 있다. 예를 들어, 오디오, 비디오, 냉장고, 세

그림 1에서와 같이, 기업에서 도입하여 업무를 효율적으로 관리하고 처리해 주는 다양한 시스템들 사이에서 전사기준정보인 BOM은 조직 간 의사소통에 필수적인 역할을 담당하고 있으며, 오류발생 시 PDM(Product Data Management) 뿐만 아니라 BOM을 활용하는 전사 모든 시스템들이 부정확한 정보를 생성시키게 된다. BOM 생성단계에서 BOM은 제품의 구성 정보와 개발 이력을 포함하고, 제품 개발에 있어서 의사소통의 단위로 통합 제품 정보 관리의 기준이 된다.

BOM은 생산 계획의 기본 단위로 자재 소요 계획의 필수 입력 요소로 공정, 공법 설계 시에는 기준 정보로 활용되고, 자재 발주 및 작업 지시의 입력 요소, 제품의 서비스 관리 대상이 되며 설계 및 제조원가 산정 시에 기준이 된다.

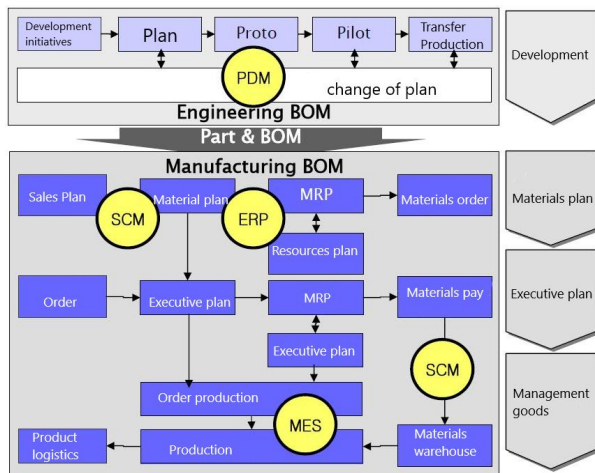


Fig. 1. BOM Operation

기업 내, 외부 활동 과정에서 BOM 이슈는 지속적으로 발생하고 있으며 제품의 개발 및 제조비용의 증가 요인이 되고 있다. 제품 스펙의 표준화가 미흡하여 BOM 정보의 조기 생성이 어렵고, 전산 정보와 실물 정보가 다르거나 설계 변경 정보가 공유되지 않아 정보 불일치, 누락 등이 발생하고 BOM 변경 사항이 공유되지 않는 등 문제점이 발생한다. 정보의 신뢰성 확보와 공유, 현장 별 정보와 실물의 일치를 통하여 최적화된 관리유지를 BOM 정보 관리 및 개선으로 해결할 수 있다.

세계화, 실시간 기업 등의 대외적 환경 변화에 대한 대응력 확보와 더불어, BOM이 기업 운영의 중요한 요소가 됨에 따라, 차세대 글로벌 BOM 개념이 주요 현안이 되고 있다. BOM 관련 당면 과제는 운영측면에서 프로세스 및 기준 정보표준화, BOM 활용 성능 및 정보 신뢰도 향상이, BOM 구조 측면에서는 현 구조의 경영 적합성 여부, 제품 사양의 적정성 등이다.

개선방향은 전체적으로 정보의 통합 일원화 관리를 지향하면서 시스템, 프로세스, 조직 측면에서 정보의 동기화, 표준화, 전문화를 이루고 구조 측면에서는 구성의 유연성 확보가 중요한 요소이다. 설계 BOM의 성능 및 UI 등 제반 기능 개선과 제조 BOM과의 정합성 향상, 목적별 BOM UI 제공, 다양한 조건별 소요량 산출 등 BOM 업무 전반의 개선이 필요하다.

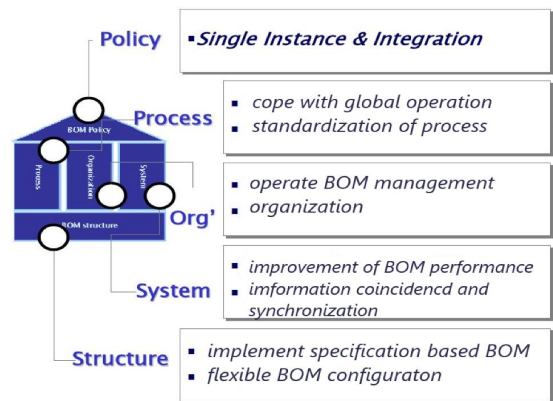


Fig. 2. The Way to Improve of BOM

#### IV. BOM Using Web Grid

앞에서 언급한 바와 같이 BOM은 제품 개발의 핵심 절차이고 이를 개선하기 위한 BOM 관리 시스템을 제안한다. 설계, 개발, 구매 단계에서 발생하는 부품정보(BOM)를 관리하기 위해 설계도면 파일과 그에 따른 부품 정보를 PLM에 구축된 표준화된 데이터베이스로 통합 관리하고, 기존의 ERP, SCM, PDM 시스템의 데이터와 연계하여 다양한 제품별 BOM 구조를 출력할 수 있도록 BOM Report 양식을 관리하여 설계하였다.

표준화된 데이터베이스 테이블을 통합 생성하고, 제품, 부품, 중간 모듈 등의 다양한 정보를 효과적으로 관리하기 위한 DB 구조를 설계하는 것이 핵심 요소이다. 해당 데이터는 BOM 에디터, 설계변경, ERP 인터페이스 등의 다양한 기능 구현을 위한 기본정보로 활용되므로 데이터의 안정성과 검색 속도를 향상시킬 수 있는 구조로 설계되어야 한다. 제품을 구성하는 부품 및 BOM 정보를 체계적인 관리하면서 부품 제조사별 서로 다른 정보와 업무프로세스에 맞춤형 BOM 에디터를 제공함으로써 업무편의성과 정보의 정합성이 보장된다.

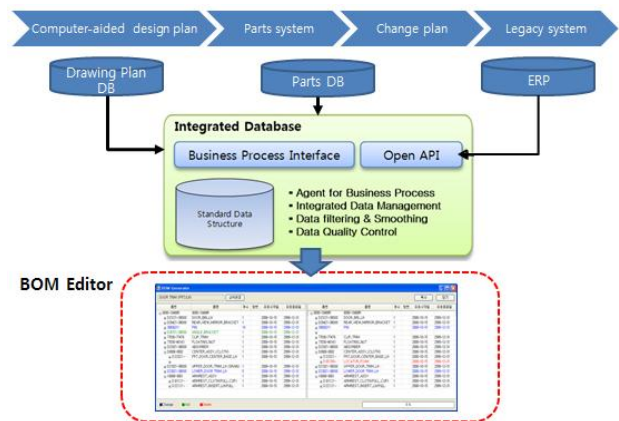


Fig. 3. Customized BOM Editor

컴퓨터의 자원사용률은 서버에서 클라이언트로 전송되는 데이터의 양에 비례하여 메모리가 변화 하게 된다. 최근 마이크로소프트의 엣지, 구글의 크롬 등 브라우저 JavaScript 엔진 성능의 향상으로 자원 사용 효율성이 더욱 높아지고 있다.

제품 구조와 부품 정보관리를 위한 BOM 에디터를 구현함에 있어 효율과 정합성을 향상시키기 위해 데이터의 로딩속도가 큰 이슈 중 하나이다. 데이터 그리드는 빅 데이터 처리 기술을 응용한 그리드(Grid)로 표현되는 영역만을 컨트롤하는 기술을 활용하여 처리속도를 높일 수 있다. 대용량 데이터를 웹에서 처리한다는 것은 단순히 로딩 만의 문제는 아니다. 읽은 데이터를 브라우저 상에서 빠르게 스크롤하거나 정렬, 그룹핑 할 수 있어야 하는데 다양한 웹 그리드 제품이 존재한다. UI의 핵심 요소인 그리드는 화면 개발에서 다중 레코드를 처리하는 개발 도구로 특히, 국내 환경에서 많이 사용되고 있다. 웹 개발에서 화면 개발이 70%라면, 화면 개발 중 그리드의 비중이 대부분을 차지할 정도로 비중이 대단히 높다.

기존의 테이블을 이용하는 방법보다 웹 그리드를 활용함으로써 BOM 데이터 로딩 시에 수분이 소요되던 것을 십여 초로 단축할 수 있다. 레이지 로딩(Lazy Loading) 기술은 우선 화면에 보이는 영역만을 로딩하고 스크롤 이동 시 추가적으로 로딩하는 기술이고, 다이내믹 스크롤링(Dynamic Scrolling) 기술은 데이터 스크롤 시에 사용자의 눈에 보이는 모든 영역을 변경 하지 않고 DOM(Document Object Model)의 순서를 변경하고 변경 된 영역만 데이터를 반영하여 화면 갱신을 최소화 하는 기술이다. 이러한 기술을 지원하는 JavaScript 기반의 웹 그리드 제품을 활용하여 아래 그림 3과 같이 대용량 데이터의 로딩 속도를 개선할 수 있다.

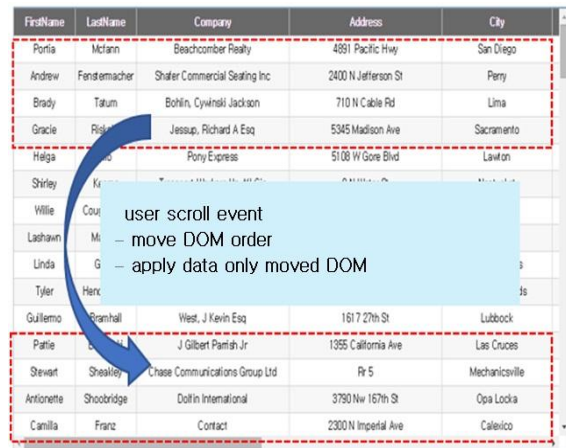


Fig. 5. Loading of BOM Data Using Dynamic Scrolling of JavaScript Web Grid

### V. BOM Using IMDG

IMDG(In Memory Data Grid)는 메인 메모리에 대용량의 데이터를 저장하고 데이터에 대한 일치성과 가용성을 보장하여 데이터 로딩 시에 높은 성능을 구현할 수 있다. 메인 메모리에 데이터를 저장하기 때문에 캐시나 MMDB(Main Memory DB)와 비슷하나, 캐시에 저장된 데이터는 대한 읽기 성능 개선만 가능하고, MMDB는 SQL을 통해 관리된다.

이미 많은 IMDG 제품이 상용이나 오픈소스로 존재한다. 다음은 주로 사용되는 제품의 일부이다.

Hazelcast (<http://www.hazelcast.com>)

JBoss Infinispan (<http://www.jboss.org/infinispan>)

GridGain DataGrid (<http://www.gridgain.com/features/in-memory-data-grid/>)

VMware Gemfire (<http://www.vmware.com/nl/products/application-platform/vfabric-gemfire/overview.html>)

Oracle Coherence (<http://www.oracle.com/technetwork/middleware/coherence/overview/index.html>)

Gigaspaces XAP (<http://www.gigaspaces.com/datagrid>)

Terracotta Enterprise Suite (<http://terracotta.org/products/enterprise-suite>)

기존의 데이터베이스의 데이터 수용 제약은 대략 1TB 정도이며 OLTP(OnLine Transaction Processing) 데이터 연산 성능은 잘 증대되지 않는다. 만약 서버가 메인메모리를 1TB나 그 이상을 쓰는 일이 일반적이 된다면, 적어도 OLTP 분야에서는 전체 데이터를 메인메모리에 넣고 운영을 수행할 수도 있다.

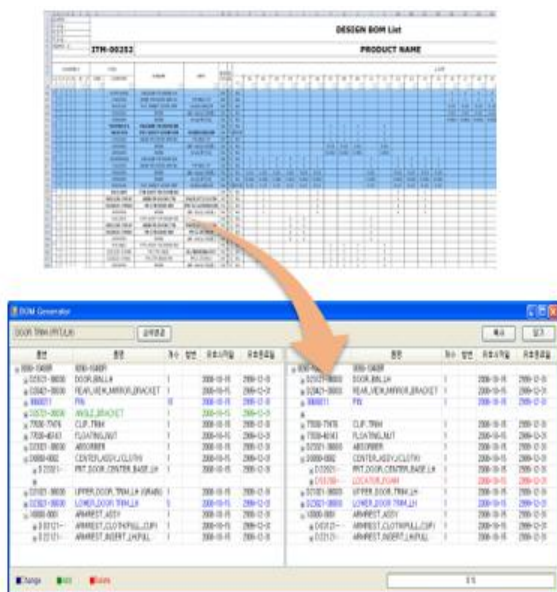


Fig. 4. Loading of BOM Data Using Lazy Loading of JavaScript Web Grid



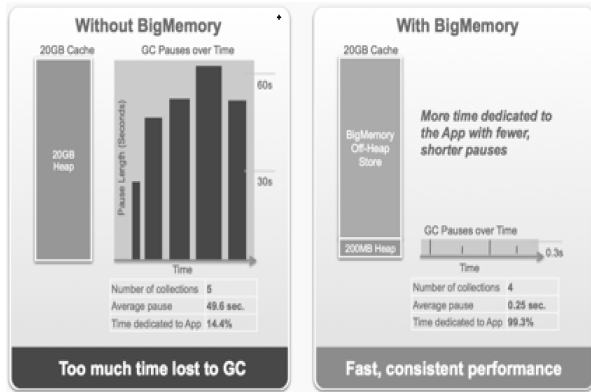


Fig. 6. Comparison IMDG[9]

메인 메모리를 저장장치로 사용하기 위해서는 두 가지 문제점을 극복해야 한다. 서버의 메인 메모리의 수용력을 초과하는 데이터가 포함된 경우와 시스템 오류에 따른 데이터 손실의 경우가 그것이다. IMDG는 동적 재구성과 확장의 용이성을 통해 이러한 이슈를 해결한다.

IMDG는 MMDB(Main Memory Database)와는 완전히 다르게 동작한다. MMDB의 대표적인 제품은 오라클의 TimesTen과 SAP의 HANA가 있는데, MMDB는 메모리 상에서 동작하는 완벽한 데이터베이스 관리 시스템이다. SQL을 통해 관리되며, MMDB 또한 데이터베이스 관리 시스템 특성상 무겁고 오버헤드를 동반한다. IMDG는 다르다. 테이블, 인덱스, 트리거, 저장된 프로시저나 프로세스 관리 등이 없다. 그저 단순한 저장소이다.

IMDG는 MMDB가 SQL을 사용하는 것과 달리 키(key)와 값(value)의 쌍을 데이터 모델로 사용한다. 키는 데이터베이스에서 인덱스나 주 키(primary key)처럼 값을 찾고 저장하는데 사용된다. 자바 언어는 이와 같은 키-값의 쌍으로 이루어진 데이터 구조를 잘 지원하고, 대부분의 IMDG 제품은 자바 언어로 개발되었다. 키-값 쌍의 데이터는 문자열, 숫자와 같은 단순한 타입에서부터 복잡한 객체에 이르기까지 자바 언어의 개발환경 안에서 어떤 형태로든 지원되는데 이것은 복잡함 자바 객체를 값으로 저장하기 때문에 이러한 객체를 관계형 데이터 모델로 변환할 필요가 없게 된다.

자바 언어는 메모리의 동적 할당을 위해서 주기억장치의 일부를 예약해 두고 사용한다. 이 예약된 메모리 공간을 힙(Heap)이라고 한다. 자바 애플리케이션을 통해 생성된 모든 실행 객체들은 힙에 저장된다. 힙의 크기가 증가되면 자바는 가비지 컬렉터(garbage collector)를 이용하여 유효한 객체와 유효하지 않은 객체를 체크하여 새롭게 가용한 공간을 모아 재배열하게 되는데 이는 결과적으로 성능 저하를 발생시킨다.

IMDG의 사용 제품은 이런 제약 극복하기 위해 오프-힙 메모리(Direct Buffer)를 사용한다. JVM(Java Virtual Machine)에 오프-힙 메모리 생성을 요청하면 JVM은 힙 바깥의 공간에서 메모리를 할당해 사용한다. 오프-힙 메모리 사용은 힙보다 느리지만, 가비지 컬렉션의 대상 영역이 아니므로 비교적 일정한 처리 시간 확보가 가능하다.

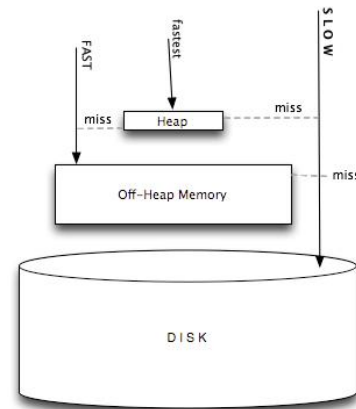


Fig. 7. Comparison Heap and Off-heap[15]



Fig. 8. BOM Java Tree using IMDG

PLM에서 대용량의 BOM 트랜잭션을 관리하는 Java 애플리케이션 구현에서 IMDG가 애플리케이션의 모든 데이터를 메모리에 넣어둠으로서 상대적으로 느린 데이터베이스에 데이터를 저장하는 것과 비교하여 빠르게 저장과 접근을 가능하게 해준다는 것을 확인하였다.

## VI. Conclusions

본 논문에서는 PLM에서 대용량 BOM(Bills Of Material) 데이터 활용 시에 데이터 로딩 속도 및 처리 속도를 향상시킬 수 있는 방안에 대해 연구했다. 해외 및 국내의 개발은 특정 산업 분야에 국한된 것이 많아서, 범용적인 BOM에서 대용량 데이터 처리에 대한 연구가 필요하다.

UI 표현에 있어서는 기존의 복잡한 테이블 기반의 레이아웃을 사용하여 데이터를 처리하는 방법 대신에 자바스크립트의 순수 웹 데이터 그리드 제품을 이용해 데이터를 처리하여 BOM

데이터 로딩 시간을 단축하였다. 특정 분야에서의 PLM 뿐만 아니라 다양한 분야의 PLM에서도 데이터 그리드를 활용한 BOM을 적용할 수 있다. 데이터의 저장소로 메인 메모리는 사용하는 IMDB(In-Memory Data Grid) 시스템을 적용하여 애플리케이션이 모든 데이터를 메모리에 넣어둠으로서 상대적으로 느린 데이터베이스에 데이터를 저장하는 것과 달리 빠르게 저장과 접근을 가능하게 하는 것을 확인하였다. 향후 제안된 표준 데이터베이스 구조에 이를 활용할 예정이다.

## REFERENCES

- [1] Saaksvuori, A. and Immonen, A., “*Product Lifecycle Management*”, Springer, 2004
- [2] Burden, R., “*PDM: Product Data Management*”, Resource Publishing, 2003
- [3] J.H. Lee, Y.G. Kim, D.K. Oh, J.G. Shin, “A Functional Review and Prototype for Ship PDM Implementation”, *Journal of the Society of Naval Architects of Korea*, Vol.42, No. 6, pp. 686~697, December 2005
- [4] Mather, H., “*Bill of Materials*” Dow Jones-Irwin, 1987
- [5] Whifield, R.I., Duffy, A.H.B., Meehan, J., Wu, Z., “Ship product modelling”, *Journal of Ship Production*, Vol.19, No. 4, pp. 230~245, November 2003
- [6] S.W. Jung, “Analysis of Internal PLM Requirement”, *Academic Journal of Computational Design and Engineering*, Vol.17, No. 3, pp. 61~63, December 2011
- [7] S.H. Kim, J.H. Lee, K.W. Kim, J.I. Jeon, “An Implementation Enterprise BOM for Marine Vessel of Pipe Equipment”, *Korean Journal of Computational Design and Engineering*, Vol.16, No. 1, pp.41~51, February 2011
- [8] D.Tony Liu, X.William Xu, “A review of web-based product data management systems”, *Computers in industry-Elsevier*, Vol.44, Issue 3, pp. 251~262, April 2001
- [9] Roy Prins, “*In-Memory Data Grid*”, *Article of Big Data Zone*, March 2013
- [10] Ann Chervenak, Ian Foster, Carl Kesselman, Charles Salisbury, Steven Tuecke, “The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets”, *Journal of Network and Computer Applications-Elsevier*, Vol.23, Issue 3, pp. 187~200, July 2000
- [11] Kim, G.Y, S.M. Kim, E.R. Hyun, “A Study on the evaluation of the UI-based usability of different sorts of responsive web interfaces”, *Journal of Korea Design Knowledge*, Vol.33, pp. 181~190, March 2015
- [12] J Andreeva, I Dzhunov, E Karavakis, L Kokoszkiwicz, M Nowotka, P Saiz and D Tuckett, “Designing and developing portable large-scale JavaScript web applications within the Experiment Dashboard framework”, *Journal of Physics: Conference Series*, Vol.396, Part 5, December 2012
- [13] DataTables, <http://datatables.net/>
- [14] Softboul, <http://www.softbowl.co.kr/>
- [15] B.H. Kim, S.Y. Jung, J.Y. Baek, S.J. Lee, “”Design of Collaborative BOM Management System for Small and Medium Enterprises“, *Korean Journal of Computational Design and Engineering*, Vol. 14, No. 4, pp. 254~260, August 2009
- [16] Terracotta, <https://confluence.terracotta.org/>
- [17] D.S. Kwon, “A Voronoi Diagram-Based Grid Structure for Efficient Nearest Neighbor Query Processing”, *Journal of the Korea Society of Computer and Information*, Vol.13, No.1, pp. 11~20, January 2008

## Authors



Jihyun Yoo received the B.S., M.S. degree and completed doctoral course in Computer Science and Engineering from Hanyang University, and received Ph.D. degree from Soongsil University. Dr. Yoo joined the faculty of the Department of

Internet Communication at Jangan University, Gyeonggi-do, Korea, in 2014. She is currently a Professor in the Department of Internet Communication, Jangan University. She is interested in mobile internet, cloud computing and content security.