

동시공학 구현을 위한 Web 기반의 공학 프로세스 지원 프레임워크

김 현*, 명재형*, 목경태*

The Web-Based Engineering Process Framework for Concurrent Engineering

Hyun Kim*, Jae Hyong Myong* and Kyung Tae Mok*

ABSTRACT

The engineering process including design, analysis/evaluation and manufacturing activities is becoming one of the key issues to embody a concurrent engineering concept. This paper proposes a framework to integrate the complicated engineering design and manufacturing processes under the concurrent engineering environment. The framework offers the following facilities: (1) to represent the complicated engineering process (2) to coordinate design activities and execute the process in a distributed environment (3) to support a communication among the related engineers. The engineering processes is depicted using process flow graphs that consist in tasks and the corresponding input and output data. The engineering activities in the defined processes can be executed in a distributed environment through process controller of the framework. Engineers can communicate to suggest their opinions and to exchange product information in the framework. We have conformed the CORBA standard to integrate various distributed engineering tasks and communicate among them, and used Java to support the platform independent environment on the Internet. Since the proposed framework can be a formal approach to integrate the engineering processes by providing formalism, parallelism, reusability, and flexibility, it can be effectively applied to embody the concurrent engineering concept in a distributed environment.

Key words : Concurrent engineering, Engineering process, Process flow graph, Collaborative design, Product data management

1. 서 론

제품개발 과정에서 공학 프로세스는 초기설계 단계에서 상세설계 및 가공계획 단계에 이르기까지 설계통합과 설계해석이 반복적으로 이루어지는 과정으로서, 서로 다른 많은 공학 활동이 유기적으로 상호 연계되어 진행된다. 최근 이러한 제품개발 프로세스에 관련된 여러 활동들을 통합함으로써 제품의 개발 기간을 단축하고 제품의 품질을 향상시키기 위해 동시공학(concurrent engineering)^[1,2]의 개념이 도입되고 있다. 동시공학의 개념을 구현하기 위한 연구는 동시공학 팀(cross functional team, CFT) 등에

의한 조직 관리적인 방법^[3], 고객의 요구를 설계에 반영하여 개발단계에서 품질 보증을 고려하기 위한 방법^[4,5], 설계에서 가공이나 조립을 미리 고려하여 재설계를 최소화 하기 위한 방법^[6,7], 협동작업의 지원에 의한 방법^[8,9] 등 여러 분야에서 다양한 방법이 제시되고 있다. 특히 최근에는 제품의 기획에서 폐기에 이르는 제품의 전체 수명 주기를 고려하여 분산된 설계 활동을 통합적으로 운용하고 관리하고자 전자적인 네트워크를 기반으로 한 분산 환경에서의 제품개발 지원 시스템에 관한 연구가 활발히 이루어지고 있으며 제품 데이터 관리(product data management, PDM) 시스템을 중심으로 구현되고 있다. 현재의 제품 데이터 관리 시스템은 복잡한 공학 프로세스에 대한 방법론적 고려보다는 관련 소프트웨어의

*한국전자통신연구원, 동시공학연구팀

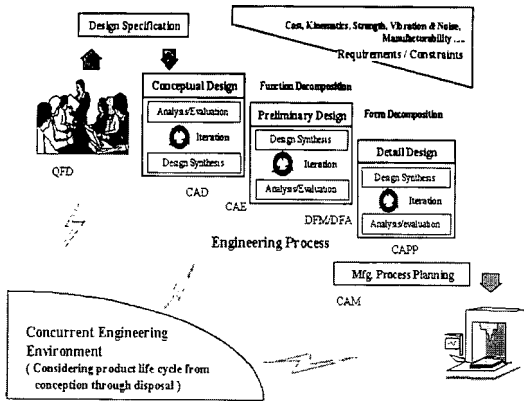


Fig. 1. Engineering process in concurrent engineering environment.

일관된 데이터 관리에 중점을 두고 통합된 제품 개발 업무를 제공하고자 하였기 때문에 설계의 업무 흐름을 지원한다는 관점에서는 유효하지만 복잡한 공학 프로세스의 특성을 고려한 설계 과정을 지원해야 한다는 관점에서는 기능적으로 미흡하다.

제품의 기획에서 폐기에 이르는 모든 과정을 통합하기 위한 동시공학 환경에서, 공학 프로세스는 개념설계로부터 생산계획에 이르는 복잡한 과정에 대한 워크플로우이다. Fig. 1은 동시공학의 환경에서 설계, 해석 및 평가, 생산계획에 이르는 공학 프로세스의 개념을 도식화한 그림이다.

즉 제품의 기획 단계에서 설계 사양이 도출되고 이로부터 개념 설계가 행해지며 여러가지 설계 요구조건을 만족시키기 위해 해석 및 평가의 반복적인 과정을 거치면서 설계가 진행되어 최종적으로 실제 가공 및 조립을 위한 생산계획이 행해지기까지의 일련의 과정을 공학 프로세스라고 정의한다.

본 논문의 목적은 이러한 공학 프로세스를 컴퓨터를 이용해 통합적으로 지원하기 위한 웹 기반의 프레임워크를 구축하는 것이다. 즉, 공학 프로세스 지원 프레임워크를 통해 복잡한 프로세스를 정형적으로 정의하고, 정의된 프로세스를 수행하면서 진행 상황을 모니터링하고, 관련작업 간의 협동설계를 지원함으로써 분산환경에서의 공학 프로세스를 컴퓨터 상에서 보다 효율적으로 수행할 수 있는 시스템을 개발하고자 한다.

2. 공학 프로세스의 표현 및 수행

2.1 공학 프로세스의 정형적 표현

공학 프로세스를 관리하고 제어하기 위해 가장 중

요한 문제는 복잡한 프로세스를 컴퓨터 상에서 어떻게 표현할 것인가 하는 것이다. 공학 프로세스를 표현하는 방법은 크게 다음 두가지의 관점에서 연구되어왔다. 첫째는 지식기반 접근방법으로써 비구조적인 공학 프로세스에 대한 지식을 규칙에 따라 표현하고 이를 추론하여 프로세스를 수행할 수 있도록 하는 방법이다^[10,11]. 이러한 지식기반 접근 방법은 분산 인공지능의 기법과 함께 많은 연구가 진행되어 왔으나 지식기반의 특성 상 특정 분야에서 문제의 해를 얻기 위한 방안으로는 유효하나 범용성이나 유연성이 부족하고 프로세스를 정형화하기가 어려운 문제점이 있다. 또 다른 접근 방법은 언어기반 또는 그래프 기반의 접근 방법으로써 프로세스를 언어, 네트워크 또는 그래프 등을 통해 정형적으로 표현하고자 하는 방법이다^[12,13]. 이러한 접근방법은 복잡한 프로세스의 수행 중 설계상태에 따라 발생하는 대안에 대한 고려가 부족하여 표현에 대한 제약이 있거나 표현된 프로세스의 수행을 제어하기가 어렵다는 문제를 갖고 있었다.

본 논문에서는 공학 프로세스를, 설계목적을 만족시키기 위해 수행되는 작업과 이로부터 산출되는 데이터의 관점에서 고려하고 이를 어떤 상위 작업으로부터의 입력을 받아 임의의 작업을 거쳐 출력으로 변환되는 연속된 과정으로 표현하였다^[14,15]. 즉 공학 프로세스를 식 (1)과 같은 프로세스 그래프를 이용하여 공학 프로세스를 정형적으로 표현하도록 하였다.

$$G = (T, S, E) \tag{1}$$

여기서 T는 공학 프로세스에서 수행되는 작업으로서 원으로 표기하며, S는 각 작업의 입력력으로서 사각형으로 표기한다. E는 작업과 입력력을 연결하는 선이다. Fig. 2는 이러한 프로세스 그래프의 개념을 도식화한 그림이다.

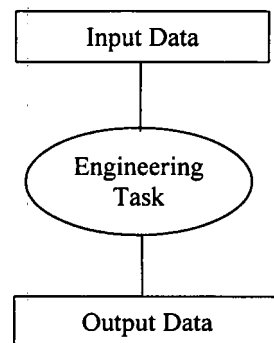


Fig. 2. Graphical representation of process graph.

복잡한 공학 프로세스를 정형적으로 정의하는 것은 설계대상에 대한 기능과 형태의 분해를 통해 프로세스의 작업들을 지정함으로써 이루어지며, 실제 프로세스의 수행 역시 이러한 과정을 통해 동적으로 진행된다. 즉 기능 또는 형태에 대한 작업을 분해할 때 상위의 작업이 보다 구체적인 작업으로 분해되고 최종적으로 컴퓨터 지원 시스템에 의해 수행될 수 있는 구체적인 작업이 정의될 때 까지 분해된다. 따라서 이러한 목적분해(goal decomposition)를 지원할 수 있도록 작업노드를 추상작업(abstract task)과 수행작업(logical task)으로 분류하였다. 수행작업이라는 것은 더 이상 분해될 수 없는 최종작업으로써 그래프 상에서 Terminal node로 표기된다. 수행작업은 실제 응용 프로그램 등이 호출되어 작업이 수행되는 구체적인 작업이다. 본 연구에서 수행작업은 다시 두가지로 분리하였는데 첫째는 제시된 프레임워크에 의해 자동으로 호출되고 수행될 수 있는 응용 프로그램이며 다른 하나는 설계자에 의해 호출되어 설계자의 작업이 요구되는 컴퓨터 지원 시스템이다. 이에 반해 추상작업이라는 것은 다시 하위의 구체화된 작업을 갖는 Non-terminal node이다. 즉 추상작업은 수행되어야 하는 방법론만을 표현하는 비구체적인 상태로써 또 다른 추상작업이나 수행작업들로 다시 분해된다. 입출력 노드는 입력사양과 작업을 수행한 후의 출력사양을 표시하며 마찬가지로 계층구조로 표현될 수 있도록 하였다.

본 개념 하에서 프로세스 그래프를 정의할 때, 초기 그래프에서 설계목표가 지정되고 어떤 입력 사양이 유효하고 어떤 출력 사양을 원하며 이를 위해 어떤 추상작업이 수행되어야 하는지가 지정된다. 이 그래프는 다시 분해되는데 이 때 상위 그래프는 분해될 하위 그래프로 대체된다. 이러한 대체가 최하위 수행작업에 이르기까지 이루어진다. 따라서 임의의 그래프를 상세 수준의 그래프로 변환하기 위한 수단이 필요하며, 이를 위해 본 논문에서는 식(2)와 같은 규칙을 적용한다.

$$P = (G_{org}, G_{new}, \sigma_{in}, \sigma_{out}) \quad (2)$$

여기서 G_{org} 는 대체될 상위 그래프이며 G_{new} 는 분해에 의해 대체할 하위 그래프이다. σ_{in} 은 G_{org} 의 입력으로부터 G_{new} 의 입력으로의 매핑으로써 두 그래프의 입력을 일치시키며 마찬가지로 σ_{out} 은 G_{org} 의 출력으로부터 G_{new} 의 출력으로의 매핑으로써 두 그래프의 출력을 일치시킨다.

실제 공학 프로세스를 정의하기 위해 작업을 분해

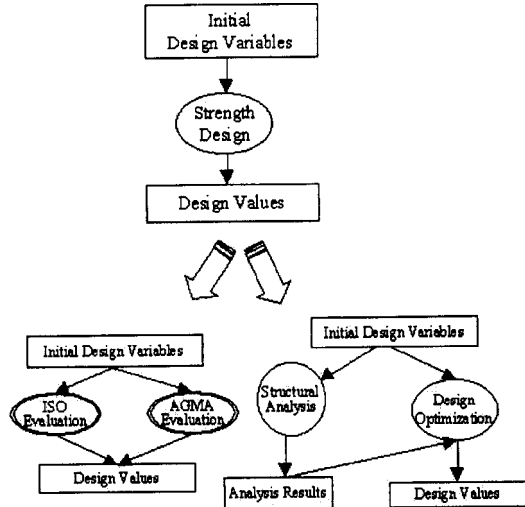


Fig. 3. Process decomposition with alternatives.

할 때 반드시 하나만의 선택을 갖고 체계적으로 분해되지는 않는다. 즉 작업을 분해하는 데는 상황에 따라서 여러 대안이 있을 수 있으며 이를 수행하기 위한 방법이나 컴퓨터 지원 시스템도 실제 프로세스가 진행되는 과정 중에 여러 선택이 있을 수 있다. 이를 지원할 수 있도록 프로세스가 진행되면서 상황에 따라 동적으로 그래프를 대체할 수 있도록 하였다. Fig. 3은 작업이 분해되는 과정의 한 예를 보여준다. 기어의 강도설계라는 추상작업은 ISO 또는 AGMA의 표준식에 의한 설계와 유한요소해석 및 최적설계에 의한 설계의 두가지의 대안으로 분해될 수 있음을 보여준다.

이상의 개념을 구현하기 위해 객체모델링 방법이 행해졌으며 Fig. 4는 공학 프로세스의 정의에 관련된 클래스들을 상위 수준에서 객체 모델링한 그림이다.

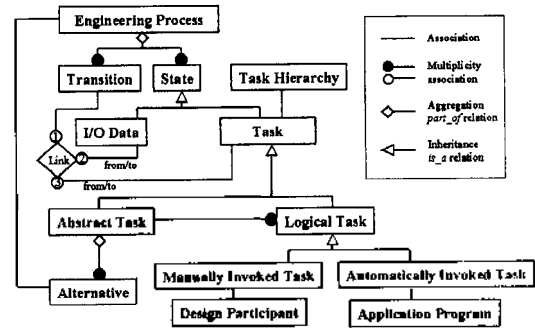


Fig. 4. High-level object model for engineering process.

2.2 공학 프로세스의 수행

공학 프로세스가 프로세스 그래프에 의해 정의되면 분산환경에서 작업이 수행되고 관리될 수 있어야 한다. 이를 위해 프레임워크는 프로세스 제어기를 가지며 이를 통해 다음과 같은 기능을 수행할 수 있도록 하였다.

- 초기 그래프를 로드하고 프로세스 객체를 생성한다.
- 프로세스를 추적한다.
- 추상작업의 경우 프로세스 규칙에 따라 그래프를 동적으로 분해한다.
- 최종작업의 경우 해당작업을 수행시킨다.
- 작업이 완료될 수 없을 경우 backtracking을 수행한다.

즉 설계 엔지니어가 정의된 프로세스를 처음으로 시작하고자 하거나 진행 중인 프로세스에 참여하고자 할 때, 프로세스 제어기는 프로세스 객체를 생성해 해당 설계자에게 넘겨준다. 이 때 프로세스에 참여하는 관련 엔지니어는 동일한 화면을 공유하면서 프로세스의 진행상태를 모니터링할 수 있다. 프로세스 제어기는 다음 프로세스를 결정하기 위해 메시지를 기다리며, 해당 메시지가 추상작업일 경우 대안평가를 통해 어떻게 작업이 확장될 것인지를 결정하고 이에 따라 보다 세부적인 프로세스를 만들어간다. 이러한 과정은 더 이상 분해될 수 없는 수행작업

에 도달될 때까지 반복되며 최종적인 수행작업의 경우 해당 작업을 수행시킨다. 수행작업이 완료되면 결과를 저장하고, 프로세스 그래프에 정의된 다음 작업으로 넘어간다.

이러한 과정 중에 작업의 결과가 만족되지 못했을 경우, 이전 작업으로 Backtracking이 요구된다. 프로세스 제어기가 이와 같은 작업 간의 상충성에 의한 Backtracking을 인지할 수 있도록 하기 위해 키 파라미터(key parameter)의 개념을 도입하였다. 프로세스의 수행 중에 핵심이 되는 설계변수 또는 설계 데이터는 해당 설계자가 키 파라미터로 등록하고 이 값이 변경되는 시점에서 프로세스 제어기가 상충성을 도출시킨다. 즉 이전 작업에서 설계변수가 결정되었는데 이후의 작업에서 그 값을 변경하고자 하는 경우나 두 작업이 동시에 수행되어 같은 설계변수에 대해 서로 다른 결과를 도출한 경우 상호 간의 상충성이 프로세스 제어기에 의해 인지된다. 이 때 Backtracking이 요구되거나 상호간의 결과에 대한 타협을 위해 협동설계 세션이 구성될 수 있다. Backtracking이 요구될 때 프로세스 제어기는 이전 작업에서 설계자에 의해 제시되는 다른 설계 변수를 적용하여 해당 작업을 수행하고 이 결과가 또다시 만족한 해를 얻지 못할 경우 다른 대안을 적용하는 과정을 반복한다. 프로세스 제어기는 설계 이력을 저장하고

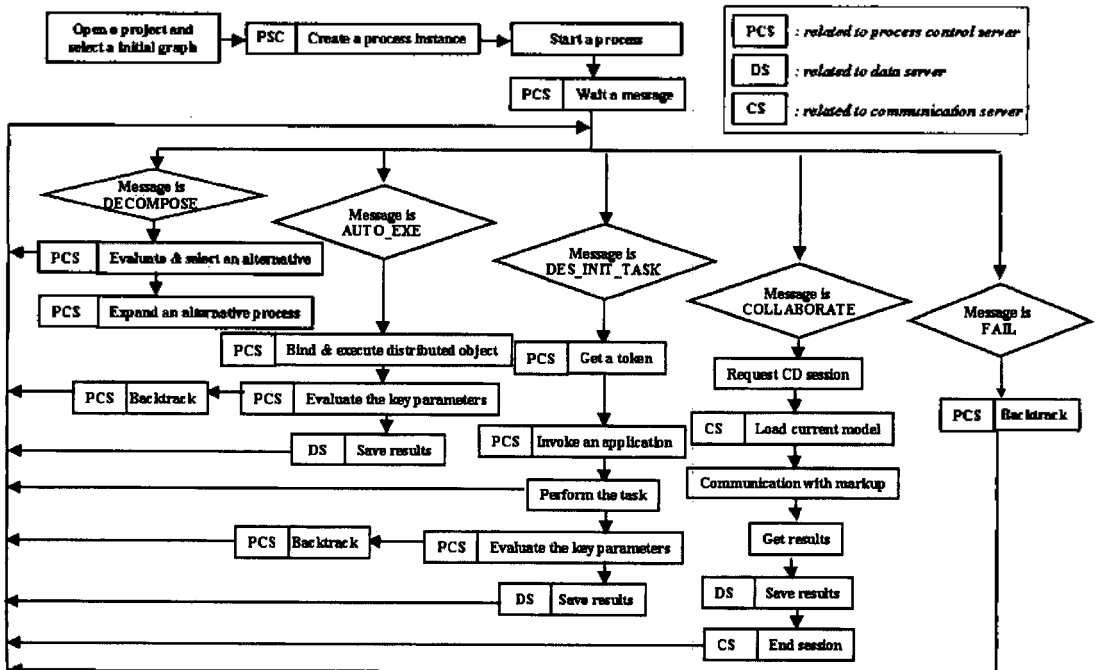


Fig. 5. Procedures for process control.

설계자가 Backtracking을 철회할 경우 이를 원래 상태로 되돌릴 수 있게 한다. 작업간의 상충성의 도출로 협동작업이 요구될 때 관련 설계자는 프레임워크를 통해 대상 모델을 공유하면서 커뮤니케이션을 수행할 수 있다. 그밖에 프로세스의 수행 상 고려해야 할 사항은 프로세스의 수행 중에 발생하는 정보에 대한 통합적 관리이며 이는 매우 중요하다. 형상정보를 포함한 기타 생산 및 제조 등 공학 프로세스 관련 정보를 통합하고자 하는 노력이 STEP¹⁶⁾을 통해 이루어지고 있다. 본 논문에서는 이에 대한 내용은 고려하지 않았으며 단지 프로세스의 수행 중에 발생하는 제품 또는 장치의 형상에 대한 계층적 구조를 관리할 수 있도록 하였다.

Fig. 5는 프로세스 수행을 위한 알고리즘을 보여준다.

3. 공학 프로세스 지원 프레임워크의 개발

3.1 프레임워크의 구조

앞서 언급한 내용에 따라 본 논문에서는 분산환경에서 복잡한 공학 프로세스를 지원하는 프레임워크를 개발하였다. 제시된 프레임워크의 개략적 구조는 Fig. 6과 같다.

프레임워크의 기반구조는 시간적 공간적으로 분산된 객체들 간의 통합, 조정을 위한 환경을 지원해야 하며 이를 위해 본 연구에서는 OMG(Object Manage-

ment Group)에 의해 제시된 CORBA 2.0 표준¹⁷⁾을 따랐다. 따라서 본 프레임워크 상의 모든 객체는 CORBA ORB(Object Request Broker)를 통해 커뮤니케이션이 행해진다.

제시된 프레임워크의 기본 개념은 공학 프로세스를 통해 실제 특정 문제를 해결할 수 있는 설계지원 객체(design support object) 또는 컴퓨터 지원 시스템들(computer-aided systems)이 미리 정의된 프로세스에 따라서 프로세스 제어기의 관리 하에서 문제를 해결해 가면서 프로세스를 진행하는 것이다.

Fig. 6에서 설계지원 객체는 프레임워크에 연결된 CORBA 구현객체로써 프레임워크에 의해 자동으로 수행되는 프로그램이며, 컴퓨터 지원 시스템들은 해당 설계자에 의해 호출되어 작업이 설계자에 의해 수행되는 프로그램이다. 이들은 프레임워크의 제어 하에서 특정 영역에서 자신의 문제를 해결하고 그 결과를 반영시킨다.

프레임워크는 실제 공학 프로세스를 진행시키는 하부구조라고 할 수 있으며 서버측과 클라이언트 측으로 나뉘어진다.

프레임워크의 서버는 프로세스 제어기의 역할을 하며 다음과 같이 구성되었다.

- 프로세스 제어 서버(process control server): 프로세스와 프로세스의 수행에 필요한 자원을 관리하고, 정의된 프로세스를 수행시키고, 관련 작업을 조정하는 서버.

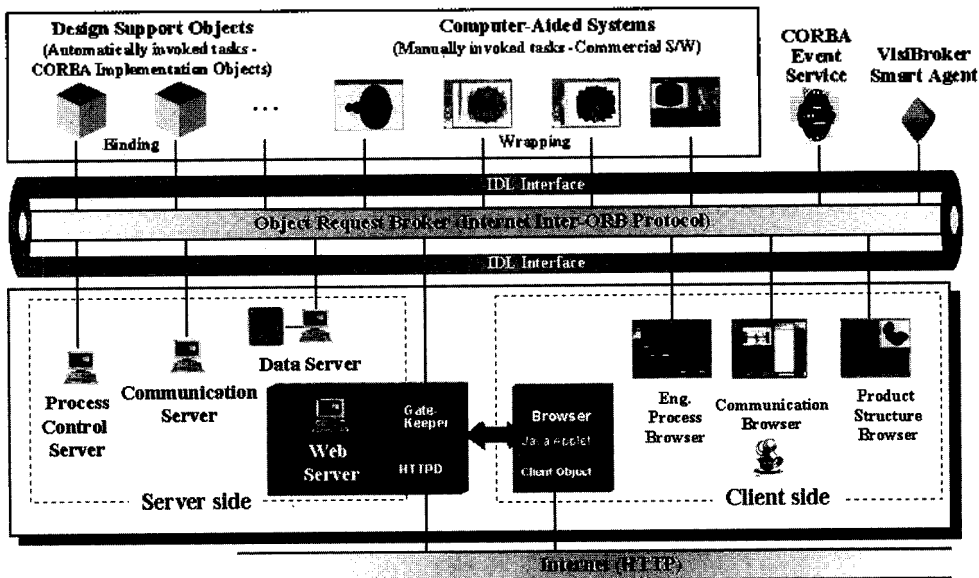


Fig. 6. Architecture of a framework.

- 데이터 서버(data server): 지역적으로 분산된 설계 엔지니어들이 파일서버에 접속해서 데이터를 이용할 수 있도록 지원하는 서버.

- 커뮤니케이션 서버(communication server): 관련 설계 엔지니어들이 서로 커뮤니케이션을 통해 협동 작업을 수행할 수 있도록 지원하는 서버.

프레임워크의 클라이언트 측은 설계 엔지니어가 Web 상에서 프로세스를 정의하고, 정의된 프로세스를 수행하고, 관련 엔지니어와 커뮤니케이션을 하기 위한 브라우저로서 프로세스 브라우저(engineering process browser), 커뮤니케이션 브라우저(communication browser) 및 제품구조 브라우저(product structure browser) 등으로 구성된다. 프로세스 브라우저는 사용자가 공학 프로세스를 컴퓨터 상에서 정형적으로 표현하고 프로세스의 수행을 제어하고 관리하는 역할을 한다. 커뮤니케이션 브라우저는 프로세스의 수행 중에 문제의 발생 시 또는 협동 작업이 요구될 시 엔지니어들 간의 의견 교환과 문제 해결을 지원하기 위한 대화용 브라우저이다. 제품구조 브라우저는 공학 프로세스의 대상인 제품 또는 장치의 구조를 관리한다.

본 논문에서 개발된 프레임워크는 시간적 공간적으로 분산된 작업들을 조정, 관리함으로써 엔지니어가 수행방법이나 컴퓨터 지원 시스템을 선정하고 이를 수행할 수 있도록 지원하며, 작업을 실행하기 위해 프로세스를 지속적으로 추적한다. 이를 위해 본 프레임워크를 통한 수행 환경은 시간적, 공간적으로 분산된 작업을 지원하고 플랫폼에 독립적이어야 한다. 이에 본 연구에서는 시스템이 인터넷 상에서 구동될 수 있도록 JAVA 언어를 이용하여 프로그램을 개발하였다.

3.2 프로세스 브라우저(Engineering process browser)

프로세스 브라우저는 새로운 프로세스를 정의하거나 기존에 정의된 프로세스를 변경하고, 또한 정의된 프로세스가 프로세스 제어기를 통해 수행될 때 이를 추적하여 그 결과를 보여준다. 프로세스 브라우저는 공학 프로세스를 컴퓨터 상에 표현하거나 편집하기 위한 주화면(process editor)과 작업을 분해하고 계층구조를 만들기 위한 작업화면(task browser) 및 프로세스에 대한 대안(alternative)을 고려해 분해하고 이를 표현하기 위한 분해화면(decomposition browser) 등으로 구성된다. Fig. 7은 프로세스 브라우저에 대한 화면을 보여준다.

프로세스를 정의해 가는 과정에서 추상작업의 경

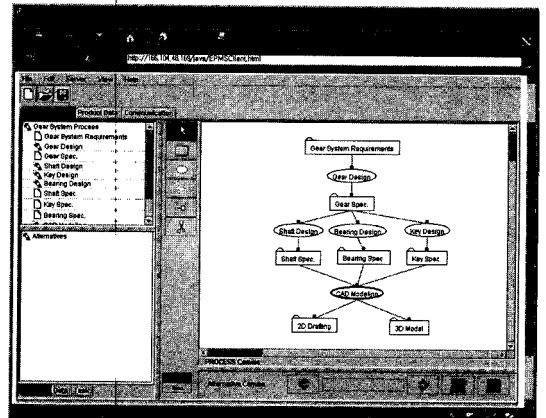


Fig. 7. Engineering process browser.

우 이를 분해하기 위한 대안들을 함께 정의할 수 있도록 하였고, 수행작업의 경우 실제 수행 프로그램의 위치 및 수행 명령어, 설계자 등을 속성으로 정의할 수 있도록 하였다. 또한 각 작업의 입출력 데이터에 대한 정보 역시 파일의 위치, 데이터의 유형 등을 속성으로 정의할 수 있도록 하였다. 이와 같은 정보와 함께 프로세스가 정의되고 각각의 프로세스 개체와 필요한 외부 수행 프로그램이 프레임워크에 연계되면, 프로세스 제어기는 프로세스 제어 서버를 통해 프로세스를 수행시킨다. 프로세스 제어기는 프로세스 그래프에 정의된 추상작업인 경우 보다 상세한 하위 프로세스로 분해하며, 수행작업의 경우 속성에 정의된 정보에 의해 수행 프로그램을 호출한다. 이러한 프로세스 상황은 프로세스 브라우저의 주화면에 표시되며 지금까지 어떤 작업이 수행되었고 앞으로 어떤 작업이 남아 있는지를 알려준다.

현재 프레임워크는 프로세스 수행을 위해 사용자와 대화식으로 한단계씩 진행하도록 하였다. 즉, 프레임워크는 사용자가 하나의 대안작업을 선택하거나 또는 사용자가 어떤 작업을 수행하기를 기다리고, 사용자는 프로세스의 진행 상황에 근거해 프로세스를 수동으로 진행시킬 수 있도록 하였다. 프로세스의 수행 중에 잘못이 발생하는 경우 backtracking함으로써 프로세스를 문제가 발생한 작업에서 다시 진행시킬 수 있도록 하였다. 또한 프로세스 편집기에서 정의된 프로세스는 파일로 저장되어 필요 시 재사용할 수 있도록 하였다.

3.3 커뮤니케이션 브라우저(Communication browser)

커뮤니케이션 브라우저는 프로세스 수행 중에 작

업간에 상충성이 도출되거나 설계자들 간의 의사교환이 요구될 경우, 커뮤니케이션 서버를 통해 이를 관련 설계자에게 게시함으로써 협동작업을 지원하기 위한 브라우저이다.

커뮤니케이션 서버는 다수의 클라이언트 애플릿을 통하여 작업을 하고 있는 사용자들이 대화를 할 수 있도록 해주는 통로이며 웹 서버가 구동되는 호스트에서 작동되도록 만든 자바 애플리케이션 프로그램이다. 한 설계자가 관련 설계자와의 의사교환이 요구될 때 커뮤니케이션 서버를 통해 커뮤니케이션 세션을 생성한다. 이 때 현재 진행되고 있는 설계 대상에 대한 3차원 형상모델이 세션 참여자들에게 공유될 수 있도록 하였다. 대상 모델은 STEP AP203 표준을 따르는 CAD파일을 Java 3D API를 이용하여 Web 상에서 가시화하였다.

커뮤니케이션 서버에서 대상 모델을 공유하고 작업자들 간의 커뮤니케이션을 지원하기 위해 CORBA 서비스 중의 하나인 이벤트 서비스를 이용하였다. Fig. 8은 커뮤니케이션 서버의 구조를 보여준다. 자신의 설계의도를 제시하는 설계자가 대상 모델과 커뮤니케이션 메시지를 이벤트 채널로 보내고 이벤트 채널은 이 내용을 관련 대상자들에게 전송함으로써 커뮤니케이션이 이루어진다. 이 때 특정한 엔지니어에게만 전송하겠다는 메시지와 함께 전송내용을 서버로 보내면, 서버는 메시지를 구분하여 특정한 엔지니어에게만 그 내용을 전달하도록 하였다.

커뮤니케이션 브라우저 내의 커뮤니케이션 패널은 공학 프로세스 중에 관련 엔지니어들 간의 대화창구로서의 역할을 한다. 이 패널은 작업의 대상이 되는 부품이나 장치에 대한 형상정보의 공유기능과 문자 또는 그림 정보의 전송을 통한 대화 기능을 위한 사용자 인터페이스를 제공한다. Fig. 9은 커뮤니케이션 브라우저의 화면을 보여준다.

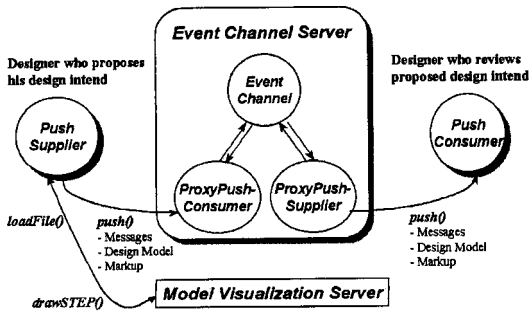


Fig. 8. Structure of a communication server.

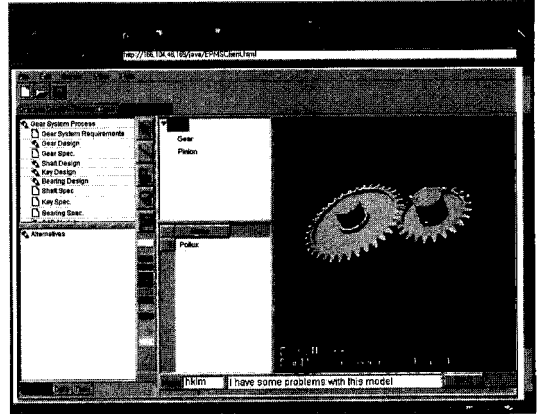


Fig. 9. Communication browser.

3.4 제품 구조 브라우저(Product structure browser)

제품구조 브라우저는 작업대상인 제품 또는 장치의 구성에 대한 계층구조와 이들 각각의 부품에 관련된 정보를 관리한다. Fig. 10은 제품구조 브라우저의 화면을 보여준다.

그림에서 좌측에 보여지는 것이 제품의 구조에 대한 계층구조 정보이다. 이는 제품의 기능(function)에 관련된 설계작업의 계층구조와 달리 형상(form)에 관련된 정보를 보여준다. 설계자가 공학 프로세스를 정의할 때 설계 대상인 제품의 구조를 정의할 수 있도록 하였으며 이 때 설계자는 프로세스 정의 시 작성된 작업의 입출력 데이터와 제품구조 내의 부품 또는 조립체를 연계하여 제품의 계층구조를 구성할 수 있도록 하였다.

그림 우측 상단에 보여지는 것이 선택된 조립체 또는 부품의 형상정보이며 우측 하단이 속성정보로써 이들 정보는 파일서버를 통해 관리된다. 형상정

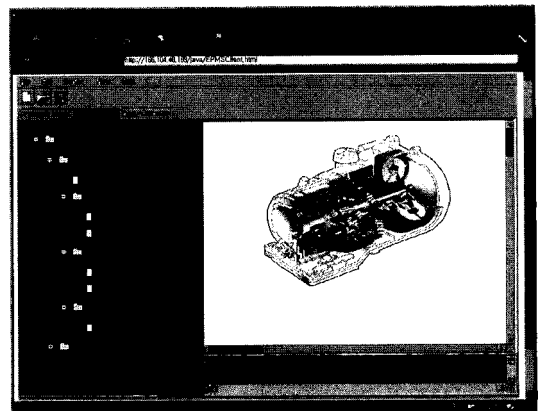


Fig. 10. Product structure browser.

보를 가시화하는 것은 STEP 파일을 이용할 수 있다.

4. 적용 및 검토

본 논문에서는 개발된 공학 프로세스 지원 프레임워크의 적합성을 검토하기 위하여 간단한 예제를 적용하여 프로세스를 분산환경에서 Web을 기반으로 수행하였다. 사용된 예제는 치차, 축, 베어링 및 키 등의 여러가지 기계요소가 복합되어 사용되는 치차 장치로써 이에 대한 최상위 설계 프로세스는 Fig. 7과 같다. 프로세스 그래프의 수행작업 노드에 해당되는 각각의 단위 프로그램들은 CORBA 구현객체로써 원격지의 컴퓨터에 위치시키고 이들을 개발된 프레임워크에 연결하였다.

설계사양은 일반 산업용 1단 외접 평치차 장치로 감속비는 4:1, 전달동력은 3마력, 입력속도는 300 rpm으로 고려하였다. 정의된 프로세스에서 초기설계의 입력제원은 중심거리 90 mm, 치폭 25 mm로 결정하고 전위는 고려하지 않은 설계로 결정하였다. 피니언과 기어의 재질은 데이터베이스의 검색을 통해 선정되고 이에 따라 재질의 특성이 결정된다. 다

음 프로세스에서 모듈 2 mm, 압력각 20°, 피니언 잇수 18개, 기어 잇수 72개를 초기값으로 하여 최적설계를 수행하였으며 그 결과로 피니언 잇수 20개, 치폭 30 mm로 설계값이 변경되었다. 다음에는 이 결과값을 가지고 강도평가를 수행하였다. 여기에는 치차의 파손에 직접적인 영향을 미치는 굽힘강도, 면압강도, 스코링강도 및 항복응력, 단위하중, 전달동력 등을 포함한다. 강도평가 과정에서 결과가 만족되지 않아 Backtracking이 발생되었으며 재설계가 진행되어 모듈 2.5 mm, 압력각 20°, 피니언 잇수 20개, 기어 잇수 80개, 치폭 30 mm의 표준치차로 설계를 완료하였다. Fig. 11은 이와 같은 진행 과정에 대한 화면 예를 보여준다. 수행이 끝난 작업은 녹색, 현재 수행 중인 작업은 청색, 수행 대기 중인 작업은 황색, 수행이 실패한 작업은 적색으로 표시된다.

치차 설계가 완료된 후 축과 키와 베어링을 선정하는 프로세스가 수행되어 피니언 측 축의 직경은 28 mm로, 기어 측 축의 직경은 40 mm로 결정되었다. 그리고, 피니언 측은 키가 체결되지 않았으며, 기어측은 10×8 mm의 키가 체결되었다. 베어링은 원통형 로울러 베어링(cylindrical roller bearing)이 선정

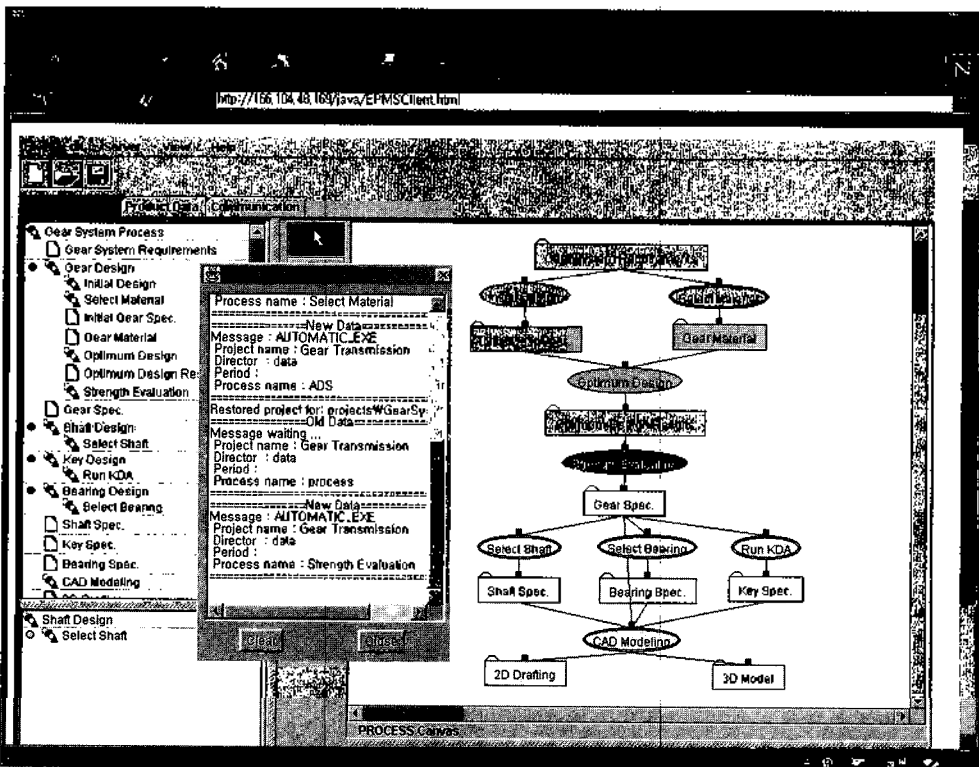


Fig. 11. Example display of the process execution for gear drive design.

Gear Design Results

- Gear Type : Spur
- Module : 2.5
- Pressure Angle : 20°
- Helix Angle : 0°
- Center Distance : 125.0 mm
- Normal Pitch : 7.380329 mm
- Circular Pitch : 7.853982 mm
- Backlash : 0.08 mm
- Clearance : 0.625 mm
- Tooth Depth : 5.625 mm
- Material of Pinion : SMC45C
- No. of Teeth of Pinion : 20
- Tooth Width of Pinion : 30 mm
- Addendum of Pinion : 1 Mn
- Pitch Dia. of Pinion : 50 mm
- Base Dia. of Pinion : 46.9846310 mm
- Tip Dia. of Pinion : 55 mm
- Root Dia. of Pinion 43.75 mm
- Addendum Modification Coef. of Pinion : 0
- Material of Gear : SMC45C
- No. of Teeth of Gear : 80
- Tooth Width of Gear : 30 mm
- Addendum of Gear : 1 Mn
- Pitch Dia. of Gear : 200 mm
- Base Dia. of Gear : 187.938524 mm
- Tip Dia. of Gear : 205 mm
- Root Dia. of Gear 193.75 mm
- Addendum Modification Coef. of Gear : 0

Shaft Dimensions

- Dia. of Shaft1 : 28 mm
- Dia. of Shaft2 : 40 mm
- L1 : 45 mm
- L2 : 45 mm
- L3 : 65 mm
- L4 : 45 mm

Key Dimensions

- h : 10 mm
- b : 8 mm
- t1 : 5 mm
- t2 : 3.3 mm

Bearing at Shaft 1

- Radial Load of Bearing A : 1494.962 N
- Radial Load of Bearing B : 1494.962 N
- Bearing Type : Cylindrical Roller Bearing
- Nominal No. of Bearing A : NU1005
- Inner Dia. of Bearing A : 25 mm
- Basic Dynamic Load Rating of Bearing A : 9370.999 N
- Nominal No. of Bearing B : NU1005
- Inner Dia. of Bearing B : 25 mm
- Basic Dynamic Load Rating of Bearing B : 9370.999 N

Bearing at Shaft 2

- Radial Load of Bearing C : 1766.773 N
- Radial Load of Bearing D : 1223.150 N
- Bearing Type : Cylindrical Roller Bearing
- Nominal No. of Bearing C : NU1007
- Inner Dia. of Bearing C : 35 mm
- Basic Dynamic Load Rating of Bearing C : 11074.81 N
- Nominal No. of Bearing D : NU1007
- Inner Dia. of Bearing D : 35 mm
- Basic Dynamic Load Rating of Bearing D : 7667.181 N

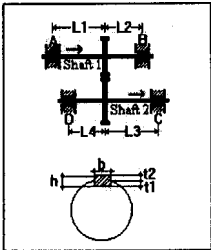


Fig. 12. Final results of gear drive design.

되었으며, 내경이 각각 25 mm, 35 mm로 결정되었다. Fig. 12는 최종 결과값들을 보여준다.

제시된 적용예는 프레임워크의 기본적인 기능에 대한 검증을 위한 간단한 예로써 결과를 얻기까지의 여러 프로세스들이 제시된 프레임워크 하에서 유기

적으로 상호 연계되어 진행됨을 확인할 수 있었다. 향후 보다 큰 문제에 적용될 경우 유사한 접근방법에 의해 통합된 공학 프로세스의 수행을 검증할 수 있으리라 예상되며 프로세스의 수행과 더불어 커뮤니케이션 브라우저와 제품 구조 브라우저를 함께 사용함으로써, 분산 환경하에서의 여러 설계활동이 보다 통합적이며 동시에 프로세스를 수행할 수 있을 것으로 기대한다.

5. 결 론

본 논문에서는 동시공학 환경에서 복잡한 공학 프로세스를 지원하기 위한 프레임워크를 제시하고 개발하였다. 제시된 프레임워크는 다음과 같은 기능을 수행한다.

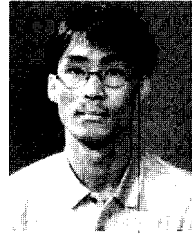
- 복잡한 공학 프로세스를 프로세스 흐름 그래프를 이용해 정형적으로 표현한다.
 - 공학 프로세스 내의 작업들을 초기의 추상적인 개념에서 실제 프로그램이 수행될 수 있는 구체적인 작업으로 분해할 수 있도록 한다.
 - 설계 수행 시 보다 넓은 설계공간을 탐색할 수 있도록 여러 설계 대안을 고려할 수 있도록 한다.
 - 정보 인프라를 이용해 분산환경에서 공학 프로세스가 동적으로 수행될 수 있도록 한다.
 - 공학 프로세스의 수행 중에 문제의 발생 시 또는 협동 작업이 요구될 시 설계자 간의 커뮤니케이션을 지원한다.
 - 공학 프로세스의 수행 중에 설계대상인 제품 또는 장치의 구조를 관리할 수 있도록 한다.
- 제시된 프레임워크를 통해 분산환경에서 공학 프로세스를 효율적으로 수행할 수 있으며 이에 따라 제품개발 기간을 크게 단축할 수 있을 것이다.

참고문헌

1. Institute of Defense Analysis, "The Role of Concurrent Engineering In Weapon System Aquisition", IDA Report R-338, 1988.
2. Chi-jac Oh, Hyun Kim, Sook-Chan Jeong, Haw-Gyoo Park, "A Concurrent Engineering System for New Product Development", *Proceedings of The 2nd International Conference & Exhibition, CALS Korea '95*, pp. 162-175, 1995.
3. Zelenak, S., Ridenour, M.A., Troxell, W.O., "A Team Approach to Concurrent Engineering: A Case Study", *Successful Implementation of Concurrent*

- Engineering Products and Processes*, Edited by S.G. Shina, Van Nostrand Reinhold, pp. 65-87, 1994.
4. Takezawa N. and Takahashi M., "Deployment of quality and reliability", *Standardization and Quality Control, Tokyo*, pp. 88-97, 1986.
 5. Akao, Yoji, *Quality Function Deployment: Integrating Customer Requirements into Product Design*, Norwalk, Conn., Productivity Press, 1990.
 6. John Corbett, Mike Dooner, John Meleka, Christopher Pym, *Design For Manufacture*, Addison-Wesley Publication Company, 1991.
 7. Boothroyd G. and Dewhurst P., *Product Design for Assembly Handbook*, Boothroyd Dewhurst Inc., Wakefield, RI, USA, 1987.
 8. Ishii, H., "TeamWorkStation: Toward A Seamless Shared Workspace", *Proceedings of Computer-Supported Cooperative Work-90*, pp. 12-26, 1990.
 9. Shu, L. and W. Flowers, "Teledesign: Groupware User Experiments in Three-Dimensional Computer-Aided Design", *Collaborative Computing*, pp. 1-14, 1994.
 10. Bond, A., "The Cooperation of Experts in Engineering Design", *Distributed Artificial Intelligence*, Vol. 2, Gasser, L. and Huhns, M.N. (eds), Pitman Publishing Ltd, London, UK, pp. 463-483, 1989.
 11. Lander, S.E., Staley, S.M. and Corkill, D.D., "Designing Integrated Engineering Environments: Blackboard-Based Integration of Design and Analysis Tools", *In Proceedings of IICAI-95 Workshop Intelligent Manufacturing Systems*, AAAI, 1995.
 12. Peter R.S. and Stephan W.D., "A Description Language for Design Process Management", *In Proceedings of 33rd Design Automation Conference*, Las Vegas, Nevada, pp. 521-529, 1996.
 13. Bosch, K.O., Bingley, P. and Van der Wolf, P., "Design Flow Management in the NELSIS CAD Framework", *In Proceedings of the 28th Design Automation Conference*, pp. 711-716, 1991.
 14. Reid A., Baldwin and Moon J. Chung, "A Formal Approach to Managing Design Processes", *IEEE Computer*, pp. 54-63, 1995.
 15. 박화규, 김 현, 오치재, 정문정, "분산환경 하의 설계 및 제조활동을 위한 프로세스 관리 기법", *한국 CALS/EC학회지*, 제 2권 제 1호, pp. 21-37, 1997.
 16. ISO Technical Committee 184, "Product Data Representation and Exchange-Part1: Overview and Fundamental Principles", *ISO Documentation*, ISO DIS 10303-1, 1992.
 17. Jon Siegel, *CORBA*, John Wiley & Sons Inc, 1996.

김 현



1984년 한양대학교 기계설계학과 학사
 1987년 한양대학교 기계설계학과 석사
 1997년 한양대학교 기계설계학과 박사
 1990년 - 현재 한국전자통신연구원 동시공학연구팀 선임연구원
 관심분야: Concurrent Engineering, Virtual Engineering, CAD/CAM/CAE/PDM

명제형



1993년 한양대학교 기계설계학과 학사
 1995년 한양대학교 기계설계학과 석사
 1995년 - 현재 한양대학교 기계설계학과 박사과정
 1997년 - 현재 한국전자통신연구원 위촉연구원
 관심분야: 동시공학, 가상공학

목경태



1996년 한양대학교 기계설계학과 학사
 1997년 - 현재 한양대학교 기계설계학과 석사과정
 1997년 - 현재 한국전자통신연구원 위촉연구원
 관심분야: 동시공학, 가상공학
