

개선된 성분조립모델을 활용한 웹사이트 개발 프로세스

천보경* · 박만곤 **

*부경대학교 대학원 전자계산학과

**부경대학교 전자컴퓨터정보통신공학부

초 록

웹사이트는 정보 미디어와 애플리케이션 플랫폼을 가지고 널리 사용되어지고 있다. 작은 기업들이 큰 기업들과 경쟁하기 하기 위해서 웹사이트 구축과 웹에 기반한 애플리케이션 개발이 절실히 요구되어지고 있다. 하지만, 대부분의 웹사이트 구축과 웹 애플리케이션의 생성은 체계적인 프로세스 없이 개발자들의 지식과 경험에 의존하여 임시적으로 개발되어 왔다. 따라서, 이러한 웹사이트 구축과 웹 애플리케이션 생성에 체계적인 프로세스를 가지는 소프트웨어 공학이 적용되어 개발, 유지되어야 한다. 본 논문에서는 최근에 ActiveX, Java, ASP, CORBA 등 다양한 형태의 코드와 플러그-인으로 개발되는 웹사이트와 웹 응용 애플리케이션에서 컴포넌트의 형태로 요소들을 구성, 접근하여 사용하고, 다른 웹사이트 구축이나 다른 웹 애플리케이션에서 이미 사용한 구성 성분을 재사용 함으로써 새로운 웹사이트 구축과 웹 애플리케이션 개발이 가능한 성분조립모델(the Component Assembly Model)을 웹사이트 개발 프로세스로서 활용하고자 한다.

I. 서 론

웹은 작은 크기와 작은 생명주기를 가지는 서비스들로부터 많은 분산된 서버들로부터 전체적인 작업흐름을 가지는 시스템들의 거대한 크기까지, 모든 종류의 애플리케이션들의 양도를 위한 대중적인 환경으로 진화해왔다. HTML 기반으로 개발된 애플리케이션들은 보편화되고, 다중 플랫폼으로 접근이 가능한 많은 웹 브라우저들의 탄생은 웹을 더욱 강력하게 만들었다.

특히, 인터넷과 WWW의 보급으로 인해 멀

티미디어 소프트웨어공학은 기업이 시간과 공간을 초월할 수 있도록 해주고 있으며, 작은 기업들이 대기업과 경쟁을 할 수 있도록 도와준다. 비록 회사의 규모가 작고 영업망이 없다고 할지라도 그들의 상품과 서비스를 웹사이트에 넣어두고, 세계전역에 판매 할 수 있다. 그러므로, 소규모의 기업들이 대기업과 경쟁하기 위해서는 대규모의 영업망과 영업사원을 대신할 수 있는 웹을 이용한 애플리케이션 개발과 사이트 구축이 절실히 요구되며, 이러한 개발 및 구축에 멀티미디어 소프트웨어 공학 기법이 적용되어야 한다[1,2].

성분조립모델(Component Assembly Model)은 소프트웨어 컴포넌트 라이브러리에서 재사용 가능한 컴포넌트를 추출하여 새로운 소프트웨어 시스템에 적용할 수 있는 소프트웨어 프로세스 모델로서 고객과의 의사소통과 고객평가를 중요시하고 있다. 이 모델은 중요한 핵심영역부터 주변영역으로 확장해 나가면서 소프트웨어를 개발하고, 소프트웨어 컴포넌트를 재사용 함으로써 개발기간을 단축하고, 비용을 절감할 수 있다.

본 논문에서는 기존의 성분조립모델을 개선하여, 개선된 모델을 활용한 웹사이트 개발 프로세스를 제안하고 있다. 2장에서는 관련연구로 웹사이트와 웹 애플리케이션에 대해서 알아보며, 소프트웨어 개발 프로세스로서 성분조립모델에 대해서 알아본다. 3장에서는 성분조립모델의 엔지니어링 활동을 개선시켜 웹사이트 개발 프로세스로 적용을 시켜 보았다. 4장에서는 본 논문에서 제안한 성분조립모델이 다른 소프트웨어 프로세스 모델에 비해 웹사이트 개발에 더 효율적인 개발 프로세스임을 비교, 고찰하였다. 5장은 결론으로 구성되어 있다.

II. 관련연구

II.1 웹사이트와 웹 애플리케이션

현재 웹에서는 많은 애플리케이션들과 서비스들이 만들어져서 제공되어지고 있으며, 빠르게 변화하고 있는 웹 기술들은 새로운 제품들과 애플리케이션을 지속적으로 확장시키고 있다. 그러한 기술의 파급은 초보자로부터 실질적인 전문가들까지 영향을 미치고 있으며, 교육, 엔터테인먼트, 의료, 종교, 판매를 비롯한 여러 분야에 상당한 영향을 주고 있다[2,3].

HTML을 기반으로 사용하는 애플리케이션들은 보편화되어가고, 다중 플랫폼으로 접근을 위한 많은 웹 브라우저들의 탄생은 웹을 더욱 강력하게 만들었다. 또 다른 웹이 가지는 장점은 소수의 고객들에게도 서비스가 가능하고 한 부분에 대한 집중적인 유지보수도 가능하며, 적은 비용으로 소프트웨어 업데이트를 수 시간만에 가능하다는 것이다. 이렇게 웹의 대중성과 클라이언트-서버 형태와 같은 웹의 장점들은 셀 수 없을 정도로 많은 HTML에 기반한 웹사이트와 웹 애플리케이션을 만들어 내고 있다.

초기의 웹은 단지 문서 페이지들에 대한 정보 미디어의 종류로서 특징을 가졌지만, 현재에 와서는 또 다른 효과로 전통적인 애플리케이션과는 다르게 웹사이트는 고정된 시작점을 가지지 않는다는 것이다. 클라이언트 측면의 관점에서, 웹 애플리케이션의 서버 부분은 다른 URL들에 의해 식별되는 많은 엔트리들을 통해서 접근되어질 수 있고, 일반적인 경우, 이러한 엔트리들을 서로 독립적으로 접근되어질 수 있다. 이것은 본래 단지 그들의 URL들을 사용하여 독립적으로 검색할 수 있는 서버 측에 있는 정적인 페이지들의 집합이었다. 완전한 하나가 되는 대신에, 웹 기반의 서버 부분은 일반적으로 정적 페이지 문서, CGI 프로그램, 스크립트 파일, 기타 등등의 서로 다른 형태의 많은 단위들로 고려되어지고, 실질적으로 오늘날 대부분의 웹사이트들은 그러한 것들의 혼합이다. 이것은 업그레이드를 하거나 다른 것에 대한 간섭 없이 애플리케이션의 일부 부분을 대체하는 것을 가능하게 만든다. 더구나, 이런 특징은 또한 애플리케이션 단위들로 재사용하는 잠재성을 가져온다[4,5].

웹 페이지들은 웹 애플리케이션들의 사용자 인터페이스들이다. HTML은 웹 페이지들이 멀티미디어로 특징 지어지는 폭넓은 표현들을

제공한다. 이후로, DHTML, Java applet, Java/VB Script, 기타 여러 가지가 그것들을 단지 보여주기만 하던 것에서, 애플리케이션 단위로 실행되어질 수 있도록 진화시켰다. 애플리케이션 부분에서, 그것들은 최종 사용자와 상호작용 하는 동안 이벤트에 대한 응답으로 이벤트를 운영하는 방식으로 작동한다. 서버부에서는 전형적으로 다르게 절차적 방법으로 운영된다. 그것은 서버 부분의 작업이 요구에 대한 응답으로 웹 페이지를 만들기 때문이다. 이것은 사용자 인터페이스가 비동기적인 사건들을 만들지 못하고, 요구가 주어졌을 때 그것의 실행 context가 고정되기 때문에 따라서 실행 코스는 순차적이다[4,5].

전통적인 소프트웨어는 생명주기에서 개발과 유지보수 단계들간에 명시적인 분류를 일반적으로 가진다. 달리 웹 애플리케이션은 첫 번째로 사용한 이후로부터 일반적으로 변화하고, 갱신하고, 업그레이드한다. 생명주기의 관점으로부터, 이것은 웹사이트 생명주기동안 평생 개발 혹은 유지보수의 단계를 가지며, 초기의 개발보다 더욱 값비싸고 강력하고 자원이 요구된다. 그래서, 고려할 만한 개발 접근은 웹사이트가 부드러운 방법으로 진화하는 것이 가능하게 하고, 편리하게 수행되어지기 위한 작업으로 개발되어져야 한다.

II.2 성분조립모델(the Component Assembly Model)

성분조립모델은 달팽이모델(Spiral Model)의 많은 특성을 수용하고 있는데, 소프트웨어 생성을 위한 반복적인 접근을 요구한다. 성분조립모델은 진화적인 모델로서 반복적인 진행에 따라 소프트웨어 시스템을 점점 더 완전한 버전으로 개발 가능하다.

그림 1은 성분조립모델을 보여주고 있는데, 고객과의 의사소통(customer communication) 단계에서는 개발자와 고객사이에 효과적인 의사소통을 수립하기 위해서 요구되는 작업들이 수행되고, 계획(planning)단계에서는 자원, 시간기준선 및 다른 프로젝트 관련 정보를 정의하기 위해서 요구되는 작업이 수행되고, 위험분석(risk analysis)단계에서는 기술적 위험과 관리적 위험 양쪽을 평가하기 위해서 요구되는 작업들이 수행된다. 그리고 엔지니어링(engineering)단계에서는 응용된 하나 이상의 산출물을 제작하기 위해서 요구되는 작업들이,

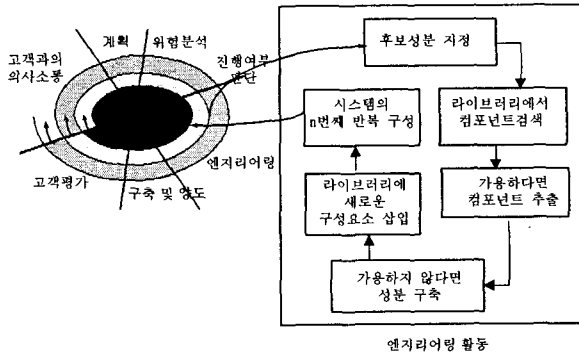


그림 1. 성분조립모델(Component Assembly Model)

구축과 양도(construction & release)단계에서는 사용자들을 위한 자원을 구축, 검사, 설치 및 제공을 하기 위해서 요구되는 작업들(예, 문서화와 훈련들)이 수행되고 고객평가(customer evaluation)단계에서는 엔지니어링 단계동안 생성되어지고 설치단계 동안에 구현되어진 소프트웨어 산출물들의 평가에 기반을 둔 고객의 피드백을 얻기 위해서 요구되는 작업들이 수행된다. 이러한 단계들의 반복에 의해서 소프트웨어는 개발되어진다[6].

III. 개선된 성분조립모델을 활용한 웹사이트 개발 프로세스에 대한 연구

웹사이트는 지금 정보를 위한 중요한 미디어이다. 몇 년에 걸쳐 개발되어진 후에, 웹사이트는 그것의 기능성과 내용에서 훨씬 더 복잡하게 되었다. 더욱이, 웹사이트는 그것의 내용과 기능성을 항상 업데이트하고 있다. 이것은 지속적으로 개발되고 유지보수에 대한 요구는 웹 개발자들에게 계속된다. 이것은 또한 소프트웨어 개발에서와 유사하게 거대한 웹사이트를 운영하는 동안 가장 큰 부담이다. 그러한 투자를 감소시킬 수 있는 모델들, 기술들과 도구들이 개발자의 작업을 도울 필요가 있다 [7].

웹사이트에는 많은 구성 성분들을 포함하고 있다. 텍스트와 그래픽스, 이미지, 그래픽스, 동영상, 사운드 등과 같은 멀티미디어의 정보를 표현하는 성분요소와 이를 제어하고, 표현하기 위해 사용되어지는 ASP, CGI, JAVA, HTML 등과 같은 개발언어의 성분요소가 합성되어서 하나의 웹사이트를 구성하게 된다. 웹사이트와 웹 애플리케이션은 최종 사용자와

상호작용적인 인터페이스를 하고, 웹사이트의 각종 컴포넌트들은 다른 웹사이트나 웹 애플리케이션에 다시 사용되어 질 수도 있다. 이와 같은 웹사이트의 특성을 고려해 볼 때, 성분조립모델(Component Assembly Model)은 개발자와 고객간의 원만한 의사소통을 유지할 수 있고, 고객의 평가와 함께 엔지니어링을 진행하면서, 엔지니어링 시 라이브러리에 있는 소프트웨어 컴포넌트를 재사용 할 수 있다는 점에서 웹사이트 개발 프로세스로 적용되어질 수 있다. 본 장에서는 향상된 엔지니어링 활동을 가진 성분조립모델을 제안하고, 계획정의 단계, 스토리보드 단계, 위험분석 단계, 웹사이트 엔지니어링 단계, 고객평가단계 등 5단계의 개발단계로 구성된 성분조립모델을 활용한 웹사이트 개발 프로세스를 보이고자 한다.

III.1 향상된 엔지니어링 활동을 가진 성분조립모델의 제안

기존의 성분조립모델에서는 엔지니어링 활동 동안 1단계만으로 소프트웨어 컴포넌트 라이브러리에서 웹사이트 개발에 가용한 성분만을 찾아서 추출하던 방법을 사용했지만, 본 논문에서 제안하고 있는 향상된 엔지니어링 활동에서는 이전의 개발업무단계에서 사용된 컴포넌트를 먼저 검색한 후, 소프트웨어 컴포넌트 라이브러리를 검색하는 다단계에 걸친 소프트웨어 컴포넌트 라이브러리를 검색, 추출하는 방법을 사용함으로써 기존 컴포넌트들의 재사용의 검색시간과 효율성을 높이고자 한다. 그림 2는 기존의 성분조립모델이 가지는 엔지니어링 활동이고, 그림 3은 본 논문이 제안하는 성분조립모델의 향상된 엔지니어링 활동을 적용한 것이다.

- | |
|--|
| <ol style="list-style-type: none"> 1. 후보 성분지정 2. 라이브러리에서 컴포넌트 검색, 가용 여부 판단 <p>적 합 : 컴포넌트 추출, 적용
부적합 : 새롭게 컴포넌트 구축, 라이브러리에 추가</p> |
|--|

그림 2. 기존 성분조립모델의 엔지니어링 활동

<1단계>

1. 후보성분 지정
2. 이전 개발업무단계에서 적용된 컴포넌트를 우선 검색, 가용여부 판단
합 : 컴포넌트 추출, 적용,
부적합 : 2단계로 확장

<2단계>

1. 기존 라이브러리 재검토
기존 라이브러리에서 후보성분과 유사성을 가진 이용 가능한 컴포넌트를 재검색
2. 후보성분과 유사한 기존 컴포넌트의 존재여부
 - 1) 존재한다면
 - 기존 컴포넌트를 수정, 확장
 - 앞 단계의 업무 프로그램에 적합한지 테스트 후 사용
 - 2) 존재하지 않는다면
새롭게 컴포넌트 구축, 라이브러리에 추가

그림 3. 향상된 성분조립모델 엔지니어링 활동

그림 3의 2단계에서 소프트웨어 컴포넌트를 재사용하기 위해 사용되는 컴포넌트 분류방식에는 컴포넌트들이 가지는 공통적인 측면의 특성을 합성하여 하나의 패킷(faceted)으로 표현하는 패킷분류방식(faceted classification scheme)을 사용하고, 라이브러리에 저장되는 컴포넌트들간의 유사도를 측정하여 서로 가까운 위치에 구조적으로 구축한다. 적용되는 검색모델로는 사용자의 요구사항을 정확하고 간단하게 표현할 수 있으며, 질의 작성과 구성이 용이하고, 구현된 시스템에서 짧은 검색시간을 제공하는 불리안 검색(Boolean Search)을 사용하면 된다[8].

III.2 개선된 성분조립모델을 활용한 웹사이트 개발 프로세스

본 논문이 제안한 향상된 엔지니어링 활동을 가진 성분조립모델을 웹사이트 개발 프로세스에 적용시켜 보자. 웹사이트 개발은 5단계의 개발단계에 의해서 이루어지며, 웹사이트 개발에서 핵심이 되는 중요한 업무영역부터 개발을 시작해서 개발업무의 영역을 점차 넓혀 가도록 한다. 그림 4에서는 성분조립모델을 활용한 웹사이트 개발 단계를 나타내고 있다.

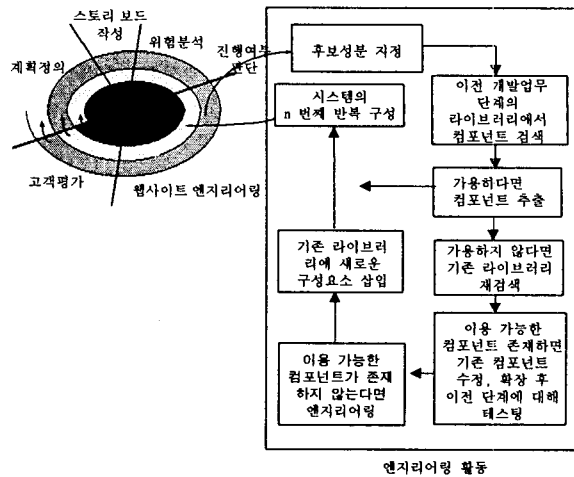


그림 4. 웹사이트 개발에 향상된 엔지니어링 활동을 가진 성분조립모델의 적용

웹사이트 개발을 위한 성분조립모델의 각 단계에서는 다음과 같은 작업을 수행한다.

(1) 계획정의 단계

계획정의 단계동안, 명확한 목표시장과 콘텐츠 목표와 개체가 설정되어야 한다. 웹사이트 개발 프로젝트는 매우 시간 소비가 크고, 중소기업체 주인이 주된 자본 투자가가 되므로, 대기업들이 무시해왔던 특별화 된 시장들에 대한 기회를 식별하는 것이 더 효율적이다. 또한 네티즌들에 대한 성향을 분석하고 즉각적인 전략과 계획을 수립해야 한다[1].

(2) 스토리보드 작성단계

구체적인 작업지시서와 디자인을 결정하는 웹사이트 설계의 단계이다. 웹사이트의 디스플레이 구성과 그래픽, 음악, 디스플레이 처리효과 등 더 나은 웹사이트 개발을 위한 최적의 방안을 모색함으로써 전체 구성의 통일성을 유지하고 개발 시간 및 예산의 낭비를 최소화할 수 있다.

(3) 위험분석 단계

웹사이트 개발하는데 있어 기술적인 측면이나, 관리적인 측면에서 위험요소가 있는지 평가하기 위해 요구되어지는 작업들을 수행한다.

(4) 웹사이트 엔지니어링 단계

실제 웹사이트 구축에서 필요한 프로그래밍은 물론, 필요한 이미지를 그리고, 애니메이션을 만든다. 이 단계에서는 그림 4에서 보는 바와 같이 웹사이트 개발에 필요한 컴포넌트를 라이브러리로부터 추출하여 재사용하는 엔지니어링 활동을 가진다. 여기에서, 재사용할 수 있는 컴포넌트는 Java의 각종 클래스들과 JavaBeans모델과 같은 소프트웨어 컴포넌트도 해당되며, 이미지나 동화상, 사운드 같은

멀티미디어 정보를 가진 파일이 될 수도 있다.
(5) 고객평가

개발되어진 웹사이트 업무부문에 대한 문서화와 훈련을 통해서 고객으로부터 평가를 받는다.

성분조립모델은 위와 같은 단계들을 반복적으로 수행함으로써 웹사이트를 개발할 수 있다.

IV. 성분조립모델을 활용한 웹사이트 개발 프로세스의 비교 및 고찰

IV.1 성분조립모델의 엔지니어링 활동 비교

성분조립모델의 기존 엔지니어링 활동은 1단계 검색방법으로 재사용의 검색속도와 효율성이 낮다. 그러나, 본 논문에서 제안한 향상된 엔지니어링 활동은 더 나은 검색속도와 효율성을 가져오게 한다.

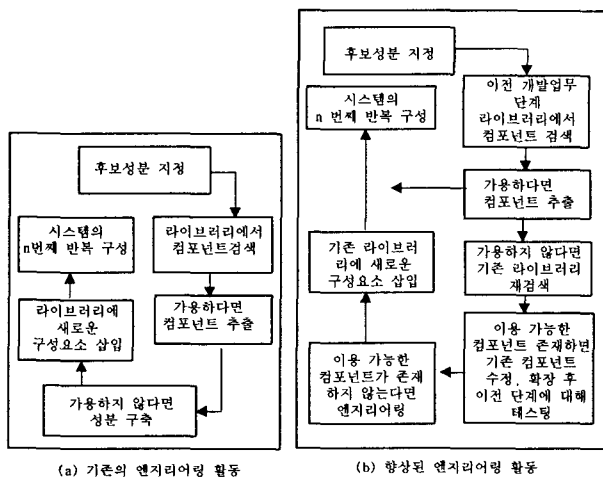


그림 5. 성분조립모델의 엔지니어링 활동

그림 5의 (a)는 기존의 엔지니어링 활동이다. 여기서는 재사용 컴포넌트 추출을 위해 1단계 검색을 적용한다. 하지만, 그림 5의 (b)와 같이 향상된 엔지니어링 활동에서는 이전 단계의 업무개발에서 적용된 컴포넌트들을 먼저 검색, 적용하고, 적합한 것이 없다면 컴포넌트 라이브러리로부터 의미적인 유사성을 파악하여 컴포넌트를 재사용 함으로써 컴포넌트 검색 속도와 재사용의 효율성을 개선시켰다.

IV.2 고찰

이제는 단순히 웹이 정보를 제공하는 문서들의 집합이 아니라, 여러 가지의 멀티미디어 객체를 수용하고 있는 사이트와 이 위에서 응용되는 애플리케이션들의 복합적 집합임으로, 객체 지향적 접근과 컴포넌트의 위주의 재사용이 가능한 성분조립모델이 웹사이트 개발을 위한 프로세스로 적용시켰다.

이렇게 컴포넌트 라이브러리에 있는 컴포넌트의 재사용성을 고려하는 성분조립모델은 객체 위주의 접근과 컴포넌트 위주의 사용이 많은 웹사이트에서 다음과 같은 많은 이점을 준다. 첫째, 웹사이트를 빠르게 개발할 수 있다. 둘째, 이전의 컴포넌트를 재사용함으로써 비용절감의 효과를 가져온다. 셋째, 기존의 성분을 한층 나은 성분으로 향상시킴으로 인해서, 소프트웨어 시스템 전체의 향상을 가져올 수 있다. 넷째, 유지보수에 있어 제한된 부분에 대한 집중화된 수정으로 다른 요소에 대해서는 독립적으로 수정이 가능하다.

웹사이트 개발 프로세스로서의 개선된 성분조립모델은 고객과의 의사소통을 열어두고 있고, 컴포넌트의 재사용이 가능하게 하며, 재사용에 있어 더욱 효율적인 엔지니어링이 가능하므로 다른 소프트웨어 프로세스 모델에 비해 효율적이라 할 수 있다.

V. 결론 및 향후 연구

웹사이트의 개발과 구축에는 많은 개발시간과 지속적인 유지보수비용이 소요되며, 많은 콘텐츠들과 웹 응용 애플리케이션은 객체지향적인 기술과 컴포넌트를 기반으로 한 개발을 요구한다. 이에 반해, 소프트웨어 개발 프로세스인 성분조립모델은 고객과의 의사소통을 열어두고, 소프트웨어 시스템 개발에서 중요한 업무영역부터 개발하여 점차 그 폭을 넓혀가면서, 컴포넌트를 위주로 한 재사용을 가능하게 한다. 그러므로, 웹사이트 개발 프로세스 모델로써 성분조립모델을 활용한다면, 개발업무의 피드백 줄이고, 컴포넌트 재사용을 통해 개발기간과 비용 및 유지보수의 시간과 비용도 절감할 수 있다. 더욱이, 본 논문에서 제안된 개선된 엔지니어링 활동을 가진 성분조립모델을 가지고 웹사이트 개발을 한다면, 재사용을 위한 검색속도와 성능의 향상을 가져올 것이다.

향후 연구에서는 본 논문에서 제안하였던 성분조립모델의 컴포넌트 재사용을 위한 컴포넌트웨어 하우스를 설계하고 여기에서 필요한 컴포넌트를 추출할 수 있는 자동화된 검색도

구를 개발하여 웹사이트 개발에 적용 시켜보고, 효율성을 평가 해 보아야 할 것이다.

참고문헌

- [1] Shi-Kuo Chang, "Perspectives in Multimedia Software Engineering", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.74-78, 1999.
- [2] Timothy Arndt, "The Evolving Role of Software Engineering in the Production of Multimedia Applications", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.79-84, 1999.
- [3] Chwan-Hwa Wu & J. David Irwin, "Emerging Multimedia Computer Communication Technologies", Prentice Hall PTR, 1998.
- [4] Hans-W. Gellersen and Martin Gaedke, "Object-Oriented Web Application Development", IEEE Internet Computing Vol. 3, No.1, pp.60-68, January, 1999.
- [5] Wei-quan Zhao, Jian Chen, "CoOWA: A Component Oriented Web Application Model", Proceedings of the Technology of Object-Oriented Languages and Systems - TOOLS 31, pp.191-199, February, 1999.
- [6] Roger S. Pressman, "Software Engineering", McGRAW-HILL International Editions, Fourth Edition, 1997.
- [7] Xin Fan, Jian Chen, "A Framework and Methodology for Development of Content-based Web sites", Proceedings of the Technology of Object-Oriented Languages and Systems - TOOLS 31, pp.316-319, February, 1999.
- [8] David C. Rine, Robert M. Sonnemann, "Investments in reusable software. A study of Software reuse investment success factors", Journal of Systems & Software, V.41 N.1, April 1998.
- [9] John G. Burch, "System Analysis, Design, and Implementation", boyd & fraser publishing company, 1992.
- [10] Stuart Kent, John Howes and Anthony Lauder, "Modelling Software Components", Proc. the 9th International Workshop on Database and Expert Systems Applications, pp.789-800, August 1998.
- [11] Masahito Hirakawa, "Do Software Engineers Like Multimedia?", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume 1 pp.85-90, 1999.
- [12] William I. Grosky, Ramesh Jain and Rajv Mehrotra, "The handbook of multimedia Information management", Prentice Hall PTR, 1997.
- [13] Boehm B, Egyed A, Kwan J, Port D, Madachy R, "User-Centred Design for Multimedia Applications", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.116-123, 1999.
- [14] Maria Francesca Costabile, "Usable Multimedia Applications", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.124-127, 1999.
- [15] Boehm B, Egyed A, Kwan J, Port D, Madach R, "Using the WinWinSpiral Model-A case Study", IEEE Computer, V.31 N.7, July 1998.
- [16] A.Bianchi, P.Bottoni, P.Mussio, "Issues in Design and Implementation of Multimedia Software Systems", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.91-96, 1999.
- [17] Jeffrey J. P. Tsai and Kuang Xu, "Architecture Specification of Multimedia Software Systems", Proc. IEEE International Conference on Multimedia Computing and Systems, Volume1 pp.97-102, 1999.