

# 노트와 스테레오타입을 이용한 UML 기반 웹 어플리케이션 설계 사례

엄태훈<sup>0</sup>, 최윤석, 정기원\*

송실대학교 대학원 컴퓨터학과, 송실대학교 컴퓨터학부\*

em\_sun@hanmail.net, yschoi@it.soongsil.ac.kr, chongf@computing.soongsil.ac.kr

## A Web Application Design based-on UML with Note and Stereotype

Taehoon Eom, Younseok Choi, Kiwon Chong

Department of Computing, Soongsil University

### 요약

본 논문에서는 사건흐름을 기능 별로 묶어 패턴화하는 방법을 통해 사건흐름 명세서를 표기하고, UML의 노트표기를 이용해 패턴 번호를 순차도에 적용하는 방법을 제시한다. 이와 더불어 설계 산출물 상에 표시되는 웹 페이지의 표현을 스테레오타입을 이용해 표기함으로써 시스템에 대한 이해도를 높이도록 하였고, 요구사항 명세서에 요구사항에 따른 사용사례와 사용자와의 관계를 표시하여 요구사항이 어떻게 반영되었는지를 쉽게 추적할 수 있게 하였다. 제안한 방법은 실제 영화 예매 시스템의 분석 및 설계 사례를 통해 그 사용 예를 보였다.

### 1. 서론

인터넷의 대두 이후 웹 어플리케이션은 소프트웨어 개발의 중심이 되고 있다. 그러나, 웹 어플리케이션은 개발 후 내용에 대한 잦은 갱신에 요구된다. 이와 더불어 멀티미디어 기능과 사용자 인터페이스 디자인, 웹 페이지내의 예외처리와 같은 세부적인 기능을 처리하는데 많은 노력을 필요로 한다. 웹 어플리케이션 개발은 크게 사용자 인터페이스 개발과 비즈니스 로직 개발로 구분 지을 수 있다. 사용자 인터페이스는 대개 HTML, 자바 스크립트, 비주얼 베이직 스크립트 등의 스크립트 언어로 작성하여 페이지별로 개발, 관리하고, 비즈니스 로직은 CGI, ASP, PHP, 자바 등으로 개발, 관리하는 추세이다. 사용자 인터페이스와 비즈니스 로직의 서로 다른 개발 특성으로 객체지향 설계와 더불어 사용자 인터페이스 부분에 대해 별도의 관리가 요구되고 있다. 웹 어플리케이션의 특성은 일반 객체 지향 어플리케이션과는 달리 사건흐름의 분기가 많이 발생하여, UML표기법으로 표현해야 하는 사건흐름의 양이 많고, 링크가 많이 생기므로 UML에서 지원하는 확장 메커니즘을 사용하여 확장된 표기를 사용해야 한다.

본 논문에서는 UML의 노트표기와 스테레오타입표기를 이용하여 웹 어플리케이션을 분석 및 설계하는 방법을 제안한다. 또한 실제 적용 사례를 선정하여 제시한 기법을 적용하고 이에 대한 산출물을 제시한다.

### 2. 관련된 연구

UML(Unified Modeling Language) : 분석 및 설계 모델을 표기하는 언어의 표준으로서, 데이터 모델링(Data Modeling concepts (Entity Relationship Diagram)), 업무 모델링(Business Modeling(work flow)), 객체 모델링(Object Modeling), 컴포넌트 모델링(Component Modeling)을 지원한다. UML은 구현 언어에 관계 없이, 전체 개발 생명 주기(Development life Cycle)에 걸쳐 모든 프로세스에 적용될 수 있다.[1]

UML은 설계를 위한 언어이므로 개발자가 객체지향 분석 및 설계를 어떻게 해야 하는지에 대한 지침이나 어떤 개발 프로세스를 따라야 하는지에 대한 지침을 제공하지는 않는다.[2]

UML에서는 표기의 확장이 필요할 경우 확장 메커니즘(stereotype, tagged value, constraints)을 사용할 수 있다. [3]

### 3. 설계 산출물 표기의 확장

3.1 요구사항 명세서 표기의 확장 : 개발 과정 중 요구사항의 변경이 발생했을 때, 해당 요구사항과 관련 있는 산출물의 연관성을 나타내는 표기가 없을 경우 변경을 산출물에 적용하기가 쉽다.

업데이트, 변경사항이 발생했을 때, 변경의 범위와 대상을 관리하기 용이하게 하기 위해 요구사항에 차례대로 번호를 부여하고, 사용사례도(Use-case Diagram)를 작성한 후 요구사항과 연관된 사용사례(Use-case)와 사용자(Actor)를 요구사항 명세서에 추가하였다. 또 요구사항에 우선순위를 부여함으로써 같은 사용사례로 포함되는 요구사항은 우선순위가 높은 쪽으로 흡수하였다. 위 방법을 이용하여 요구사항이 변경되거나 추가 되었을 때, 유사한 요구사항이 존재하면 우선순위가 높은 쪽으로 흡수하고 변경에 따른 사용 사례를 요구사항 명세서 표기의 확장을 이용하여 용이하게 변경할 수 있다. 또한 변경 범위와 요구사항의 흐름을 파악하기 쉽다는 장점이 있다.

3.2 사건흐름(Flow of Event) 명세서 표기의 확장 : 웹 응용 시스템의 예외상황이 많아 분기가 많이 발생하고, 페이지 사이의 상호참조가 빈번하여 중복되는 사건흐름이 많이 발생한다. 따라서 각 사건흐름에 대한 순차도(Sequence Diagram)의 양이 많아지고 중복이 발생한다. 또한 모든 사건을 사전에 검증하기 어렵기 때문에, 검증이 되지 않은 보안, 세션관리와 같은 기술적 위험요소의 제거가 쉽지 않다. 이를 해결하기 위해 사용사례에 따른 사건흐름에 번호를 "패키지명.사용사례번호.사건흐름번호"와 같이 부여하고, 세부 사건흐름은 "패키지명.사용사례번호.사건흐름번호.세부사건흐름번호"와 같이 번호를 부여하여, 중복되는 사건흐름은 같은 기능을 수행하는 사건흐름의 번호를 호출해 수행한다. 이와 더불어 사건흐름과 순차도간의 연관 관계를 도식화하여 표현하는 사건흐름 배치도를 적용사례 [그림 4-5]와 [그림 4-7]과 같이 표시한다. 사건흐름 배치도의 각 사건흐름에는 사건흐름 번호와 사건흐름의 기능을 적고, 사건흐름의 종류(주요흐름/하위흐름/대안흐름), 연관된 사용사례 명과 사건흐름이 기입된 순차도 명을 적는다. 이와 더불어 사건흐름에 따른 순차도의 노트표기에 같은 기능을 수행하는 사건흐름의 번호를 기재해서 사건흐름의 분기를 표시한다. 분기 표시 방법은 각 사건흐름의 시작 지점에 사건흐름 번호를 기입한 노트를 붙이고, 분기가 발생하는 지점에 선택어나 결과에 따라 다른 분기를 선택하도록 각각의 사건흐름 번호를 기입한 노트를 붙인다. 분기가 발생했을 때 노트 표기에 적힌 해당 사건흐름 번호로 이동하여 사건흐름을 수행한다. 위 방법을 이용하여 사건흐름 변경 및 삽입을 할 때에 관련 사건흐름의 하위 번호를 추가하고 변경 및 삽입을 할 수 있으며, 또한 순차도에 노트표기를 이용해 순차도의 변경 및 삽입을 용이하게 할 수 있다. 이러한 방법은 사용사례에 따른 사건흐름 변경 및 추가사항의 삽입을 용이하게 하며, 사건흐름과 순차도의 양을 줄일 수 있다.

3.3 UML 확장 메커니즘의 이용 : 웹 페이지에 들어가는 프레임

(Frame), 타겟(Target), 폼(Form) 등의 형식은 UML에서 제공하는 표기방법안으로는 정확한 표현이 어려우므로 확장 개념인 스테레오타입을 사용하여 표기 방법을 확장하였다. 스테레오타입을 이용하여 확장한 UML 표기 방법을 통해 웹 페이지에 대한 명확한 설계가 가능하고, 이에 따라 웹 페이지간의 연결과 호출을 명확하게 표기할 수 있다. 본 논문에서는 다음과 같은 스테레오타입을 이용하였다. <<Web Client Page>>는 클라이언트의 웹브라우저 상에 표현되는 페이지를 표현하며, <<Web Server Page>>는 서버 상의 비즈니스 로직을 표현한다. 이를 통해 현재 페이지가 클라이언트의 사용자 인터페이스인지, 서버의 비즈니스 로직인지를 표현할 수 있다. <<Web Client Page>>는 하위 스테레오타입으로 <<FrameSet>>, <<Target>>, <<Form>>으로 나누어져 사용자 인터페이스를 실제 웹 페이지 상의 형태로 표현한다.[4]

4. 적용사례

제시한 UML의 표기 확장을 통해 웹 어플리케이션의 한 예인 영화 예매 시스템 설계하였다. 설계 절차는 레서널 사의 RUP 단계에 준하였다. RUP의 단계는 도입(Inception), 정제(Elaboration), 구현(Construction), 전환(Transition)으로 되어있고, 각 단계에서의 핵심 워크플로우(Workflow)는 업무 모델링(Business Modeling), 요구사항(Requirements), 분석 설계(Analysis & Design), 구현(Implementation), 검사(Test), 배치(Deployment), 형상관리 및 변경관리(Configuration & Change Management), 프로젝트관리(Project Management), 환경(Environment)이고, 각 산출물은 단계별로 반복 수행한다.[5]

본 논문에서는 필요에 따라 적절히 워크플로우를 적용하여 산출물을 작성하였다. 본 영화 예매 시스템은 각 극장마다 존재하는 극장 시스템과 연계하여 인터넷 상에서 극장 상영 영화를 예매하는 시스템으로, 이용자는 인터넷을 통해 시스템에 접속하여 영화를 예매할 수 있다. 이용자는 영화 예매, 예매 마일리지 포인트 적용 확인, 영화 소개, 극장 소개, 극장별 상영 영화 소개, 극장 및 주변 지리정보, 계시판을 통한 영화 동호회 유지 등의 서비스를 이용할 수 있다.

4. 도입 단계

[그림 4-1]은 요구사항 명세서의 확장을 이용하여 작성한 요구사항 명세서의 일부이다. 요구사항 2.1.2와 2.1.4는 유사한 기능을 수행하는 요구사항이므로, 같은 사용사례에 통합하여 요구사항 우선순위가 높은 2.1.2에 흡수하였다.

| 요구사항 # | 요구사항             | 요구사항 분류 | 주요사항 / Actor | description  | Use-case 명            |
|--------|------------------|---------|--------------|--|-----------------------|
|        | 사용자를 인증한다        | 인증      | 공통 / User    | 사용자 인증 하위에서 사용자 아이디와 패스워드를 입력 받는다                                      | login                 |
| 1.1.1  | 영화들 목록 / 취소한다    | 가용성     | 공통 / Client  | 장르, 시간, 날짜, 단계를 입력 입력 관련 서비스 정보 검색하고 선택하여 해당되는 상영 극장, 상영시간, 매수 등을 보여준다 | purchase Advanced     |
| 1.1.2  | 영화를 검색한다         | 가용성     | 공통 / User    | 국립, 공공기관, 문화재단, 공공기관, 민간기업, 개인기업, 기타를 검색한다                             | search Movie          |
| 1.1.3  | 영화 열람 한다         | 가용성     | 공통 / User    | 관람할 영화의 상세 정보 열람   | consult Movie         |
| 1.1.4  | 자료를 지난 영화들 조회한다  | 가용성     | 공통 / User    | 1.1.2와 통합  |                       |
| 1.1.5  | 자료를 열람 하고자할 한다   | 가용성     | 보통 / User    | 영화 목록을 선택한 다음 열람 받는다   | review Movie          |
| 1.1.6  | 계시판을 이용한다        | 가용성     | 공통 / Client  | 계시판에 등록된 영화를 보인다   | use BillBoard         |
| 1.1.7  | 영화 열람이 완료된 후조회한다 | 가용성     | 보통 / User    | 관람한 영화의 열람이 완료된 후 조회한다   | update Movie Statistc |
| 1.1.8  | 관람한 자료의 열람 조회한다  | 가용성     | 공통 / User    | 관람한 자료의 열람 조회한다  | query Geograpy        |
| 1.1.9  | 영화 열람 정보 조회한다    | 가용성     | 보통 / User    | 영화 열람 정보의 열람 조회한다  | inquire Comics News   |

그림 4-1 요구사항 명세서

[그림 4-2]는 [그림 4-1]에 의해 도출된 사용사례를 사용자의 등급에 따라 패키지로 한 그림이다. 웹을 통한 일반 사용자는 commonServices 패키지의 서비스를 이용하고, 고객 등록을 한 고객은 permittedServices 패키지의 서비스를 이용하고, 관리자는 managerServices 패키지의 서비스를 이용한다.

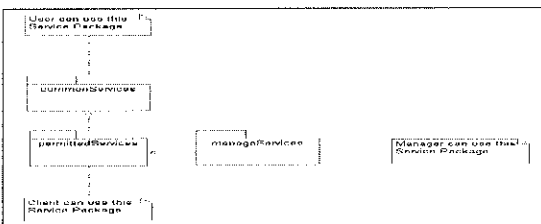


그림 4-2 사용사례 패키지

[그림 4-3]은 고객 인증후의 사용자가 이용하는 permittedServices 패키지를 표현하는 사용사례도이다. 고객이 예매를 수행하면 예매에 관한 정보가 극장 시스템으로 전송되어 예매 처리가 수행된 후 그 결과를 고객에게 보여준다.

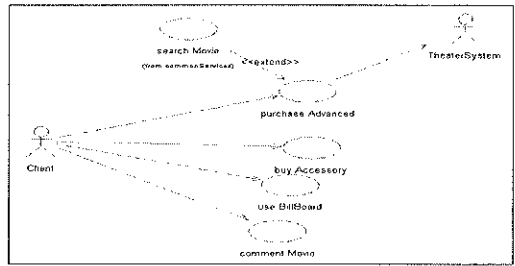


그림 4-3 permittedServices 패키지 사용사례도

[그림 4-4]는 사건흐름 명세서 중 영화 예매 사용사례에 대한 명세서이다. 영화 예매 사용사례는 사건의 분기가 많이 발생하므로 사건흐름 명세서 표기의 확장을 이용하여 permittedServices 패키지의 영화 예매 사용사례의 사건흐름으로 간소화한 예를 보여준다. 영화 검색 버튼을 사용하는 사건은 commonServices 패키지의 사용사례인 영화 검색 사건흐름번호 {1.7.1}을 수행한다. 입력된 영화가 존재하는지에 대한 검색을 하위 사건흐름번호 {2.3.1}이 수행하고, 입력한 항목에 대한 정검을 하위 사건흐름번호 {2.3.2}가 수행한다. 취소 버튼을 사용하는 permittedServices 패키지의 최상위 사건흐름은 고객 주화면 (Client Main) 사건흐름번호 {2.0}이 수행한다.

| 2.3 Primary event flow of purchase Advanced |   |
|---|---|
| Name  | purchase Advanced   |
| Brief Description                           | Subscription(영화등 예매한다)  |
| Primary Actor                               | Client  |
| Secondary Actor                             | Theater System  |
| Pre-Conditions                              | Client가 영화 예매 버튼을 누르면 시작된다  |
| Post-Conditions                             | Client가 예매하기 버튼을 누르면, 취소 버튼을 누르면 종료된다   |
| Flow of Events                              | Client는 영화 코드를 기입하고 다른 항목을 이동한다. 매수 시스템을 열람을 검색하여 해당되는 상영 극장, 상영시간, 매수 등을 보여준다. Client는 각 사항들 선택하고, 예매하기 버튼을 누른다  |
| Extension Points                            | Client가 영화 검색 버튼을 누르면, Primary event flow of search Movie (1.7.1)가 시작된다   |
| Exceptions                                  | 1. Client는 검색되지 않은 영화 코드를 기입하고 다른 칸으로 이동한다. 시스템은 Sub event flow of Check Movie (2.3.1)를 수행한다.<br>2. Client는 예매에 필요한 모든 데이터를 입력하지 않고 예매하기 버튼을 누른다. 시스템은 Sub event flow of Verify Subscription (2.3.2)를 수행한다.<br>3. Client는 영화 검색 버튼을 누른다. 시스템은 Primary event flow of View Movie (1.7.1)를 수행한다.<br>4. Client는 취소 버튼을 누른다. 시스템은 Primary event flow of Client Main (1.0)을 수행한다. |

그림 4-4 영화 예매 (purchase Advanced) 사건흐름 명세서

[그림 4-5]는 영화 예매 사용사례에 대한 사건흐름 배치도로서, 예매를 처리하는 사건흐름번호 {2.3}은 영화 예매 사용사례의 사건흐름으로 순차도의 명칭은 "seq\_pa" 라는 것을 보여준다. 사건흐름번호 {2.3.1}과 {2.3.2}는 영화 예매 사용사례의 하위(Sub) 사건흐름으로 "seq\_pa" 순차도에 같이 표기되어 있다는 것을 보여준다. commonServices 패키지의 영화 검색 사건흐름번호 {1.7}을 사건흐름번호 {2.3}에서 호출하는 관계로, <<link>> 스테레오타입을 이용하여 표기하였다.

사건흐름 명세서의 확장을 통해 각 순차도에 간소해서 동일한 사용사례의 사건흐름을 같은 순차도에 표현할 수 있다. 사건흐름 배치도의 사용으로 추가사항 및 변경이 용이하다.

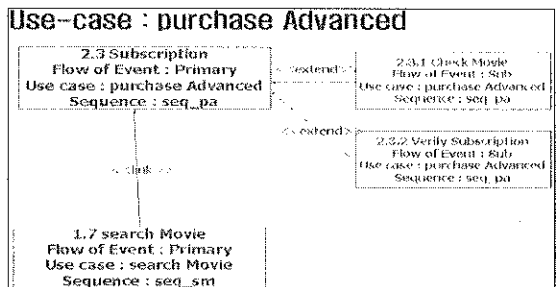


그림 4-5 영화 예매 (purchase Advanced) 사용사례 사건흐름 배치도

[그림 4-6]은 [그림 4-4]와 [그림 4-5]를 통해 작성된 사건흐름번호 {2.3}을 수행하는 순차도의 일부이다, 사건 흐름에 나타낸 분기는 순차도 상에서 노트표기를 사용하여 표현하였다.

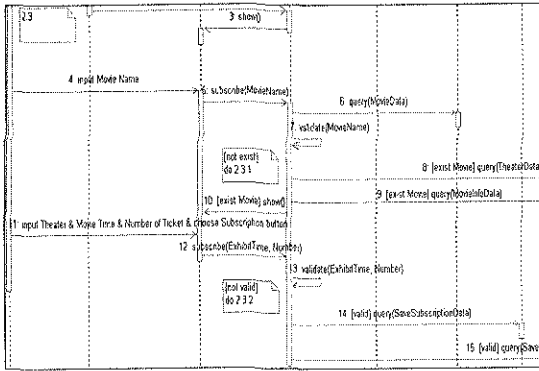


그림 4-6 seq\_sm 순차도

[그림 4-7]은 사건흐름번호 {2.3}의 예외사항 중 하나인 영화 검색 사용사례의 사건흐름 배치도이다, 영화 검색 사용사례는 commonServices 패키지 내의 사용사례로써, 사건흐름이 시작하면 사건흐름번호 {1.7.1}을 수행한다. 사건흐름번호 {1.7.1}에서 영화 영을 넣고 영화검색 버튼을 누르면 대안(Alternative) 사건흐름번호 {1.7.1.1}을 수행한다. 각 대안 사건흐름은 사건흐름번호 {1.7.1}의 순차도 “seq\_sm\_movie”에 같이 표기하였다. 극장별 검색 버튼을 누르면 사건흐름번호 {1.7.2}가 수행된다.

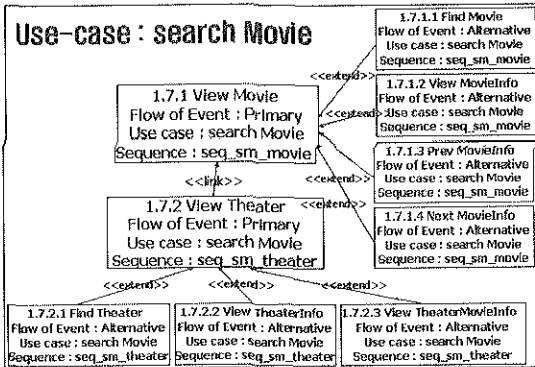


그림 4-7 영화 검색(Search Movie) 사용사례 사건흐름 배치도

[그림 4-8]은 [그림 4-7]의 사건흐름번호 {1.7.1}을 표현하는 순차도의 일부로서, 대안 사건흐름을 노트표기를 적용해서 순차도에 표기함으로 순차도의 양을 감소시켰다.

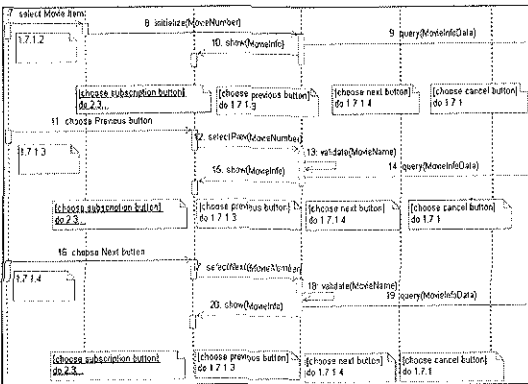


그림 4-8 seq\_sm\_movie 순차도

※ 정제 단계(Elaboration)

시스템 개발의 일관성을 위해 사용자 인터페이스를 구현할

clsBoundary 패키지, 요청 처리를 수행할 clsControl 패키지, 데이터 처리를 수행할 clsEntity 패키지로 나누었다.

clsBoundary 패키지와 clsControl 패키지는 사용자의 시스템 접근 권한 별로 패키지를 구분 지어, [그림 4-2]에 표시한 사용자 사례 패키지로 구분한 것과 일관성 있게 작성하였다. 이는 사용자 인터페이스를 담당하는 clsBoundary 패키지, 비즈니스 로직을 담당하는 clsControl 패키지, 데이터베이스와 관련된 clsEntity 패키지로 구분하는 것으로 각각 독립적인 개발을 가능하게 한다.

[그림 4-10]은 boundaryPermitted 패키지의 클래스도를 나타낸다. <<Web Client Page>> 클래스는 위해Permitted 패키지의 <<Web Server Page>> 클래스에 접근 하기 위해 controlPermitted 패키지의 ClientVerifier 인터페이스에 접근한다. 각 <<Web Client Page>> 클래스는 세부 클래스도를 가질 수 있으며, 사용자 인터페이스 설계자에 의해 <<Frameset>>, <<Target>>, <<Form>> 스타레오타입을 이용해 구체적으로 설계된다.

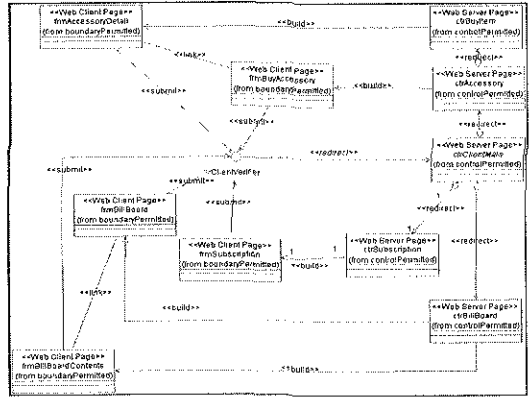


그림 4-10 고적 주화면(Client Main) 클래스도

5. 결론 및 향후 과제

웹 어플리케이션은 UML 표기 자체만으로 시스템을 설계하기 어려우므로 본 논문에서는 노트와 스타레오 타입으로 UML 표기를 확장하여 시스템을 설계하였다. 제안한 UML 표기의 확장 방법은 요구사항과 그에 해당하는 사용사례, 사건흐름, 순차도 및 이에 해당하는 클래스 패키지의 연관관계를 손쉽게 파악할 수 있도록 하며, 시스템 설계 시 순차도의 중복을 많은 부분 감소시킨다.

향후에는 제안한 표기방법의 정제가 필요하며, 실제 설계자료를 바탕으로 시스템을 구축하여 제안한 방법의 활용성을 확인해야 할 것이다. 이와 더불어 웹 어플리케이션의 사용자 인터페이스 설계 방법에 대한 연구가 필요하다.

참고문헌

[1] Booch, Rumbaugh, Jacobson, "The Unified Modeling Language User Guide", 1998, Addison Wesley  
 [2] Craig Larman, "Applying UML and Patterns", 1998, Prentice Hall  
 [3] OMG, "UML Specification v1.3", 1998  
 [4] Jim Conallen, "Modeling Web Application Design with UML", 1999, Addison Wesley  
 [5] Rational Co., "Rational Unified Process 2000", <http://www.rational.com/products/rup/whitepapers.jsp>