

XML 을 이용한 자바 빈즈 컴포넌트 명세의 표현

◦ 김영익*, 이성은**, 류성열***

*숭실대학교 컴퓨터학과, **동서서울대 사무자동화과, ***숭실대학교 컴퓨터학부

Representation of Java Beans component specification using XML

◦ Young Ick Kim*, Sung Eun Lee**, Sung Yul Rhew***

*Dept. of Computing, Soongsil Univ., **Dept of Office Automation, Dong Seoul College, ***School of Computing, SoongSil Univ.

요 약

오늘날 소프트웨어는 컴포넌트 기반의 소프트웨어라고 할 수 있다. 새로운 소프트웨어를 개발함에 있어서 컴포넌트를 사용하면 개발 기간과 비용 등 여러 면에서 장점이 있다. 그러나 이러한 컴포넌트 기반의 소프트웨어의 단점은 3자가 개발한 컴포넌트를 사용함에 따른 유지보수의 어려움이다. 제 3자가 개발한 컴포넌트의 경우 관련 문서, 설계 명세, 도움말 등이 존재하지 않는 경우 유지보수가 불가능하거나 아주 어렵게 된다. 이러한 경우 컴포넌트로부터 설계 명세 등의 정보를 추출할 수 있다면 기존 컴포넌트 기반의 소프트웨어를 기능을 향상시키거나 새로운 기능을 추가하는 등의 컴포넌트의 유지보수가 가능하다. 본 논문에서는 자바 빈즈 컴포넌트 모델을 기반으로 한 컴포넌트 명세 추출을 보인다. 자바 빈즈 컴포넌트로부터의 명세 추출은 자바의 리플렉션이나 자바 빈즈의 인트로스펙션 기능을 이용하여 소스 파일이 없이 자바 빈즈 컴포넌트의 클래스 파일만을 가지고 가능하다. 추출된 정보로부터 컴포넌트의 추상적인 설계 명세를 얻을 수 있다. XML은 웹 기반의 마크업 언어로써 구조적, 다중 링크, 자기 서술적, 정확성 등의 장점이 있다. 따라서 본 논문에서는 자바 빈즈 컴포넌트의 명세 표현 형태를 XML을 이용하여 자바 빈즈 컴포넌트 명세의 재사용성, 검색, 효율적인 관리가 가능함을 제시한다.

1. 서론

최근의 소프트웨어와 시스템을 개발에서의 중요한 기술적인 변화는 조립과 구성이 가능하며, 독립적으로 추출된 컴포넌트들의 조합에 의해 만들어진 시스템을 제공하는, 컴포넌트 기반의 소프트웨어 개발로의 전환이라고 할 수 있다. 컴포넌트 기반의 소프트웨어의 장점은 개발 기간의 단축과 개발 비용의 감소 등을 들 수 있다[3]. 이미 개발된 컴포넌트를 사용함으로써 얻는 개발 기간 단축은 컴포넌트나 모듈을 자체 개발을 하는 경우보다 외부로부터 구입하는 경우에 비용면에서 효율적이다. 그러나 소프트웨어는 항상 사용자의 요구에 부합하도록 유지보수 되어야 한다[2]. 컴포넌트 기반의 소프트웨어는 자체 개발 컴포넌트와 제 3자 개발 컴포넌트를 사용하게 된다[2]. 제 3자 개발 컴포넌트를 이용하여 소프트웨어를 개발하는 경우 컴포넌트 설계 명세, 소스 코드, 도움말 등의 문서가 존재하지 않거나 부족할 수 있다. 이런 경우 개발된 소프트웨어의 유지 보수가 어렵거나 불가능하게 된다[2]. 일반적인 컴포넌트는 소스 코드가 존재하지 않고 바이너리 코드만이 존재한다. 이런 상황에서는 어떻게 원하는 기능을 추가하고 향상시키기 위한 유지보수를 할 것인가를 고려해야 한다. 자바 빈즈 컴포넌트 모델을 예를 들면, 소스 코드 없이 클래스 파일만을 이용하여 컴포넌트

의 속성, 메소드, 이벤트 등의 정보를 얻을 수 있다[1,3,5]. 이 정보들을 이용하여 컴포넌트의 재사용을 위한 명세를 얻는다.

2. 관련 연구

2.1 자바 빈즈

자바 빈즈는 Sun에서 제시한 자바 언어를 위한 소프트웨어 컴포넌트 모델이다. 자바 빈즈는 자바 언어의 장점을 그대로 살려 이식성이 있고 플랫폼에 독립적이며, 재사용 가능한 소프트웨어 컴포넌트로서 개발 도구에서 시각적으로 조적이 가능하다.

자바 빈즈는 다음과 같은 특징을 갖는다.

○ 속성(Properties)

자바 빈은 소프트웨어 컴포넌트로서 자신만의 속성을 갖는다.

○ 이벤트(Events)

자바 빈에서 이벤트 모델이 컴포넌트들간의 통신을 위해서 사용된다. 이벤트를 발생하는 이벤트 소스와 이벤트를 받는 이벤트 리스너가 있다

○ 지속성(Persistence)

빈의 상태를 영구적으로 저장하는 것을 말한다.

○ 인트로스펙션(Introspection)

빈의 역할이 무엇인지를 알게 한다. 빈의 개발자가 빈을 설명하는 BeanInfo 클래스를 제공하거나 자바 리플렉션을 이용하는 두 가지 방법이 있다.

○ 커스터마이제이션(Customization)

일반적으로 자바 빈을 사용할 경우 직접 코드 작성을 하여 빈을 제어할 수도 있지만 어플리케이션 개발 도구에서 시각적으로 사용자의 목적에 맞게 설정할 수 있다.

2.2 XML(eXtensible Markup Language)

XML은 1996년 W3C에 의해 발표된 SGML의 부분 집합으로 기존의 웹 언어인 SGML과 HTML의 여러 단점을 보완한다[6]. XML은 자신의 마크업 요소들을 정의할 수 있기 때문에 확장성이 가능하고 유연성, 다중 링크 등의 특성을 갖는다[4]. 따라서 원하는 문서를 구조적으로 표현하는 것이 가능하다. XML 문서에서 사용할 태그를 정의하고, 이들이 어떤 순서로 동작하며, 어떤 태그가 다른 태그를 포함하는지 정의할 수 있다. 또한, 그 문서의 문서 타입 선언(Document Type Declaration)을 지정할 수 있다. DTD는 많은 XML 문서에서 반드시 필요한 것은 아니지만, XML이 규칙에 맞는지 확인하기 위해서는 DTD를 반드시 포함해야 한다. 또한 XSL은 XML 문서를 원하는 형태로 변형시켜 사용자에게 제공하는 역할을 한다. XSL이 없을 경우, 문서는 XML의 태그를 포함하여 보여주지만 XSL을 사용하면 HTML처럼 태그는 보이지 않고 원하는 데이터만 볼 수 있게 해준다.

3. 자바 빈즈 컴포넌트 명세의 표현 과정

3.1 자바 빈으로부터 명세 정보 추출

자바 빈즈 컴포넌트에서 명세 정보를 추출하는 것은 getBeanInfo() 메소드에 의해 가능하다. 이 메소드는 클래스의 BeanInfo 객체를 먼저 검색함으로써 BeanInfo 객체를 만든다. 예를 들어 Foo라는 자바 빈즈 클래스에는 FooBeanInfo라는 클래스를 만들 수 있다. 그리고 BeanInfo에 의해 제공되지 않는 정보를 위해 코어 리플렉션을 사용하여 클래스를 분석한다.

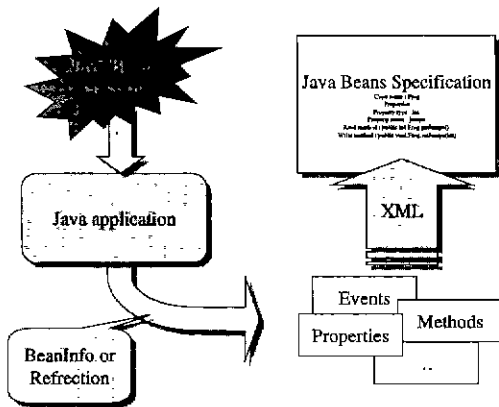


그림 1. 빈즈 컴포넌트로부터 명세 정보 획득

만약 제공되는 BeanInfo 클래스가 없다면, getBeanInfo() 메소드는 이벤트, 메소드, 속성 디스크립터를 생성하기 위해 모든 이벤트, 메소드, 속성을 분석한다. 그림 1은 클래스 파일로부터 빈의 명세 정보를 얻는 과정을 나타낸다.

이처럼 자바 빈즈 컴포넌트로부터 얻을 수 있는 정보로는 다음과 같은 것들이 있다.

표 1. 빈즈 컴포넌트로부터 얻는 정보

클래스 이름	자바 빈즈 클래스 이름
상위 클래스	자바 빈즈 클래스가 상속 받는 상위 클래스
인터페이스	자바 빈즈 클래스가 구현하는 인터페이스
생성자 메소드	클래스의 컨스트럭터
프로퍼티	속성
메소드	오버레이션
이벤트	어떠한 행위를 유발시키는 사건

본 논문에서는 테스트를 위해 이들 정보 중에서 클래스 이름, 속성, 메소드, 이벤트만을 사용하기로 한다.

3.2 DTD 정의

언어진 정보들을 이용하여 빈의 정형적인 명세를 XML로 표현하기 위해서는 DTD를 정의하여야 한다. DTD가 정의된 XML 파일은 논리적으로 유효하고 일관적이다. DTD는 XML 파일의 내부와 외부에 또는, 양쪽 모두 둘 수 있다. 본 논문에서는 한번 정의된 DTD가 여러 빈들의 명세에 똑같이 사용될 수 있으므로 외부 DTD를 정의해서 사용하기로 한다. 정의된 DTD는 다음 그림과 같다.

```
<ELEMENT document (bean)
<ELEMENT bean (className,properties,methods,events)
<ELEMENT className (#PCDATA)
<ELEMENT properties (property)*
<ELEMENT property (propertyType,propertyName,readMethod,writeMethod)
<ELEMENT propertyType (#PCDATA)
<ELEMENT propertyName (#PCDATA)
<ELEMENT readMethod (#PCDATA)
<ELEMENT writeMethod (#PCDATA)
<ELEMENT methods (method)*
<ELEMENT method (#PCDATA)
<ELEMENT events (event)*
<ELEMENT event (listenerType,listenerMethods,addListenerMethod,removeListenerMethod)
<ELEMENT listenerType (#PCDATA)
<ELEMENT listenerMethods (listenerMethod)*
<ELEMENT listenerMethod (#PCDATA)
<ELEMENT addListenerMethod (#PCDATA)
<ELEMENT removeListenerMethod (#PCDATA)
```

그림 2. 자바 빈즈 컴포넌트 명세의 DTD

3.3 XSL 생성

XSL은 XML 문서와 데이터의 포매팅 정보를 기술하기 위해 개발된 스타일 시트 언어(Style sheet language)이다. 문서의 내용과 포맷의 정보를 독립된 형태로 저장하도록 스타일 시트를 표준화하면 문서를 만들고 관리하는 모든 소프트웨어의 상호 호환성을 증진시킬 수 있다. 또한, XML 파일만을 브라우저로 보게 되면 데이터 뿐만 아니라 태그까지 보이게 된다. 따라서 사용자에게 데이터만을 적절한 형태로 제시하기 위해서는

XSL을 생성하여야 한다. 다음은 본 논문에서 사용될 XSL의 일부분이다.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-xsl">

  <xsl:template match="/">
    <html>
      <head>
        <title>JavaBeans XML Specification</title>
      </head>
      <body>
        <div style="text-align:left">
          </div>
          <xsl:apply-templates select="/bean/className"/>
          <xsl:apply-templates select="/bean/properties/property"/>
          <h4>Methods</h4>
          <xsl:apply-templates select="/bean/methods"/>
          <xsl:apply-templates select="/bean/events"/>
        </body>
      </html>
    </xsl:template>
  </xsl:stylesheet>
```

그림 3. XSL

3.4 XML 생성을 위한 자바 프로그램 작성

XML 파일은 자바 프로그램에서 생성된다. 빈즈 클래스 파일의 이름을 입력으로 XML 파일이 생성되며 빈즈 클래스로부터 얻어진 정보들이 DTD에서 정의된 형식으로 작성된다.

4. 테스트

지금까지 제시된 단계별로 생성된 빈즈 컴포넌트의 명세 정보, DTD, XSL, 그리고 자바 프로그램을 이용하여 테스트를 한다. 전체 아키텍처의 그림은 다음과 같다.

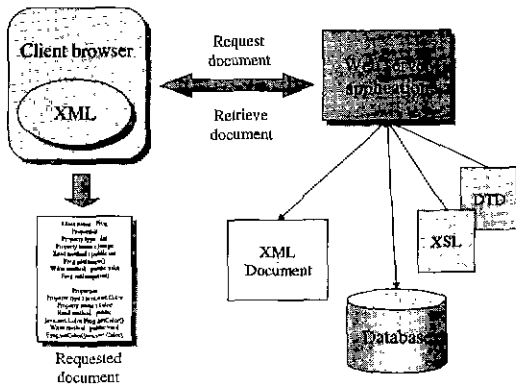


그림 4. 전체 아키텍처 구성도

우선 입력 예제로 Frog 라는 자바 빈즈 컴포넌트를 생성한다. 이 컴포넌트는 4개의 속성과 13개의 메소드, 2가지 형태의 이벤트를 포함한다. 컴파일하여 생성된 클래스 파일을 입력으로 위에서 작성한 자바 프로그램을 실행하면 XML 파일이 생성된다. XML 파일을 보기 위해 브라우저를 가동시키고 해당 XML 파일을 요청하면 XML, DTD, XSL 파일이 함께 전송되고 브라우저에서 처리되어 XML 문서를 XSL에서 정의된 형식으로 볼 수 있게 된다. 출력 결과는 다음 그림과 같다.

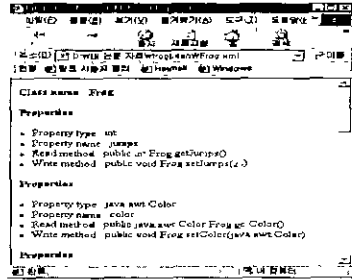


그림 5. 출력 결과

5. 결론

소프트웨어 개발 시에 컴포넌트를 이용한 개발은 개발 기간의 단축과 비용의 감소라는 효과를 거둘 수 있다. 그러나 제 3자 개발 컴포넌트의 경우 설계 명세를 나타내는 문서나 도움말, 소스 코드 등이 제공되지 않거나 부족한 경우, 유지 보수의 커다란 어려움을 겪게 된다. 따라서 사용되는 컴포넌트의 명세를 얻을 수 있다면 유지 보수에 커다란 도움이 될 수 있다. 자바 빈즈 컴포넌트의 경우 클래스 파일이나 클래스 파일을 묶은 jar 파일의 형태로 사용된다. 본 논문에서는 이러한 자바 빈즈 컴포넌트 클래스 파일로부터 명세를 추출하는 것을 보였다. 자바 빈즈 컴포넌트로부터 명세를 얻어내는 방법은 자바의 리플렉션이나 자바 빈즈의 BeanInfo를 이용하는 두 가지 방법이 있다. 명세를 표현하는 방법으로는 차세대 웹 언어로 떠오르는 XML을 이용하였다. XML을 이용하여 명세를 작성할 때의 장점은 다음과 같다.

- 자바 빈즈 컴포넌트의 명세를 표현하는 표준의 제공
 - 자기 서술적으로 명세를 표현 가능
 - 보다 명확하고 구조적이며 문서 자체에서 문서 구조의 검증이 가능
 - 유효하고 잘 구성된 문서 가능
 - 문서의 재사용이 용이
 - 웹 기반으로 사용자가 어느 곳에서나 문서에 접근 가능
- 작성된 자바 빈즈 컴포넌트의 명세와 XML의 장점인 검색 기능을 연동하면 명세 문서의 재사용성을 높일 수 있다. 앞으로 XML 명세를 효율적으로 관리하고 검색 가능한 레퍼지토리의 연구가 필요할 것이다.

6. 참고 문헌

- [1] Bruce Eckel, Thinking in Java, Prentice Hall, 1998
- [2] Jeffrey Voas, "Maintaining component-based systems", IEEE Software, July/August 1998
- [3] Reaz Hoque, Programming JavaBeans 1.1, McGraw-Hill, 1998
- [4] Richard Light, Presenting XML, Sams, 1998
- [5] Robert C. Seacord, "AGORA: A search engine for software components", IEEE Internet, November 1998
- [6] XML Specification, <http://www.w3c.org/TR/WD-xml.html>,