

도식화된 토픽맵 편집기

안기진[○], 이재호

서울시립대학교 전자전기컴퓨터공학부 인공지능연구소
3ky@naver.com[○], jaeho@uos.ac.kr

Graphical TopicMaps Editor(GTM Editor)

Kijin Ahn[○], Jaeho Lee

Artificial Intelligence Laboratory, Department Of Electrical and Computer
Engineering, University Of Seoul

요 약

국내에서도 표준화 작업이 진행 중인 토픽맵은 분산된 정보와 지식의 관리를 지원하기 위한 국제 표준이다. 이 논문에서는 이클립스(Eclipse) 플랫폼의 플러그인 개발환경인 GMF를 이용하여 비교적 작은 도메인에 대한 토픽맵의 구조를 도식화된 방법으로 표현하고, 간편하고 직관적인 편집기능을 사용자에게 제공하며, 이러한 구조의 간단한 프로토타입 정보라 할 수 있는 Linear Topic Map Notation(LTM)을 만들어 내는 플러그인을 소개할 것이다.

1. 서론

토픽맵(TopicMaps)은 분산된 정보(Knowledge)와 지식(Information)을 관리하기 위한 국제 표준(ISO/IEC 13250:2000)으로써 정보 자원의 구성, 추출, 네비게이션을 제공하는 새로운 패러다임이다. 이는 토픽(Topic), 연계(Association), 어커런스(Occurrence)의 TAO구조가 핵심인데, 주제를 중심으로 하여, 정보자원을 연계시켜 특정 도메인에 관한 맵을 구성한다. 한 주제는 다른 도메인의 주제와 연계될 수 있다. 관심 있는 정보를 찾을 때 있어서도 도식화된 맵이 만들어져 있다면 사용자에게 큰 도움이 될 것은 분명할 것이고, 정보간의 연계와 구조에 대한 부차적인 정보까지도 쉽게 이해시킬 수 있을 것이다. 또한 이러한 정보의 맵을 구성하는 과정에서도 도식화된 표기를 이용한다면, 정보 생산자에게 정보의 구성에 대한 전체적인 모습을 알아보기 쉽게 제공하여, 짜임새 있는 구조를 만들어내는 데 도움을 줄 것이다. 게다가 만약 사용하기 편한 편집기가 제공된다면, 맵을 만드는데 있어, 아이디어 도출에만 자신의 용력을 집중할 수 있을 것이다.

이클립스는 프로그래밍 도구로서의 역할로 정평이 나 있지만, 뛰어난 확장성과 안정성을 가진 플랫폼으로써의 역할도 훌륭히 해낸다. 이클립스 플랫폼은 플러그인의 모습으로 개발 가능한 수많은 확장점(extension point)을 제공한다. 이러한 개발과정을 *Contribution to Eclipse*라 하여 개발자의 욕구를 자극하고 있고, 이미

오래 전부터 이 플랫폼의 능력을 입증할 만한 여러 분야에서의 다양한 주제를 가진 플러그인들이 여럿 배포되어 있는 상태이다.

이클립스의 플러그인 프로젝트는 여러 분야로 세분화돼 있는데, 그 중 이 논문에서는 GTM Editor를 위한 개발환경으로 Graphical Modeling Framework(GMF) 프로젝트가 중심이 될 것이다. 간단히 그림을 그려보자면 토픽맵이라는 주제가 GMF의 모델이 될 것이다. GTM Editor를 통해, 토픽맵의 구조를 다이어그램이라 불리는 모습으로, 플랫폼상에서 표현할 것이며, 편집기능을 통해, 이 구조를 변경시키고, 실제 모델에 반영이 될 것이다.

2. 토픽맵의 개념

2-1. 토픽맵의 TAO 구조

◆ 토픽(Topics)

이야기의 주제(subject of discourse)이고, 여러 이름(name)을 가질 수 있고 타입(type)이 있다.

◆ 연계(Associations)

토픽들간의 관계(relationship)이다. 타입을 가진다

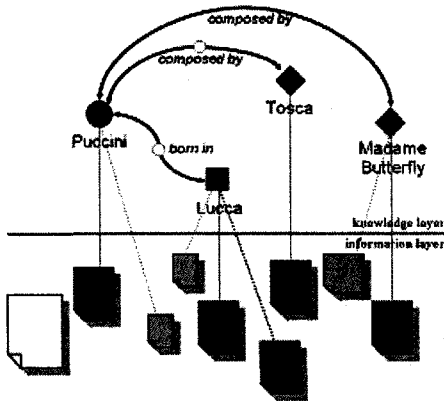
◆ 어커런스(Occurrences)

어떤 토픽에 관련된 실제 콘텐츠를 담고 있는 정보(information)이다.

책의 목차를 가지고 TAO 구조로 설명하면 좋은 예가 될 것이다. 백과사전에서 오페라라는 토픽을 목차에서 찾아본다면, 오페라의 정의나 역사 등의 관한 실제 내용을 담은 쪽수를 알게 될 것이며, 오페라와 연계되는 뮤지컬과 같은 다른 주제들이 백과사전의 목차에 의해 제공될 것이다. 즉, 찾고자 하는 책 속의 주제가 토픽이고, 그것의 책의 쪽수가 가리키는 책의 내용이 어커런스이다. 또한 이 주제와 관련된 다른 주제가 있다면 연계가 성립한다. 즉 책의 목차가 토픽맵이 되는 것이다.

2-2. 지식층(Knowledge layer), 정보층(Information layer)

토픽맵은 다시 그림 [2-1]에서와 같이 지식층과 정보층으로 분류할 수 있다. 정보층은 지식의 실제 자원으로 구성되는 부분이라 할 수 있겠다.



그림[2-1]. 토픽맵과 계층

2-3. Linear Topic Map Notation (LTM)

LTM은 온토피아(Ontopia.net)에서 정의한 토픽맵의 빠른 프로토타이핑을 위한 간단한 아스키(ASCII) 구문이다. 직감적으로도 토픽맵의 구조를 쉽게 파악할 수 있다. GTM Editor는 다이어그램에 그려진 토픽맵을 LTM 형식의 파일을 생성해 주는데, 다이어그램의 도식에 관한 정보가 아닌 모델에 관한 정보만을 담게 된다.

LTM 구문과 사용 예제

◆토픽

```
[topic-id : topic-type]
[tosca : opera]
[topic-id = basename; sortname?; dispname?]
[la-scala = "Teatro alla Scala"; "scala, teatro alla"; "La Scala"]
```

◆어커런스

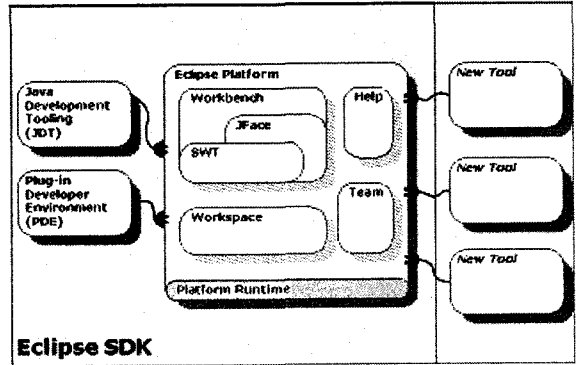
```
{topic-id, occurrence-type, [URL | data]}
{la-boheme,
homepage,"http://www.opera.it/.../La-Boheme.html"}
```

◆연계

```
assoc-type ( [role-player] : role-type, [role-player] : role-type )
composed-by( puccini : composer, tosca : work )
born-in ( puccini : person, lucca : place )
```

3. 이클립스 플랫폼과 Graphical Modeling Framework (GMF)

3-1. 이클립스 플랫폼



그림[3-1]. Eclipse SDK Platform Architecture

이클립스 플랫폼은 플러그 인이라는 개념으로 구조화돼 있으며, 확장 가능하다. 플러그 인은 시스템의 기능에 기여하는 특별한 코드나 데이터라고 할 수 있다. 이클립스에서는 확장점을 제공하고, 개발자는 이것을 확장한다는 개념이다.

그림 3-1 에서 이클립스 플랫폼의 내부 구조를 보여 주고 있다. 플랫폼 런타임(Platform Runtime)은 플러그 인의 시작(start-up)과 로딩(loading)을 지원한다. 워크벤치 이클립스(Workbench Eclipse) 유저 인터페이스(User Interface, UI)와 관련이 있는데, 세부적으로는 Editor, View, Perspective 로 구성돼있다. 이 외에도 내부적으로 Graphic 기능과 UI 작업 지원을 위한 SWT와 JFace가 있다.

3-2. Graphical Modeling Framework(GMF, 이클립스 플러그 인 개발환경)의 개발단계

이클립스내의 플러그 인 개발단계에서 자바(java)언어를 이용한 실제 프로그래밍은 피할 수 없는데, 각각의

확장점을 확장하고 연관 짓는 것과, 효율적이고, 유효한 구조를 유지하는 것은 상당히 어렵다. GMF 에서는 개발자의 이러한 어려움을 덜어주고자 프로그래밍과정을 구조화 단계에서 제시된 개발환경을 제공한다. 다음의 일련의 과정이 그것이고, 마지막 단계에서 다이어그램을 위한 실제 플러그인 코드를 만들어 준다.

①Domain Model 의 정의

다이어그램을 모델링할 도메인 모델의 정의가 필수인데, 여기에는 어떤 노드간의 정보, 타입, 관계의 정의가 포함된다. 이는 또한 Eclipse Modeling Framework(EMF)라 불리는 개발환경과 관련이 있다. 하지만 이 논문에서 자세한 언급은 하지 않고, 뒤에서 나오는 토픽맵에 대한 도메인 모델을 다루는 정도만 하겠다.

②Graphical, tooling 의 정의

도메인 모델에 정의된 노드나 그들간의 관계를 실제 다이어그램에 보이는 모습과, 각각을 그려줄 tooling의 정의 과정이다.

③Mapping 정의

도메인 모델과 Graphical, tooling의 정의 세가지를 맵핑시키는 과정인데, 모델 구조와 그에 따른 실제 다이어그램의 구조적 모습이 여기서 결정된다.

④generator Model

맵핑된 파일을 가지고, 다이어그램 플러그인 코드를 만들어줄 generator Model 파일을 만든다.

⑤Diagram 코드 생성

Generator Model을 통해 다이어그램의 실제 플러그인 자바 코드와 메타 정보를 위한 파일 등이 생성된다. 이렇게 최종 생성된 에디터는 다이어그램이라 불린다.

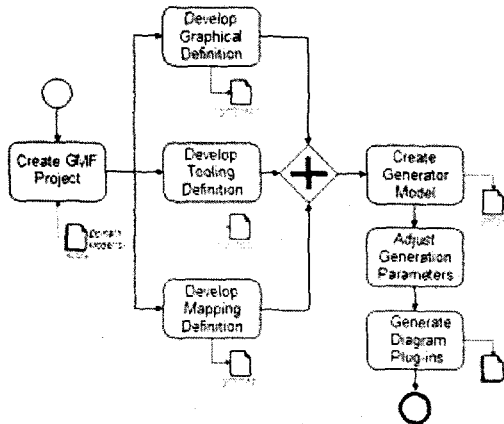


그림 [3-2]. GMF의 개발단계

GMF를 써서 토픽맵을 도메인 모델로 한 다이어그램을 그려주기 위해서는 토픽맵이라는 도메인에 관한 모델링이 필요하다. 이 방법으로 EMF에서 세가지 방법을 제시하고 있는데, XML Schema나 Java Interface, 또는 UML 표기법이 그것이다. 여기서는 UML을 사용하여, TAO 등을 클래스로 정의하고, TAO관의 연관성을 association관계, aggregation관계 등으로 표현하였다. 그림[4-1]에서 전체적인 도메인 모델의 구조를 보여주고 있다. 간단히 살펴보자면, 토픽은 어커런스를 여러 개 가질 수 있다. (aggregation 관계) 연관과 토픽과 관련하여 역할(Role)이 있는데, 연관은 기본적으로 역할간의 관계이다. 역할과 토픽과의 관계는 특정 토픽이 역할의 플레이어(player)가 된다. 연가는 이러한 역할들을 다수를 갖는데, 이는 어떤 오페라라는 주제와 관련하여 등장인물들과 제작자, 장소, 시간 등의 역할이 다수 등장할 수 있다는 예를 들 수 있겠다. TAO 와 역할 Class는 기본적으로 타입(type)을 갖는다. 각 클래스가 포함하는 성분들은 기본적으로 횡수 제한이나, 필요조건 등의 특성을 갖는다. 이 와 같은 모델링이 유일하다 할 수 없으며, 실제로 이 모델 이전에도 수많은 제안이 있었다. 다만 모델링을 할 때, 이 모델링이 토픽맵의 실제 개념을 포괄해야 하는 것은 물론 또한 기본적으로 GMF에서 제공하는 그리기 방법이 이 모델링에 제한을 받는다. 예를 들어서 GMF에서 제공하는 compartment라는 표현방식은 aggregation관계의 클래스들을 만을 위한 것이다.

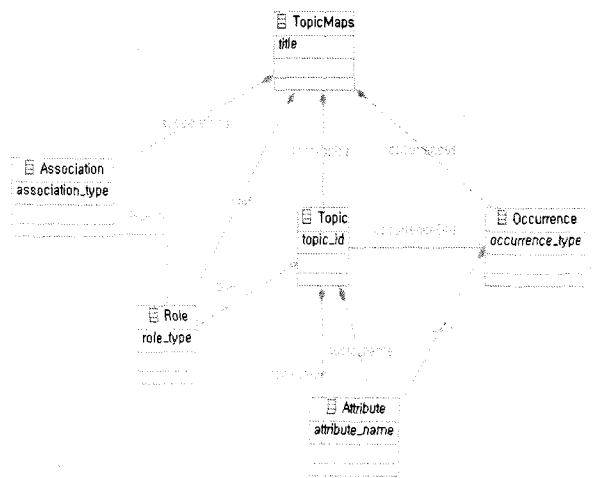


그림 [4-1]. GTM Editor의 Domain Model

4. GTM Editor의 도메인 모델링과 토픽맵구조의 표기 정의

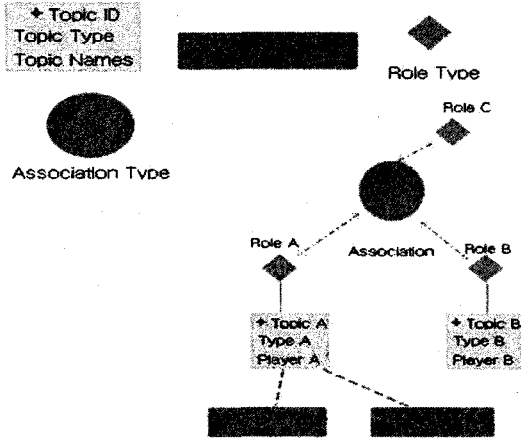
4-1. 토픽맵의 도메인 모델링

4-2. 토픽맵의 도식화된 표기(Graphical Notation)

아래 그림[4-2]은 TAO구조와 역할(role)과 같은 노드들에 관한 표기화 각 노드들간의 연결에 관한 도식화된

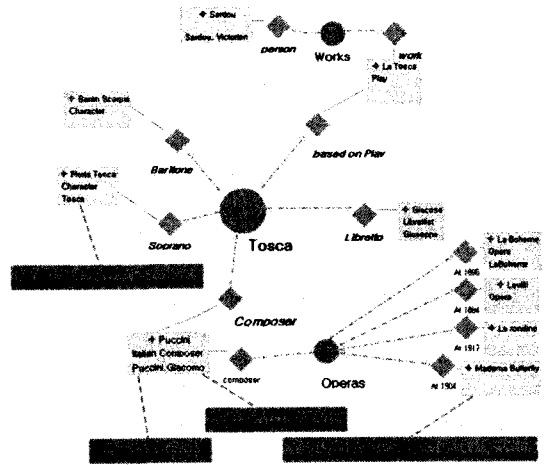
표기들이다.

위의 도메인 모델에서 정의한대로 토픽은 고유한 아이디를 갖고, 타입과 이름들이 있다. 모양은 층이 진 사각형 모양이다. 어커런스는 타입과 데이터를 갖는데, 모양은 둥근 사각형이다. 역할은 타입을 갖고, 마름모 모양이다. 연계는 타입을 갖고, 둥근 모양이다. 연계는 역할과의 연결들로 구성을 이루고, 역할에는 대응되는 토픽이 올 수 있으며, 토픽은 어커런스를 가질 수 있다.



그림[4-2]. 토픽맵의 도식화된 표기

아래 그림[4-3]은 정의한 도식화된 표기를 이용하여, 오페라 토스카를 도메인으로 하여 토픽맵을 구성하여 본 것이다. 토스카는 작곡가(composer)와 소프라노(soprano), 바리톤(baritone), 원작(based on play) 등으로 구성되며, 각각의 역할에 작곡가는 푸치니(Puccini) 등으로 대응되는 실제 토픽들이 있다. 또한 토픽에 대한 자세한 정보는 어커런스에 표현된 URL 등을 통해 찾아볼 수 있다. 또한 하나의 토픽은 다른 주제에 연계되는 모습도 보인다.



그림[4-3]. 오페라 토스카를 중심으로 한 토픽맵 예제

5. GTM Editor

그림[5-1]에 GTM Editor 플러그 인의 전체적인 모습이 있다.(버전 1.0.1) 오른쪽의 Palette 도구 도움을 가지고서, 실제 노드와 연결을 그려줄 수 있다. 크기도 변화 가능하며, 연결의 모양도 노드의 상대적 위치에 따라 가변적이다. 지우기, 만들기, 고치기 등의 기본적인 편집 기능 외에도 여러 유용한 기능을 제공하는데, 확대, 축소와 관련된 기능, 미리 보기 기능, 이미지 파일로 다이어그램의 도식 저장하기, 다이어그램을 위치를 정리해주는 기능, 특정 조건을 만족하는 것들만 보이게 하는 기능 등이 있고, 개발자가 다른 기능을 충분히 개발 가능하다. GTM Editor에서는 어떤 다이어그램을 생성하면 확장자가 topicmaps_diagram, ltm, topicmaps이 자동으로 생성되는데, 각각은 실제 다이어그램의 크기나 위치 등의 실제 도식에 관한 정보, 토픽맵에 데이터의 구조에 대한 것을 LTM으로 표현한 것, GMF와 관련된 데이터의 구조적인 모습을 표현한 모델 파일이다. 그림 [5-2]에서는 그림[5-1]에 표현된 맵을 저장하였을 때 나타나는 LTM 파일을 보여주고 있다.

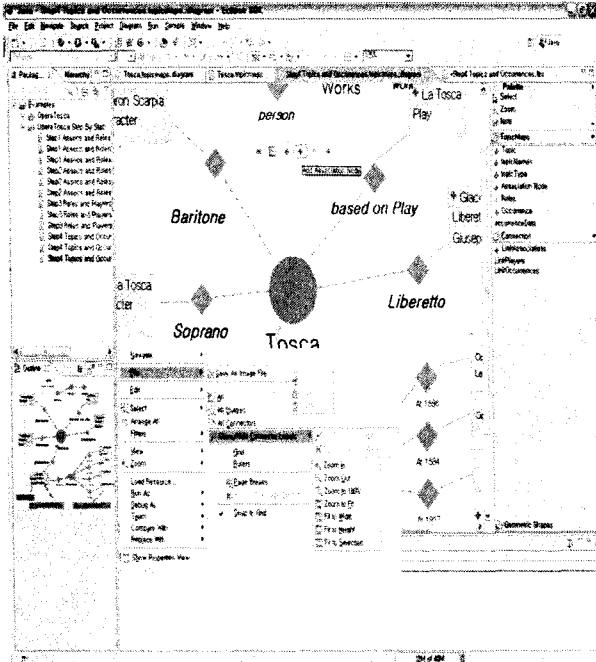


그림 [5-1]. 이클립스내에 플러그인된 GTM Editor의 모습

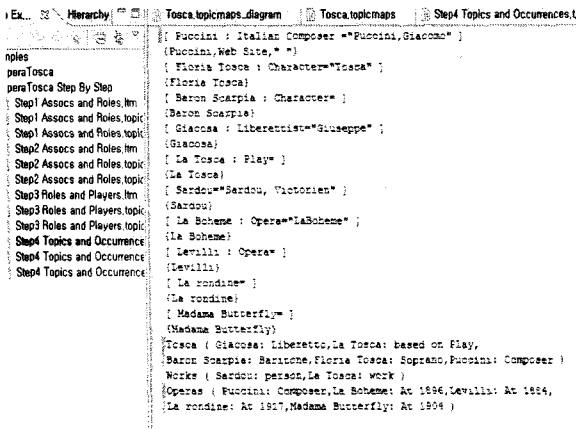


그림 [5-2]. 편집기를 통해 생성된 LTM 파일

6. 결론 및 다음 연구방향

GTM Editor에 적용한 토픽맵 예제를 살펴 보았는데, 비교적 작은 도메인에 대한 맵의 파악에 분명한 도움을 얻을 수 있음을 확인할 수 있었다. 사람을 중심으로 보면, 특정 토픽 찾기에 효율성도 얻을 수 있었고, 토픽간의 연계관계도 직관적인 이해가 가능했다. 그러나 본 편집기의 모델상 그 사용이 작은 용도로 국한 될 수 밖에 없는데, 토픽맵의 본래 목적인 분산된 컴퓨터 환경으로

의 확장이 필요하다. 그러나 LTM구문구조로는 분산돼 있는 중복되는 도메인에 관한 해결할 방법이 없다. URI와 같은 지시자의 사용이 필요하겠고, 구성된 맵 구조에 대한 유효성 검사도 지원해야 하겠다. 또한 지금은 단일 파일과, 작은 도메인에서의 토픽맵에 대한 예를 살펴 보았는데, 큰 도메인의 경우는 어떻게 표현해야 할 것인가에 대한 문제와, 같은 주제가 존재할 경우 이것을 동일시 하는 방법과 여러 토픽맵간의 통합문제를 어떻게 지원할 수 있을까도 과제로 남아 있다.

Acknowledgement

본 연구는 과학기술특정연구개발사업의 인간기능 생활 지원 지능로봇기술개발사업단의 지원을 받았으며 이에 감사 드립니다.

7. 참고문헌

- Jaeho Lee, "Graphical Notations for Topic Maps", 2005
- Steve Pepper, "Introduction to Topic Maps", 2003
- 오삼균, "Introduction to Topic Maps and Its Usefulness in the Semantic Web Environment", 차세대 지식정보공공 국제컨퍼런스2006
- The Topic Map Company Ontopia, <http://www.ontopia.net>
- The Italian Opera Topic Map, <http://www.ontopia.net/operamap>
- Eclipse GMF Project Website, <http://eclipse.org/gmf>
- Eclipse GMF Tutorial, MindMap1, 2, 3
- Eclipse Articles
- Eclipse Help Contents
- Eclipse Mailing List