

MapAppGen: Drupal 기반 맵 응용 생성기

정민경*, 남미진*, 강혜림*, 음두헌**
*덕성여자대학교 컴퓨터공학부
**덕성여자대학교 디지털미디어학과
pongki5833@duksung.ac.kr

MapAppGen : Drupal Based Map Application Generator

Min-Kyung Jeong*, Mi-Jin Nam*, Hye-Rim Kang*, Doohun Eum**
*School of Computer Engineering, Duksung Women's University
**Dept of Digital Media, Duksung Women's University

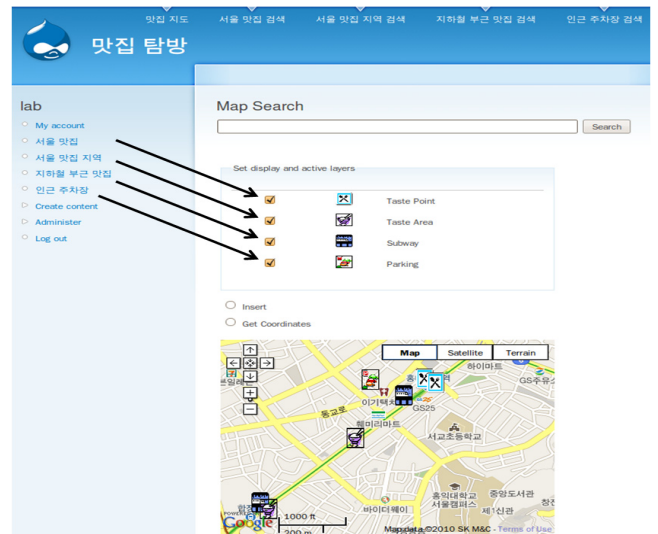
요 약

Google을 시작으로 포털 사이트들이 Maps API를 공개하면서 맵 응용에 대한 수요가 급증하고 있다. 맵 응용의 개발은 대부분 수작업으로 이루어지고 있기 때문에 맵 응용의 생산성이 떨어질 뿐만 아니라 유지·보수에도 많은 시간과 노력이 요구되고 있다. 우리가 설계하고 구현한 MapAppGen은 ModuleGen, IndexGen, MapGen으로 구성되며 맵 응용을 자동으로 생성한다. MapAppGen은 Google Maps API를 사용하여 CMS(Content Management System) 중의 하나인 Drupal에 적용 가능한 모듈들을 생성해 맵 응용에 대한 생산성을 향상시켜 준다. MapAppGen처럼 맵 응용을 위해 Drupal 모듈을 생성하는 생성기는 현재 존재하지 않으며 Gmap이나 NodeMap과 같이 고정된 맵 인터페이스 모듈들은 존재한다. 그러나 Gmap이나 NodeMap 모듈은 Drupal의 기본 콘텐츠 타입 모듈에 의존하기 때문에 사용자는 원하는 지형/지물 콘텐츠 타입을 생성해 활용할 수 없고 지형/지물들을 유형별로 맵 상에 표시하고 연관된 콘텐츠들을 검색할 수도 없다.

1. 서론





맵 응용은 이제 단순한 맵 검색을 뛰어 넘어 사용자의 특정 기호에 맞게 발전하고 있으며 Google을 시작으로 포털 사이트들이 Maps API를 공개하면서 맵 응용에 대한 관심과 수요가 급증하고 있다.[1] 그러나 맵 응용의 작성은 ArcGIS[2]와 같은 값비싼 개발 환경이나 MapServer[3]와 같은 공개 개발 환경을 사용한다고 해도 많은 부분이 아직 수작업으로 이루어지고 있어 맵 응용에 대한 생산성이 떨어질 뿐만 아니라 유지·보수에도 많은 시간과 노력이 요구된다.

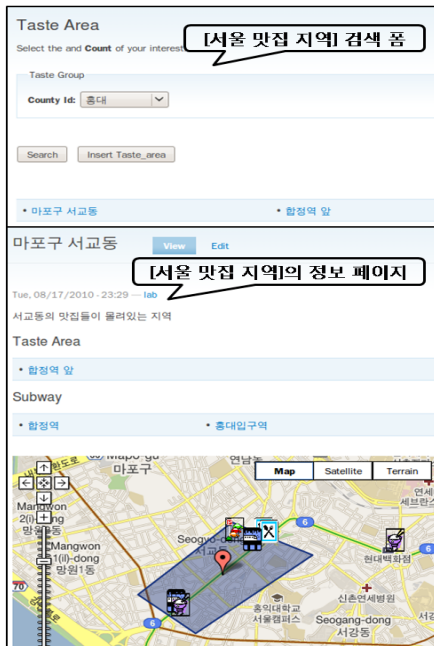
우리가 설계하고 구현한 MapAppGen은 ModuleGen, IndexGen, MapGen 도구들로 구성되며 Google Maps API를 사용하고 CMS(Content Management System) 중의 하나인 Drupal[4]에 적용 가능한 모듈들을 자동으로 생성해 줌으로써 맵 응용에 대한 생산성을 향상시킨다. Drupal은 소셜 네트워킹 사이트의 신속한 개발을 위해 콘텐츠들을 생성하고 구성하며 관리해주는 모듈 기반 시스템으로서 추가적인 기능 및 콘텐츠를 제공하는 모듈을 생성해 Drupal에 적용시킴으로써 그 기능을 확장시킬 수 있다. MapAppGen은 개발자가 정의하는 선언적 구성파일에 따라 Drupal에 적용 가능한 모듈들을 생성한다. ModuleGen 도구는 사용자 정의의 지형/지물 콘텐츠 타입 모듈을 생성한다. IndexGen은 콘텐츠 타입 모듈로 생성된 지형/지물의 신속한 유형별 검색을 지원하는 색인 모듈을



(그림 1) MapAppGen으로 생성된 맛집탐방 응용

생성한다. MapGen은 생성된 지형/지물들을 맵 상에서 마커를 통해 한눈에 볼 수 있고 검색할 수 있게 하는 맵 인터페이스 모듈을 생성한다. MapAppGen의 기능을 설명하기 위해 MapAppGen을 이용해 작성한 그림 1의 맛집탐방 응용을 예로 든다. 그림 1은 서울에 위치한 맛집들의 정보를 제공하는 맛집탐방 응용의 메인 화면을 보인 것이다. 이 메인 화면은 MapGen이 생성한 맵 인터페이스 모듈에 의해 제공되며 맛집 정보는 콘텐츠 타입에 따라 point 형

태의 Taste Point 마커() , polygon 형태의 Taste Area 마커() , 지하철 부근의 맛집을 나타내는 Subway 마커() , 인근 주차장을 나타내는 Parking 마커() 들로 표시된다. 마커의 말풍선에는 각 콘텐츠 제목이 나타나며 이 제목을 클릭하면 해당 콘텐츠의 정보 페이지로 이동된다. 콘텐츠 정보 페이지는 맵 인터페이스를 포함한다. 좌측 메뉴 4개는 ModuleGen으로 생성된 콘텐츠 타입 모듈들이 제공하는 4개의 삽입/편집 폼과 각각 연결되어 사용자가 콘텐츠를 삽입/편집할 수 있게 한다. 그림 1의 상단 메뉴들 중 [서울 맛집 지역 검색]은 IndexGen으로 생성된 색인 모듈이 제공하는 그림 2 상단의 검색 폼과 연결된다.



(그림 2) IndexGen으로 생성된 콘텐츠 검색 폼과 콘텐츠 정보 페이지

Taste Area 타입의 검색 폼인 [서울 맛집 지역] 검색 폼에서 '홍대'로 검색한 결과 중 '마포구 서교동'을 선택하면 하단의 '마포구 서교동'에 대한 정보 페이지로 연결된다. 이 정보 페이지에서 Taste Area 항목은 '마포구 서교동' 콘텐츠와 연관된 Taste Area 타입의 '합정역 앞' 콘텐츠를 함께 검색해 표시한다. Subway 항목은 '마포구 서교동' 콘텐츠와 연관된 Subway 타입의 '합정역'과 '홍대입구역' 콘텐츠를 검색해 표시한다.

이와 같이 각각의 콘텐츠에 대한 정보 페이지를 생성하고 연결하기 위해 사용자는 그림 1의 좌측 메뉴 항목과 연결된 그림 3의 삽입/편집 폼을 이용한다. 그림 3은 그림 1의 [서울 맛집 지역]과 연결된 삽입/편집 폼의 일부를 보인 것이다. Parent Taste Area Id 드롭다운 리스트에는 이미 생성된 [서울 맛집 지역] 타입의 콘텐츠들이 표시된다. 드롭다운 리스트에서 연결하고 싶은 콘텐츠를 선택하면 콘텐츠 생성 시 자동으로 연결된다. 하단의 맵 인터페이스

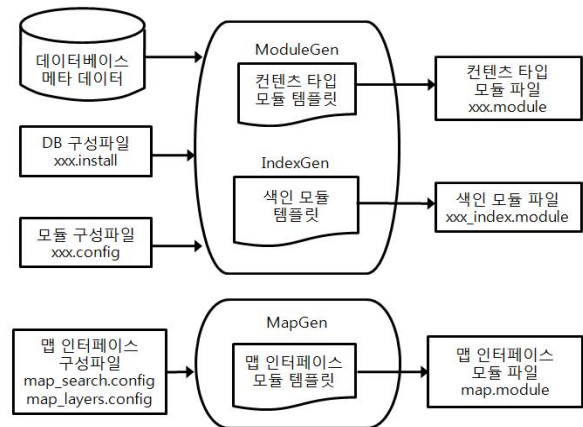


(그림 3) [서울 맛집 지역] 타입의 삽입/편집 폼

이스는 사용자가 point 또는 polygon으로 위치 지정할 수 있게 한다. 2절에서 MapAppGen의 개요를 설명하고 3절에서는 MapAppGen으로 생성된 맵 응용과 Gmap이나 NodeMap을 이용해 생성된 맵 응용을 비교한다. 4절에서는 MapAppGen의 도구들 중 ModuleGen 도구의 구현을 간략히 설명하고 마지막 절에서는 결론과 향후 연구 방향을 정리한다.

2. MapAppGen의 개요

본 절에서는 MapAppGen의 모듈 생성 과정을 설명한다. MapAppGen은 그림 4와 같이 ModuleGen, IndexGen, MapGen 도구들로 구성된다. ModuleGen은 지형/지물 콘텐츠를 생성, 저장, 삭제하는 Taste Area 모듈이나 Taste Point 모듈과 같은 사용자 정의 콘텐츠 타입 모듈을 생성한다.



(그림 4) MapAppGen의 구성

그림 4에서 xxx는 생성될 Drupal 모듈의 이름을 나타낸다. 모듈 구성파일(xxx.config)은 Google Maps API를 사용하기 위해 필요한 Google Maps Key, 드롭다운 리스트에 포함될 속성 등 생성하고자 하는 지형/지물 콘텐츠 타입 모듈에 대한 정보를 제공하는 php 파일이다. 이 파일은 php 문법을 이용한 선언문들을 포함하며 프로그램을

작성하기 위한 것은 아니다. DB 구성파일(xxx.install)은 생성될 지형/지물 콘텐츠 타입 모듈과 연동될 테이블들의 스키마 정보를 제공하는 php 파일이다. 이 테이블들은 사용자 정의의 지형/지물 콘텐츠 타입에 대한 속성들을 저장한다. 콘텐츠 타입 모듈 템플릿은 생성될 모든 콘텐츠 타입 모듈 파일들의 기본 틀을 제공하는 템플릿 파일이다.

ModuleGen은 모듈 구성파일, DB 구성파일 그리고 Drupal이 생성한 데이터베이스의 메타 정보를 입력 받아 그 내용을 내장된 콘텐츠 타입 모듈 템플릿에 적용해 지형/지물 콘텐츠 타입 모듈 파일(xxx.module)을 생성한다. 이 모듈은 지정된 테이블들을 생성하고 콘텐츠 정보 페이지와 삽입/편집 폼을 제공한다.

그림 5은 맛집탐방 응용의 Taste Area 콘텐츠 타입 모듈의 모듈 구성파일을 보인 것이다. \$google_map_key는 Google로부터 배정받은 Google Maps API의 키를 저장한다. \$options_parent_taste_area_id와 \$options_d_county_id에는 삽입/편집 폼의 드롭다운 리스트에 포함될 항목들을 지정한다. \$linked_tables는 연결하려는 테이블에 대한 정보를 배열로 가진다. table_name은 연결하려는 테이블의 이름을 저장하는 변수이다. Taste Area 콘텐츠 타입의 속성들을 저장하는 테이블은 taste_area인데 여기서는 자기 자신인 taste_area 테이블과 Subway 콘텐츠 타입의 속성

```
<?php
( ... 중략 ... )
$google_map_key =
"ABQIAAAAWmi9n0Pep_HNHokjC0ygChQbwIHzPvqgLr2V
TIO4ademzrStaxQNhhLd3GoaViYUdrIoXDOLCG-B5g";
( ... 중략 ... )
$options_parent_taste_area_id = $arResult;
$options_d_county_id = array(
1 =>'가로수길', '강남역', '논현동', '대치동', '도곡동', ...);
$linkd_tables = array(
array(
'table_name' => 'taste_area',
'type' => 'to_many',
'parent_column' => 'parent_taste_area_id'),
array(
'table_name' => 'subway',
'type' => 'to_many',
'parent_column' => 'taste_area_id'),
);
?>
```

(그림 5) Taste Area 콘텐츠 타입 모듈의 구성파일

들을 저장하는 subway 테이블을 명시한다. type은 테이블 간의 관계를 설정하는 변수인데 to_many는 1:N의 관계를 지정한다. parent_column은 연결하려는 테이블의 외래키로서 taste_area 테이블의 parent_taste_area_id 컬럼과 subway 테이블의 taste_area_id 컬럼을 지정한다. 이렇게 하면 그림 2의 [서울 맛집 지역]의 정보 페이지에 표시된 Taste Area 항목과 Subway 항목과 같이 현재 콘텐츠와

연결된 콘텐츠들이 SQL 질의를 통해 검색되어 표시된다.

3. 기능 비교

MapAppGen처럼 Drupal 모듈을 생성하는 생성기는 현재 존재하지 않는다. 따라서 본 절에서는 MapAppGen으로 생성된 세 가지 모듈을 사용해 만든 MapAppGen 응용과 Gmap 응용, NodeMap 응용의 기능을 비교 설명한다. 표 1은 MapAppGen 응용, Gmap 응용과 NodeMap 응용의 기능을 비교한 것이다.

MapAppGen 응용은 그림 1과 같이 콘텐츠 타입별로 레이어를 형성하기 때문에 체크박스를 통해 콘텐츠 타입의 유형별로 검색이 가능하다. 또한, MapAppGen 응용은 그림 2와 같이 한 콘텐츠와 연관된 모든 콘텐츠들을 검색해 표시할 수 있다. 그러나 Gmap 응용과 NodeMap 응용은 유형별 검색은 물론 단순 검색도 지원하지 않는다.

MapModuleGen 응용은 생성될 모듈들의 기본 틀인 템플릿을 기반으로 모듈들을 생성하기 때문에 응용 환경의 변화에 따라 템플릿만 수정해 모듈의 재생성이 가능하기 때문에 생성된 응용의 유지·보수가 Gmap 응용이나 NodeMap 응용에 비해 용이하다.

<표 1> MapAppGen 응용, Gmap 응용, NodeMap 응용의 기능 비교

	Gmap 응용	NodeMap 응용	MapAppGen 응용
연동 DB 서버	MySQL, PostgreSQL	MySQL	PostgreSQL (PostGIS)
유형별 검색	없음	없음	지원
연관된 콘텐츠 표시	없음	없음	지원 (1:1, 1:N, N:N)
위치 정보 형태	point, polygon, line, circle	point	point, polygon
구성 모듈	Location 모듈, Drupal 기본 콘텐츠 타입 모듈	Drupal 기본 콘텐츠 타입 모듈	사용자 정의 콘텐츠 타입 모듈, 색인 모듈, 맵 인터페이스 모듈
유지·보수	+	+	++

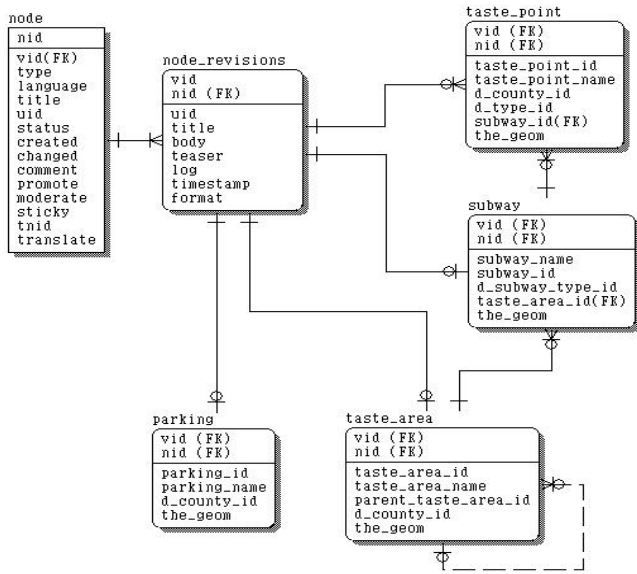
4. MapAppGen의 구현

MapAppGen은 Drupal 6.16, PHP 5.2.6, PostgreSQL 8.3.9, PostGIS 1.3.3-3, Apache 2.2.11, Ajax를 사용해 구현되었다. 본 절에서는 ModuleGen이 생성하는 콘텐츠 모듈 파일을 간략히 살펴본다.

Drupal은 콘텐츠들을 노드들로 구성한다. 노드는 페이지, 스토리, 블로그 항목 등과 같은 콘텐츠 타입의 개체를 나타낸다. Drupal을 설치하면 응용을 위한 기본 소프트웨어 패키지인 코어 모듈들이 포함된다. Drupal 코어는 사용자 관리, 기본 콘텐츠 관리, 테마 등과 같은 기능들을 제

공한다. 노드는 기본적으로 타이틀과 바디의 속성을 가진다. ModuleGen이 생성하는 사용자 정의 콘텐츠 타입 모듈은 콘텐츠 타입의 속성들을 지원하기 위해 기본적인 Drupal 노드의 구조를 그림 6과 같이 확장한다. 그림 6은 ModuleGen으로 생성된 콘텐츠 타입 모듈들이 맛집탐방 응용의 Taste Point, Taste Area, Subway, Parking 콘텐츠 타입의 속성들을 지원하기 위해 개발자가 제공하는 DB 구성과일에 따라 생성하는 taste_point, taste_area, subway, parking 테이블들과 Drupal 노드의 속성을 저장하는 node 테이블의 관계를 보인 것이다.

node 테이블은 노드 식별자(nid)와 함께 노드의 기본 속성들을 포함한다. taste_point, taste_area, subway, parking 테이블들은 node_revisions 테이블을 통해 Drupal



(그림 6) node 테이블, node_revisions 테이블, 사용자 정의 콘텐츠 테이블 간의 관계

의 node 테이블과 연결된다. 한 개의 노드는 여러 개의 확장(revision)을 갖는다. 각 확장에 대한 정보는 확장 식별자(vid)와 함께 node_revisions 테이블에 저장된다. 각 사용자 정의 테이블은 node_revisions 테이블과 vid를 통해 연결된다. 그림 7은 Taste Area 콘텐츠 타입에 대한 그림 5의 모듈 구성과일로부터 생성된 Taste Area 콘텐츠 타입 모듈 파일(PHP 파일)의 일부를 보인 것이다.

```

<?php
( ... 중략 ... )
function taste_area_taste_area_links($nid, $edit = FALSE) {
    $sql = "SELECT n.nid, n.title
    FROM {taste_area} f, {node_revisions} r, {node} n
    WHERE f.parent_taste_area_id = %d and
    f.parent_taste_area_id > 0 and
    f.vid = r.vid and r.nid = n.nid
    
```

```

Order By n.title";
$result = db_query($sql, $nid);
( ... 중략 ... ) ?>
    
```

(그림 7) Taste Area 콘텐츠 타입 모듈 파일의 일부

taste_area_taste_area_links() 함수는 그림 5의 모듈 구성과일 중 \$linked_tables에 명시된 taste_name이 taste_area인 속성을 처리 하기위해 생성된 함수이다. 이 함수는 taste_area 테이블과 자기 자신인 taste_area 테이블의 관계인 1:N(to_many) 관계에 의해 연관 콘텐츠를 검색하기 위한 SQL 질의문을 작성하고 실행한다. 함수의 파라미터인 \$nid는 Drupal node 테이블의 노드 식별자(nid)를 나타낸다. \$sql은 SQL 질의문이 작성되어 저장되는 변수인데 SQL 질의문은 사용자가 연결시키려는 콘텐츠의 노드 식별자 값을 가져오기 위해 SELECT문으로 구성된다. 이 SELECT문에서는 연결하려는 콘텐츠의 노드 식별자에 해당하는 nid와 콘텐츠명에 해당하는 title을 검색하는데 FROM절에는 taste_area 테이블, node 테이블, 두 테이블을 연결하는 node_revisions 테이블들이 명시된다. WHERE절에는 4가지 조건이 명시된다.

5. 결론

Drupal CMS(Content Management System)를 기반으로 하는 MapAppGen은 맵 응용을 위해, Drupal 모듈들을 자동 생성해 주는 도구로서 맵 응용의 생산성을 향상시키고 생성된 응용의 유지·보수를 용이하게 한다. MapAppGen은 사용자 정의의 지형/지물 콘텐츠 타입 모듈을 생성하는 ModuleGen, 사용자의 유형별 지형/지물 검색을 위해 색인 모듈을 생성하는 IndexGen 그리고 모든 콘텐츠 타입 모듈들의 정보를 가지고 생성된 지형/지물들을 맵 상에서 마커를 통해 한눈에 보고 검색할 수 있게 하는 맵 인터페이스 모듈을 생성하는 MapGen으로 구성된다.

MapAppGen이 생성하는 모듈들은 Drupal에서 제공하는 Gmap이나 NodeMap과 같은 고정된 맵 모듈과 비교하여 표 1과 같은 장점을 가진다.

향후, MapAppGen은 개발자의 구성과일을 통한 기술 대신 향상된 GUI를 제공하는 위저드를 통한 명세로 맵 응용의 생산성을 더욱 향상시킬 예정이다.

참고문헌

[1] 심화영, http://www.dt.co.kr/contents.html?article_no=2008040902010531727002, 디지털타임즈, 2008.
 [2] ESRI, <http://www.esri.com/software/arcgis/index.html>, 2010.
 [3] Open Source Geospatial Foundation, <http://mapserver.org/>, 2010.
 [4] Matt Butcher, Learning Drupal 6 Module Development, PACKT, 2008.