



Софийски университет „Св. Кл. Охридски“

Факултет по математика и информатика

*Бакалавърска програма
„Софтуерно инженерство“*



Предмет: XML технологии за семантичен Уеб

Зимен семестър, 2020/2021 год.

Тема 045: Преглед на езика Geography Markup Language (GML)

Есе

Автори:

Янислав Янков, фак. номер 62357

Павлина Тодорова, фак. номер 62324

ноември, 2020

София

Съдържание

1	Въведение.....	3
2	Характеристики и използване на GML	4
2.1	Дефиниции	4
2.2	Основни характеристики	4
2.2.1	GML профили	5
2.2.2	Схеми на употреба	5
2.2.3	GML текст	6
2.2.4	GML геометрия	6
2.2.5	GML свойства (фийчъри)	6
2.2.6	GML Координати	7
2.2.7	Координатна референтна система	7
2.3	Въведение в използването на езика	8
2.4	Ограничения при използването на GML.....	9
2.5	От кои технологии зависи GML.....	9
3	Сравнителен анализ	9
3.1	Критерии за сравнение.....	9
3.2	Сравнение със SVG	10
3.3	Сравнение с KML	11
3.4	Сравнение с VML	12
3.5	Сравнение с GeoJSON.....	13
4	Примери на използване	14
4.1	Кодиране на геометрични обекти в GML 1.0 и 2.0	14
4.2	Представяне на сграда без свойства	15
4.3	Представяне на сграда със свойства	15
4.4	Изображения, кодирани с GeoSciML.....	16
4.5	Изображения, кодирани със CityGML.....	17
4.6	Изображения, кодирани със GPlates	18
5	Добри практики и методи за използване	19
6	Заклучение и очаквано бъдещо развитие.....	19
7	Разпределение на работата	20
8	Използвани литературни източници	20

1 Въведение

Настоящият документ има за цел да представи възможностите, приложенията, както и структурата на езика GML.

Geography Markup Language (GML) е XML базиран програмен език за кодиране на географска информация, разработен от Open Geospatial Consortium (OGC). GML се използва за моделиране на географски системи, служи и като формат за отворен обмен на географски транзакции в Интернет.

Създателят на GML Рон Лейк, започва разработването на езика през 1998 година. Той представя своята работа за първи път на среща на OGC през Февруари 1999 година под името xGML. GML се справя безупречно с третирането на пространствената информация. Той е съвместим с уеб-мрежата и е първият, който предоставя публичен стандарт за кодиране на пространствени данни.

Програмният език е способен да интегрира най-различни форми на географска информация. За да може да се изразяват географските данни с GML, се създава специфична XML схема. Тя описва типове обекти, които са основната част от приложенията, създадени чрез GML. Използването им е от голямо значение, тъй като в някои сфери има много специфични ситуации, които не може да се пренебрегнат и трябва да се подходи с голямо внимание към детайла. Това обаче далеч не означава, че този език не е способен да се използва и за по-обща цели. Ето някои от най-популярните стандартизирани модели за интерпретация на данни чрез GML:

- GeoSciML се използва в геологията най-често за създаване на геоложки карти и диаграми
- GPlate е софтуер с отворен код, който предлага интерактивно представяне на движението на тектоничните плочи.
- CityGML е софтуер, който позволява да се направят 3D модели на градове и поддържа много обекти, които можем да срещнем в градовете като например сгради, пътища, реки, мостове и растителност.

В следващите точки ще разгледаме по-подробно описание на GML. Представени са характеристиките и използване на GML, основни обекти, с които GML работи, както и примери. След което сме съпоставили таблично езика GML със SVG, KML и VML. Есето завършва с очакваното бъдеще развитие на езика и използваните от нас източници.

2 Характеристики и използване на GML

GML се занимава с представянето на географски данни. Тези данни обаче могат да бъдат представени и графично, но за да бъде постигнато това, се нуждаем от допълнителен инструмент за рендиране, който тълкува GML данните. Това е един начин за изработване на карта, но той противоречи с идеите на GML да стандартизира и отделя съдържанието и представянето. За да изработим карта от GML по друг начин, трябва да оформим обектите от GML във форма, която може да бъде интерпретирана от уеб браузър. SVG (Scalable Vector Graphics – „мащабируема векторна графика“), VML (Microsoft Vector Markup Language) и X3D (безплатен отворен стандарт за публикуване, преглед, печат и архивиране на интерактивни 3D модели в мрежата) са графични формати, които могат да генерират карта от GML, чрез локализиране на GML елементите и тълкуването им, използвайки конкретни графични стилове.

2.1 Дефиниции

- Географски данни - Те са кодирани в GML и чрез тях изразяваме информация относно характеристики и геометрични свойства на заобикалящите ни обекти.
- Геометрия на обект – съвкупност от елементи на GML - Points, LineStrings, Polygons.
- Графична интерпретация на географските данни - изобразяване на географските данни върху карта чрез цветове и линии.

2.2 Основни характеристики

GML съдържа богат набор от примитиви, които се използват за изграждане на схеми или приложения за конкретни приложения. Тези примитиви включват:

- Фийчър (обект на приложение)
- Геометрия
- Координатна референтна система
- Топология
- Време
- Динамичен фийчър
- Покритие (включително географски изображения)
- Мерна единица за измерване

- Посока
- Наблюдения
- Правила за оформяне на картата

Оригиналният модел на GML се основава на Resource Description Framework (RDF). По-късно OGC въвежда XML схеми в структурата на GML, за да помогне за свързването на различните съществуващи географски бази данни. Въпреки промяната, GML запазва много функционалности на RDF, включително идеята за дъщерни елементи като свойства на родителския обект (RDFS) и използването на отдалечени препратки към свойства.

2.2.1 GML профили

GML профилите са логически ограничения на GML, които могат да бъдат изразени чрез документ, XML схема или и двете. Предназначението им е да улеснят приемането на GML и на неговия стандарт. Профилите са част от GML пространствата от имена. Те имат за цел да опростят използването на езика, разделяйки го на подмножества, които съдържат само елементите, нужни за енкодирането на дадената задача. Профилите се създават в подкрепа на извлечените от GML езици. Те се разпростират в различни области на приложение като например търговско въздухоплаване, навигационни карти или експлоатация на ресурсите. Следните профили са одобрени за публична употреба:

- Point Profile - Профил на точка - използва се за приложения с точкови геометрични данни, които нямат нужда от цялата граматика на GML
- Simple Features profile - поддържа заявки и транзакции на векторни функции (например с Web Feature Service).
- GML профил за GMLP2 (GML in JPEG 2000)
- GML профил за RSS

2.2.2 Схеми на употреба

Илюстрирането на географските данни на приложението в GML става с т.нар. схема на употреба. Тя е специфична за целите и темата на разглежданото приложение. Схемата описва типовете обекти, чиито данни представляват интерес за него. Например, схема на употреба за приложение за туризъм ще описва обектни типове, които включват музеи, галерии, пътища, забележителности. Тези типове обекти реферират към примитивните типове на обекти, които са описани в GML стандарта. Важно е да се прави разлика между профилите и схема на употреба. 10 Профилите са подмножества на целия език, а схемите са XML речник, деклариран чрез GML и те пребивават в пространство от имена, дефинирано за приложението. Видове схеми на употреба са:

- AIXM - Aeronautical Information eXchange Model – модел за обменяне на информация в авиацията

- CAAML - Canadian Avalanche Association Markup Language – език, използван от канадската асоциация за борба с лавини
- CityGML – GML схема на употреба за виртуални 3D обекти
- WXXM - Weather Information Exchange Model – модел за обменяне на информация за метеорологичните условия

2.2.3 GML текст

GML представя географската информация като текст, също както XML представя информация като текст. Тази идея доскоро е била неприемлива, но в момента набира все повече популярност. Текстът е добър, защото е лесен за разбиране и представяне визуално. Също така е лесно да се анализира и променя. А когато се добави XML към него може да бъде контролиран.

2.2.4 GML геометрия

GML кодира географските характеристики на географските обекти. Това се случва съгласно модела “вектор”. Географски обекти, които могат да бъдат описани като елементи в GML документа са например пътища, реки, мостове, сгради. Основни типове геометрични обекти в GML 1.0 и GML 2.0 са:

- Point (Точка)
- LineString (Линия)
- Polygon (Многоъгълник)

В GML 3.0 и по-новите версии има много повече информация от сензори и сателитни данни, които позволяват ползването на растерна графика и по-добро описание на обхвата.

2.2.5 GML свойства (фийчъри)

GML дефинира свойства, различни от геометричните обекти. Свойство е обект на приложение, който представлява физическо лице, напр. сграда, река или човек. Един фийчър може или не може да има геометричен аспект. За разлика от фийчърите, един геометричен обект дефинира локация или регион вместо физическа единица. В GML, един фийчър може да има различни геометрични свойства, които описват геометрични аспекти или характеристики на фийчъра. Например, фийчър на сграда в конкретна схема на приложение на GML може да има позиция, дадена от примитивния тип обект на геометрия - GML Point. Сградата обаче е отделен обект от Точката, който определя нейната позиция. В допълнение, обектът на приложението може да има няколко геометрични свойства (или изобщо да няма), например степен и позиция.

2.2.6 GML Координати

Координатите в GML представляват координатите на геометричните обекти. Координатите могат да бъдат изразени, чрез някои от следните елементи на GML:

- `<gml:coordinates>`
- `<gml:pos>`
- `<gml:posList>`

2.2.7 Координатна референтна система

Координатна референтна система(CRS) определя геометрията на всеки геометричен елемент в GML документ. GML не приема координатна система по подразбиране ако не му се предостави такава. Вместо това, желаната координатна система трябва да се посочи изрично със CRS (Common Reporting Standard - глобален стандарт за обмяна на информация). Елементите, чиито координати се интерпретират по отношение на такава CRS, включват:

- `<gml:coordinates>`
- `<gml:pos>`
- `<gml:posList>`

Атрибутът *srsName* добавен към геометричен обект определя CRS на този обект, както в този пример:

```
<gml:Point gml:id="p1" srsName="#srs36">  
  <gml:coordinates>100,200</gml:coordinates>  
</gml:Point>
```

Изображение 1 (Източник [2])

Стойността на атрибут *srsName* е Uniform Resource Identifier(URI). Тя реферира дефиницията на CRS, който се използва за интерпретация на координатите в геометрията. CRS дефиницията може да е в документ или онлайн ресурс.

URI на *srsName* може да бъде и Uniform Resource Name(URN) за рефериране на чести CRS дефиниции. OGC са разработили URN структура и задават специфични URN за кодиране на някои често използвани CRS. URN преобразувател преработва тези URN-и до дефиниции на GML CRS.

2.3 Въведение в използването на езика

2.3.1 Пренос поведение чрез GML

Самият GML не задава поведение, въпреки това може да бъде използвано в комбинация с езици като Java и C++ за пренасяне на географско поведение от едно място на друго. Това може да бъде постигнато използвайки прост factory обект, който ще инстанцира обекти на базата на получените GML данни и ще map-ва имената на GML елементи в класове обекти. (Заб. В много приложения често се налагат поведения само за малък брой елементи.

2.3.2 GML кодира референтни системи

Съществена част от географската система е възможността за рефериране на географските особености към земната повърхност или някаква структура, която е свързана със земната повърхност. Текущата версия на GML включва земна базова пространствена референтна система, която може да се разширява и включва основни проекционни и геоцентрични референтни рамки, които се използват днес. Освен това схемата за кодиране позволява потребителят да дефинира единици и параметри на референтната система.

Оказва се, че кодирането на пространствената референтна система не е необходимо всеки път. Можем да свършим същата работа и чрез предоставяне на уникално име. Или иначе казано не се изисква изпращачът на географски данни да изпраща и кодиране на референтната система, към която се отнасят координатните стойности на данните. Но съществуват случаи, в които тази информация е много ценна, а именно защото:

- Клиентска валидация - клиентът може да поиска описанието на XML документ и да го сравни със собствените си спецификации или да го покаже на потребител за проверка
- Показване на клиент на определена от сървъра пространствена справочна система
- Използване на Службата за трансформиране на координати за потвърждаване на пространствена справочна система(ПСС) за входни данни.
- Службата за трансформиране на координати може да сравни описанието на ПСС със собствените си спецификации, за да види дали ПСС е в съответствие с избраната трансформация.
- Контрол на автоматизираната трансформация на координатите чрез предоставяне на имената на референтни системи за вход, изход и стойности на аргументи.

2.3.3 GML кодира геометрия и свойства

GML описва света около нас чрез географски обекти, наречени функции. Една функция обаче не е нищо повече от списък от свойства и геометрии. Свойствата имат име, тип и стойност. Геометриите са съставени от точки, линии, криви, повърхности и многоъгълници. В началото спецификацията на GML е ограничена до 2D геометрия, но по-късно се появява 2,5 и 3D геометрия.

GML кодирането вече позволява и по-сложни функции. Една функция може да се състои от други функции. Така например една функция летище ще може да съдържа в себе си подфункциите писти, терминали, таксиметрови седянки. Също така един географски обект може да бъде представен чрез няколко геометрии. Например от точка и линия или линия и многоъгълник.

2.4 Ограничения при използването на GML

GML документите, също както и XML документите, трябва да бъдат валидирани. Това се осъществява чрез DTD формат (Document Type Definition). DTD описва структурата на XML документа, така че един валидиращ парсер може да прецени дали XML документът е валиден. При GML валидирането е аналогично. В началото GML е можел да бъде валидиран само чрез DTD, но малко по-късно XML Schema също започва да валидира GML документи.

2.5 От кои технологии зависи GML

GML се базира на XML. Въпреки че за XML се казва, че може да бъде заменен от HTML, той се смята за най-добрия език за описание на данни. Въпреки това, XML не е език за програмиране, защото той не извършва никакви изчисления, а само описва данни. Изчислението обикновено се извършва от езици като Java и C++.

3 Сравнителен анализ

В тази точка ще сравним GML с други подобни езици като SVG, KML и VML, GeoJSON. В следващите подточки са описани по-подробно какво представляват съответно SVG, KML, VML и GeoJSON, както и сравнение с GML, представено в табличен вид. Но първо ще започнем с критериите, по които сравняваме тези езици.

3.1 Критерии за сравнение

Избрахме следните критерии, по които да сравним езиците:

- На кой език са базирани разглежданите от нас езици;
- Кой е съзателят на съответния език;

- Година на създаване;
- Какво е разширението на файловете за тези езици;
- С какви пространства работи езикът;
- Под каква форма се представя информацията от съответният език;
- Стандарт, на който отговаря дадения език;
- Приложения на дадения език.

3.2 Сравнение със SVG

Scalable Vector Graphics (SVG) или мащабируема векторна графика е език базиран на XML, който описва двумерна векторна графика. SVG се поддържа от всички модерни браузъри за десктоп и мобилни устройства. SVG се използва в много различни сфери като например уеб графики, анимация, потребителски интерфейс, мобилни приложения. Графичните обекти могат да бъдат групирани, оформени, трансформирани в предишно направени обекти.

Таблица 1: Сравнение на GML със SVG

Критерий	GML	SVG
Базиран на:	<ul style="list-style-type: none"> • XML 	<ul style="list-style-type: none"> • XML
Създаден от:	<ul style="list-style-type: none"> • Open Geospatial Consortium 	<ul style="list-style-type: none"> • World Wide Web Consortium
Година на създаване:	<ul style="list-style-type: none"> • 1998 	<ul style="list-style-type: none"> • 1999
Разширение:	<ul style="list-style-type: none"> • .xml • .gml 	<ul style="list-style-type: none"> • .svg • .svgz
Пространство:	<ul style="list-style-type: none"> • 2D • 2.5D • 3D 	<ul style="list-style-type: none"> • 2D
Основни форми:	<ul style="list-style-type: none"> • Точка • Линия • Многоъгълник 	<ul style="list-style-type: none"> • Пътеки (прави линии, многосегментни линии) • Правоъгълници • Окръжности • Елипси • Многоъгълници • Изображения • Текст
Стандарт:	<ul style="list-style-type: none"> • ISO 19136 	<ul style="list-style-type: none"> • W3C SVG
Приложение:	<ul style="list-style-type: none"> • GeoSciML • GPlate • CityGML 	<ul style="list-style-type: none"> • Съхранение на 2D изображения • 2D анимации

Макар GML да е по-старият от двата езика и да предлага повече пространства за работа, SVG е много по-популярен, защото се разпространява и поддържа от W3C.

3.3 Сравнение с KML

Keyhole Markup Language (KML) описва географски обекти в Интернет: 2D и 3D карти на земята. Google Earth е първата програма, която може да чете и променя KML файлове. KML става стандарт на OGC през 2008. Файловете с разширение .kmz всъщност са архивирани KML файлове, които трябва да бъдат разархивирани преди да се ползват. Всяко място, което се изобразява на картите, направени от геопространствен софтуер, има дължина и ширина. Допълнително описание на даденото място може да направи изгледа още по-конкретен като добави надморска височина, наклон и изглед от камера с времеви маркер.

Таблица 2: Сравнение на GML с KML

Критерий	GML	KML
Базиран на:	<ul style="list-style-type: none"> • XML 	<ul style="list-style-type: none"> • XML
Създаден от:	<ul style="list-style-type: none"> • Open Geospatial Consortium 	<ul style="list-style-type: none"> • Keyhole
Година на създаване:	<ul style="list-style-type: none"> • 1998 	<ul style="list-style-type: none"> • 2008
Разширение:	<ul style="list-style-type: none"> • .xml • .gml 	<ul style="list-style-type: none"> • .kml • .kmz
Пространство:	<ul style="list-style-type: none"> • 2D • 2.5D • 3D 	<ul style="list-style-type: none"> • 2D • 3D
Основни форми:	<ul style="list-style-type: none"> • Точка • Линия • Многоъгълник 	<ul style="list-style-type: none"> • Отбелязване на място • Снимка • Многоъгълник • Описание на текстури
Стандарт:	<ul style="list-style-type: none"> • ISO 19136 	<ul style="list-style-type: none"> • OGC KML 2.3
Приложение:	<ul style="list-style-type: none"> • GeoSciML • GPlate • CityGML 	<ul style="list-style-type: none"> • Google Earth • Marble

От горната таблица можем да забележим, че GML е по-старият език и използва по-примитивни начини за представяне на информацията, което обаче пък позволява по-голяма свобода от гледна точка на пространства, с които може да се работи.

От друга страна KML е много по-често използван, защото той е в основата на Google Earth, което е водещо приложение в световен мащаб в сферата на картографията.

3.4 Сравнение с VML

Vector Markup Language (VML) е XML-базиран файлов формат за двумерна векторна графика. VML е включен в W3C през 1998 от Autodesk, Hewlett-Packard, Macromedia, Microsoft и Vision. По същото време и други конкурентни формати, като например PGML, са включени в W3C. В резултат на това се създава нова W3C работна група, която създава SVG. По-късно през 2001 SVG става препоръчан от W3C език за описание на двумерна векторна и смесена векторна/растерна графика в XML. От тогава и VML се приема за остарял. VML се поддържа в Internet Explorer от версия 5 до версия 9. От версия 10 нататък не се поддържа. Широко използван е в MS Office 2007 документи (например Word, Excel и PowerPoint).

Таблица 3: Сравнение на GML с VML

Критерий	GML	VML
Базиран на:	<ul style="list-style-type: none">• XML	<ul style="list-style-type: none">• XML
Създаден от:	<ul style="list-style-type: none">• Open Geospatial Consortium	<ul style="list-style-type: none">• Autodesk• Microsoft Corporation• Hewlett-Parker• Macromedia• Vision
Година на създаване:	<ul style="list-style-type: none">• 1998	<ul style="list-style-type: none">• 1999
Разширение:	<ul style="list-style-type: none">• .xml• .gml	<ul style="list-style-type: none">• .htm• .html
Пространство:	<ul style="list-style-type: none">• 2D• 2.5D• 3D	<ul style="list-style-type: none">• 2D
Основни форми:	<ul style="list-style-type: none">• Точка• Линия• Многоъгълник	<ul style="list-style-type: none">• Форма• Тип на формата• Запълване на форма• Път• Сянка• Текстово поле
Стандарт:	<ul style="list-style-type: none">• ISO 19136	<ul style="list-style-type: none">• ISO/IEC29500
Приложение:	<ul style="list-style-type: none">• GeoSciML• GPlate• CityGML	<ul style="list-style-type: none">• MS Office 2007:• MS Word• MS PowerPoint• MS Excel

3.5 Сравнение с GeoJSON

GeoJSON е отворен стандартен формат, предназначен за представяне на прости географски свойства заедно с техните непространствени атрибути.

Таблица 4: Сравнение на GML с GeoJSON

Критерий	GML	GeoJSON
Базиран на:	<ul style="list-style-type: none">• XML	<ul style="list-style-type: none">• JSON
Създаден от:	<ul style="list-style-type: none">• Open Geospatial Consortium	<ul style="list-style-type: none">• Geograpic JSON working group
Година на създаване:	<ul style="list-style-type: none">• 1998	<ul style="list-style-type: none">• 2016
Разширение:	<ul style="list-style-type: none">• .xml• .gml	<ul style="list-style-type: none">• .json• .geojson
Пространство:	<ul style="list-style-type: none">• 2D• 2.5D• 3D	<ul style="list-style-type: none">• 2D
Основни форми:	<ul style="list-style-type: none">• Точка• Линия• Многоъгълник	<ul style="list-style-type: none">• Точки• Прави• Многоъгълници• Множества от точки• Множества от прави• Множества от многоъгълници• Геометрични колекции
Стандарт:	<ul style="list-style-type: none">• ISO 19136	<ul style="list-style-type: none">• RFC 7946
Приложение:	<ul style="list-style-type: none">• GeoSciML• GPlate• CityGML	<ul style="list-style-type: none">• ArcGIS• OpenLayers• Leaflet• MapServer• Geoforge software• GeoServer• GeoDjango• GDAL• Safe Software FME• CartoDB

GeoJSON пренася битката между XML и JSON в света на географията и архитектурата. GeoJSON е нов и модерен начин за интерпретиране на прости географски обекти. GeoJSON е по-бърз от GML и популярността му бързо нараства. Също така, съществува разширение за GeoJSON – TopoJSON, което съхранява топологично данни и предоставя още по-голяма гъвкавост.

4 Примери на използване

В тази точка ще разгледаме няколко примера, които ще дадат яснота относно това как се използва GML.

4.1 Кодирание на геометрични обекти в GML 1.0 и 2.0

- Многоъгълник:

```
<gml:Polygon>
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>
```

Изображение 2 (Източник [2])

- Точка:

```
<gml:Point>
  <gml:coordinates>100,200</gml:coordinates>
</gml:Point>
```

Изображение 3 (Източник [2])

- Линия:

```
<gml:LineString>
  <gml:coordinates>100,200 150,300</gml:coordinates>
</gml:LineString>
```

Изображение 4 (Източник [2])

Горните три изображения са основните геометрични обекти в GML. Чрез тях се описват други по-сложни обекти, които се използват за направата на сгради, пътища, летища и други географски обекти на картата.

4.2 Представяне на сграда без свойства

```
<Feature fid="142" featureType="school" Description="A middle school">
  <Polygon name="extent" srsName="epsg:27354">
    <LineString name="extent" srsName="epsg:27354">
      <CDATA>
        491888.999999459,5458045.99963358 491904.999999458,5458044.99963358
        491908.999999462,5458064.99963358 491924.999999461,5458064.99963358
        491925.999999462,5458079.99963359 491977.999999466,5458120.9996336
        491953.999999466,5458017.99963357 </CDATA>
      </LineString>
    </Polygon>
  </Feature>
```

Изображение 5 (Източник [1])

На горното изображение имаме представено училище, което, чрез параметъра Description, разбираме, че е средно училище. Едно училище представяме с помощта на основните обекти в GML, а именно многоъгълник и линия. Тук нямаме предоставени свойствата на сградата.

4.3 Представяне на сграда със свойства

```
<Feature fid="142" featureType="school" >
  <Description>Balmoral Middle School</Description>>
  <Property Name="NumFloors" type="Integer" value="3"/>
  <Property Name="NumStudents" type="Integer" value="987"/>
  <Polygon name="extent" srsName="epsg:27354">
    <LineString name="extent" srsName="epsg:27354">
      <CDATA>
        491888.999999459,5458045.99963358 491904.999999458,5458044.99963358
        491908.999999462,5458064.99963358 491924.999999461,5458064.99963358
        491925.999999462,5458079.99963359 491977.999999466,5458120.9996336
        491953.999999466,5458017.99963357 </CDATA>
      </LineString>
    </Polygon>
  </Feature>
```

Изображение 6 (Източник [1])

Същия примера както в 4.1, но тук се използват свойства, които задават етажите и броя на учениците в съответното училище.

4.4 Изображение, илюстрирано с GeoSciML



Изображение 7 (Източник [12])

GeoSciML (GeoScience Markup Language) е схема на приложения, която се използва за пренасяне на информация относно геологията на дадена област. Включва информация относно геоложките единици, геоложки структури и др.

4.5 Изображения, иллюстрирани със CityGML



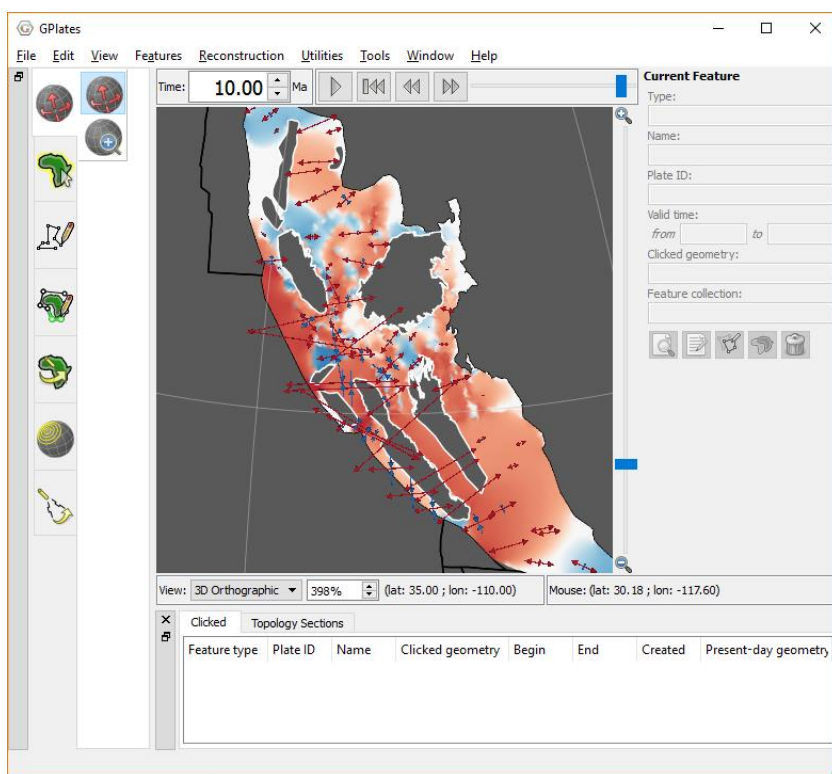
Изображение 8 (Източник [13])



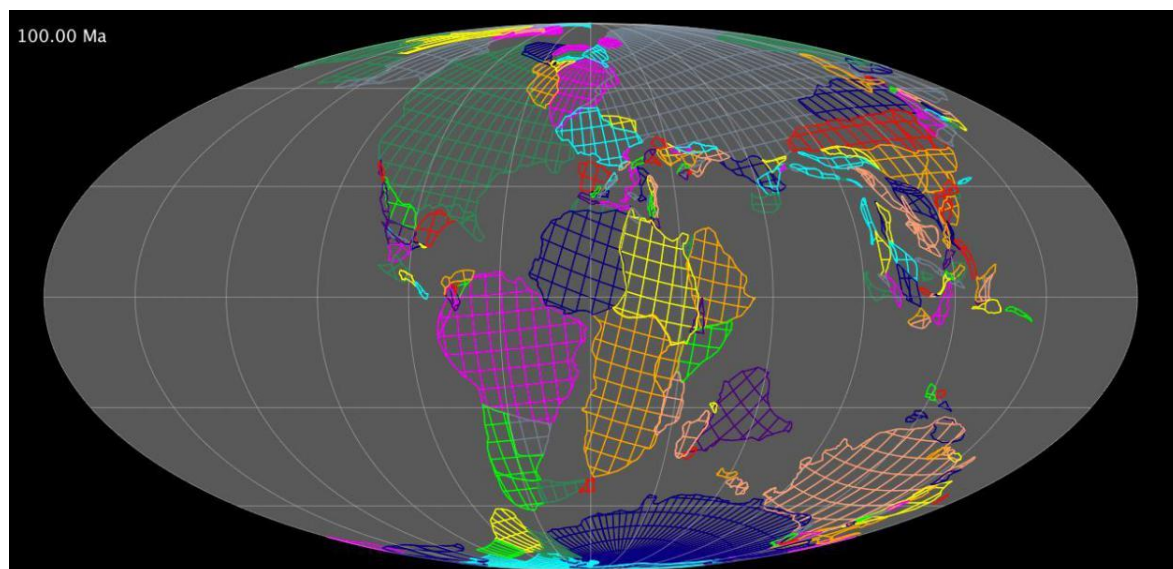
Изображение 9 (Източник [14])

CityGML е отворен стандартизиран модел на данни за съхраняване на 3D модели на градове и пейзажи. Той дефинира начини за описание на най-разпространените 3D функционалности и обекти, които могат да се намерят в града(сгради, пътища, реки, мостове, растителност и др.) и връзките между тях.

4.6 Изображения, илюстрирани с GPlates



Изображение 10 (Източник [15])



Изображение 10 (Източник [16])

GPlates дава възможност както за визуализация, така и за манипулиране на тектонични реконструкции на плочи и свързаните с тях данни през геоложко време. GPlates работи на Windows, Linux и MacOS. GPlates има онлайн ръководство за потребителя.

5 Добри практики и методи за използване

Добрите практики за използване на GML са следните:

- Използването на референции, с което се избягва повтарянето на код
- Добро и систематизирано описание на обектите
- Изготвяне на план преди да бъдат описани обектите

1.Оформлението на текста е много важно, защото лесно може да се достигнат няколко хиляди реда код, а той трябва да бъде лесно четим. Някои от добрите практики за добре форматиран код са следните:

- Всеки нов таг да бъде на нов ред
- Всеки таг да бъде с нужния отстъп, за да бъде четим кода

2.Редактирането на атрибути и релации в един GML документ може да бъде доста трудно, а понякога може да се наложи да бъде пренаписан целия проект. За да не се стига до нужда от редактиране по време на разработка, са необходими:

- Предварително подробно планиране на съдържанието на документа
- Добро планиране на семантиката на елементите в обекта, както и техните взаимоотношения

3.Многото код прави документа тежък и труден за обработка. За да се избегне огромния обем на файлове можем да:

- Използването на референции
- Внимаваме какви данни въвеждаме, за да избегнем повторения

4.Писането на документация прави документите използвани от хора, които не са ги разработвали и много улеснява поддръжката му, затова е добра практика да се:

- Пише документация на проекта
- Поддържа документацията на проекта

6 Заключение и очаквано бъдещо развитие

В днешно време, хората масово спират да използват карти в хартиен формат. На тяхно място обаче, се появяват все повече софтуерни приложения, които да ни помагат да се ориентираме. Тези приложения трябва да бъдат много точни и все по-лесни за направа и използване. Точно затова ни трябва език, който да успее да прави всичко това.

GML служи както като език за моделиране на географски системи, така и като отворен формат за географски транзакции в Интернет. Това, което е полезно в GML, е способността му да интегрира всякакви форми географска информация - не само стандартни "вектори" и дискретни обекти, но и покрития(GMLJP2) и данни от сензори.

GML е мощен начин за разглеждане на пространствена информация, използвайки XML. Това прави GML нещо повече от обикновен стандарт за кодиране. Достъпността на GML го прави много привлекателен за управлението на гео-пространствената информация.

7 Разпределение на работата

- Янислав Янков:
 - Примери за използване
 - Добри практики и методи за използване
 - Заключение и бъдещо развитие
 - Презентация
- Павлина Тодорова:
 - Въведение
 - Сравнителен анализ.
 - Характеристики и използване
 - Оформяне на документа

8 Използвани литературни източници

В тази точка сме представили източниците на информация, които сме използвали за направата на това есе.

1. <https://www.w3.org/Mobile/posdep/GMLIntroduction.html>
2. https://en.wikipedia.org/wiki/Geography_Markup_Language
3. <https://www.opengeospatial.org/standards/gml>
4. <https://bg.wikipedia.org/wiki/SVG>
5. https://en.wikipedia.org/wiki/Keyhole_Markup_Language
6. https://en.wikipedia.org/wiki/Vector_Markup_Language
7. <https://en.wikipedia.org/wiki/GPlates>
8. <https://www.gplates.org/>
9. <https://en.wikipedia.org/wiki/GeoSciML>
10. <https://en.wikipedia.org/wiki/CityGML>
11. <https://en.wikipedia.org/wiki/GeoJSON>
12. <https://www.bgs.ac.uk/news/geoscienceml-data-standard-becomes-official/>
13. https://www.ogc.org/blog/2196?utm_source=twitterfeed&utm_medium=twitter
14. <https://3d.bk.tudelft.nl/projects/geobim-benchmark/amsterdamgml.html>
15. <https://www.gplates.org/screenshots.html>
16. <https://twitter.com/KirkDBorne/status/544215389538750464/photo/2>