

Магістрант Воронцов В.П., к.т.н., доцент Білостоцький А.І.

**Національний технічний університет України
«Київський політехнічний інститут»**

ДИНАМІЧНА СУБД-ОРІЄНТОВАНА XML-МОВА

Abstract

Victor Vorontsov, student

XML Database Dynamic Language

This paper concerns the technique of design and development of the interpreter-based XML-syntax language, which contains a dynamically configurable namespace. A newcomer to the XML-based languages contains in the nonlinearity of the program, by allowing the transfer of information from tag to tag by means of a special operator. The uniqueness of this work is the ability to interpret XML-code into any object-oriented programming language, also in the interpretation and storage of code on the database level. Given approach can be taken for these purposes.

Вступ

XML (Extensible Markup Language)[1] - це мова розмітки, що описує цілий клас об'єктів даних, названих XML-документами. Ця мова використовується в якості засобу для опису синтаксису інших мов і контролю за правильністю складання документів. Більшість нових версій відомих мов програмування до числа своїх переваг зараховує саме підтримку XML[2].

Але на даний момент XML розглядається з боку програмістів скоріше як контейнер для даних і компонентів, який асоціюється тільки з лінійними інструкціями. У даній роботі ми розглянемо розробку інтерпретатора, що використовує синтаксис XML, який буде являти собою повноцінну компонентно-орієнтовану мову, з операторами присвоювання (є стандартними для XML-мов), розгалуження і циклу.

У даній роботі було розглянуто формування динамічного XML і введено поняття динамічної СУБД-орієнтованої XML-мови (XML-DBDL - XML Database Dynamic Language).

Постановка задачі

Задача полягає у розробці методики, що дозволить у максимально короткий час розробити екземпляр XML-мови програмування, орієнтованої

на вирішення класу задач(наприклад: обробка даних, робота з математичним апаратом, інструмент для ефективної розробки клієнтського інтерфейсу та інше), що інтерпретує на виході готовий об'єктно-орієнтований код необхідної мови.

Концепція динамічної СУБД-орієнтованої XML-мови(XML-DBDL)

Основним нововведенням даної роботи є динамічно налаштований простір імен, і велика кількість можливих екземплярів мов, які, у свою чергу, можуть бути орієнтовані на вирішення абсолютно різних завдань. XML-DBDL містить у собі:

- Оригінальний XML-код даних примірників;
- Правила розбору примірників мов, що містяться в СУБД;
- Інтерпретаційні блоки даної мови, що містяться в СУБД;
- Інструмент формування вихідного інтерпретованого коду.

По суті, XML-DBDL є повноцінним гнучконалаштовувемим інтерпретатором, який включає в себе етапи: розбір XML-синтаксису, перевірка коректності синтаксису в розрізі вибраної мови, створення проміжних даних на рівні СУБД і реалізація макроподстановок на рівні вибраної мови, формування і зберігання інтерпретованого коду в вигляді вихідної мови, зберігання та виконання сформованого інтерпретованого коду.

У зв'язку з цим в XML-DBDL були сформульовані концепції, з яких можна виділити:

1) XML-DBDL - це сукупність даних, що зберігаються в СУБД, які поділяються на 3 групи:

- Первинні коди програм XML-формату;
- Дані налаштування інтерпретатора;
- Вихідні коди програм, отримані в результаті обробки інформації.

2) В XML-DBDL можна виділити 3 типи даних:

- Дані тегів і дані зв'язків;
- Дані атрибутів;
- Дані макропідстановок, тобто оперування даними тега на рівні xml-документа.

3) Вихідні коди програм XML-DBDL є коректним XML-документом.

4) Імена атрибутів тега не можуть збігатися з іменами можливих підтегів тега.

5) різних мов має бути об'єктно-орієнтованим, тому що найбільша гнучкість моделі досягається тільки при роботі з об'єктами.

6) Модель тегів мови задається в деревоподібній формі, яка допускає цикли, тобто глибина вкладень тегів може прямувати до нескінченності. При цьому, якщо тег допускає підтеги, то він не може містити в собі значення, але атрибути допускаються.

Цілісність лексичної моделі XML-DBDL

Поряд зі службовою інформацією, яка задає структуру динамічного документа, вводяться для користувача реквізити, що містять інформацію, актуальну для користувача в той чи інший момент життєвого циклу. Ці реквізити мають складну структуру, неодноразово зустрічаються в документі і можуть бути зв'язані між собою. У зв'язку з цим необхідний механізм для забезпечення цілісності цих реквізитів. Були виділені чотири групи обмежень цілісності:

- Структурності - у вигляді обмежень на форму структури;
- Обов'язковості - у вигляді умови наявності значень реквізитів;
- Посильності - у вигляді наявності для одних реквізитів - інших реквізитів з відповідними значеннями;
- Балансу - у вигляді обмежень на спільні значення декількох реквізитів. [2]

У силу специфіки XML-DBDL, обмеження цілісності балансу реалізується фахівцем з налаштування конкретного екземпляру мови XML-DBDL, тобто на даному етапі розглянуто не буде. Обмеження цілісності посильності, для досягнення гнучкості налаштування і використання, буде спрощено до формулювання:

- Посильності - у вигляді наявності для одних реквізитів - інших реквізитів з відповідними значеннями;

Виникає питання, як вирішити ці завдання в середовищі XML, максимально використовуючи надані цією технологією можливості. У зв'язку з цим в роботі були виділені два класи обмежень цілісності:

- стандартні, реалізовані за допомогою схем XML-документа (декларативний підхід);
- нестандартні, що мають місце в тих випадках, коли інструментарію схем недостатньо і необхідні якісь додаткові механізми, що реалізують ці правила (недекларативний підхід). [2]

тобто для максимальної гнучкості XML-лексики мови необхідно обмежити недекларативний підхід, причому реалізація недекларативну рішень повинна розглядатись як введення нових елементів XML-синтаксису

конкретної моделі, причому цей елемент може бути необов'язковим. У XML-DBDL цей елемент тільки один, і він реалізує динамічну гнучкість всієї системи: оператор взаємозв'язку даних тегу "#". Він не є обов'язковим для тегів і атрибутів, але може гнучко і гармонійно використовуватись, і розглядається частиною синтаксису XML-DBDL-коду.

До стандартних обмежень цілісності можна віднести правила структурності, обов'язковості, посиленості. Оскільки оператор "#" може зустрічатися як в тегах, так і в атрибутах, при цьому не впливаючи на вищезгадані правила, то доповнимо правило посиленості:

– Посиленості - у вигляді наявності для одних реквізитів - інших реквізитів з відповідними значеннями оператора взаємозв'язку даних "#";

Таким чином, цілісність лексичної моделі XML-DBDL досягнута.

Обмеження структурності в XML-DBDL реалізуються за допомогою опису в екземплярі мови припустимих тегів, припустимих атрибутів для кожного тега і ієрархічних взаємовідносин тегів. Обмеження обов'язковості реалізується як на рівні тегів, так і на рівні атрибутів, за допомогою даних про структуру мови. Обмеження посиленості - так само реалізується виходячи з даних про структуру мови, а також виходячи з валідності звернення до певного тега за рахунок оператора "#", причому отримання даних за допомогою цього оператора можливо тільки при описі даної можливості тега в синтаксисі розбору мови.

Висновки

На даний момент на ринку програмного забезпечення знаходяться сотні різних мов програмування зі своїм синтаксисом і специфічною направленістю на вирішення ряду задач (наприклад: обробка даних, робота з математичним апаратом, інструмент для ефективного розробки клієнтського інтерфейсу та інше). Причому з кожним роком процент об'єктно-орієнтованих мов програмування все зростає. При цьому зазвичай програміст витрачає від пари місяців до пів року на оволодіння необхідною для нього мовою програмування. Оскільки XML є розповсюдженою та інтуїтивно-зрозумілою мовою програмування, то на засвоєння технології програмування на підвиді XML-мови займає у програміста на багато менше часу, ніж на мовах високого рівня на кшталт Java, C# та інших.

Запропонована методика дозволяє уникнути проблем довгого засвоєння нової мови програмування, так як XML-DBDL може інтерпретувати будь-яку мову програмування, таким чином позбавивши програмістів необхідності постійно вивчаючи нові мови програмування.

Література

1. *Bray T., DeRose S.* Extensible Markup Language (XML):Part 2. Linking. W3C Working Draft – <http://www.w3.org/>, 1997. – С. 1.
2. *Миронов В.В., Шакирова Г.Р.* Интерпретация XML-документов со встроенной динамической моделью – Уфа: УГАТУ, 2007. – С. 1-4.
3. *Кайт Т.* Oracle для профессионалов. Пер. с англ. – ТомКайт- СПб.: ООО «ДиаСофтЮП», 2003. – С. 37-136.
4. *Кайт Т.* <http://asktom.oracle.com>