

## РОЗРОБКА ПЗ ДЛЯ АВТОМАТИЗАЦІЇ РОЗРОБКИ SCORM-СУМІСНИХ КУРСІВ

О. І. Гороховський, к.т.н., доц. каф. ОТ, Вінницький національний технічний університет, м. Вінниця, Україна

Т. І. Трояновська, викл., Вінницький технічний коледж м. Вінниця, Україна

**Вступ.** Стандарт *SCORM* версії 2004 р. [1] є складним і великим стандартом, який визначає, як саме повинен бути організованим курс дистанційного навчання (КДН). Однак, він не містить конкретних рецептів (за винятком частини *RTE*, яка містить опис взаємодії між клієнтом дистанційного навчання та сховищем даних) реалізації цих курсів.

**Мета роботи.** З метою забезпечення можливості розробки *SCORM*-сумісних курсів, пропонується розробити додаткове програмне забезпечення (ПЗ), яке є «надбудовою» на основі даного стандарту, і дозволяє користувачу абстрагуватись від складних вимог стандарту, і проектувати курс відповідно до навчального процесу.

### **Основна частина роботи.**

Стандарт *SCORM* визначає КДН як набір компонентів (*assets*), що організовані в єдине ціле. До переліку компонентів входять практично всі види електронних документів, які є «спільними об'єктами» (*SCO - Sharable Content Object*) і можуть бути включеними в декілька різних уроків, модулів або курсів. Об'єднує і відображає компоненти, як цілісну систему веб-агент (у більшості користувачів це інтернет-браузер, що відображає *HTML*-документи).

Існує два варіанти, при яких спільні агенти можуть «сшиватись» в єдине ціле. Перший: наявність потужної серверної платформи, яка на базі маніфест-файлів «конструює» ДКН на льоту (*Moodle, Sakai* та *WebTutor*). І другий: впорядкований набір веб-сторінок, з'єднаних між собою навігаційними інструментами. Обидва шляхи визначені як частина *SCORM* (відповідно частини *CAM* та *SN*), однак, конкретних методів реалізації не передбачено. Саме для цього було розроблено дане ПЗ – як **надбудова для автоматизації розробки SCORM-сумісних курсів**. Для обох методів відображення КДН найбільш важливими є два види компонентів: 1. веб-документи (*HTML*-сторінки); 2. *XML*-документи, які описують метадані та інформацію, необхідну для динамічного конструювання КДН.

Ядром КДН є веб-сторінка, яка інтегрують компоненти, і ПЗ, що пропонується, призначене саме для її розробки. Відповідно до стандарту *SCORM*, пакет повинен формувати ієрархію у такому порядку: Пакет-Курс-Модуль-Урок-Тест. Зупинимося на рівні курсу.

Проектний менеджер розробленого ПЗ [2] створює будь-яку ієрархію в межах ДКН. Модуль може бути як цілісною веб-сторінкою, так і вузлом в ієрархічній структурі проекту. Модулі з'єднані в єдиний КДН за

допомогою навігаційної системи на базі *JavaScript*. Проект може бути вивантажено в архівний файл шляхом генерації *XML*-файлів «на льоту» для розгортання на серверній платформі. Таким чином, дане ПЗ дозволяє користувачу працювати на перспективу і без додаткових затрат переходити від локальних курсів до мережевих. Дане ПЗ виконує роль візуальної надбудови над *SCORM*. Додавання всіх видів контенту, включаючи скрипти, реалізовано в формі *WYSIWYG*, із можливістю перемикання в режим *HTML*-коду. Це дає можливість контролювати проект на низькому рівні (навіть на відповідність певному *SCORM*-сумісному серверному ПЗ), та проектувати КДН чи його частину, тут же візуалізуючи його кінцеву форму.

З точки зору *SCORM* компонент (*asset*) може бути або простим (відео-файл), або складеним (*HTML*-сторінка з *CSS* та *JavaScript*-модулями). Для користувача такий компонент повинен відображатись як єдине ціле, тоді як автор ДКН повинен бачити всі складові. «Подвійна» модель відображення, прийнята в даному ПЗ, дозволяє вирішити цю проблему.

На випадок переходу ДКН від локальної до мережевої форми в *SCORM* передбачено кілька моделей на базі *SCO*. Для забезпечення функціоналу *SCO* можна використати скрипти. Тоді роль *SCO* гратиме спеціальна сторінка, яка не містить контенту, але динамічно підвантажує фрагменти ДКН. Це досягається за допомогою «плаваючих фреймів». Тобто, фактично сама сторінка не змінюється, а змінюється тільки вміст фрейму (рисунок 1).

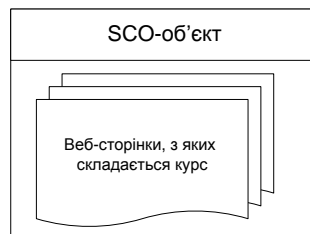
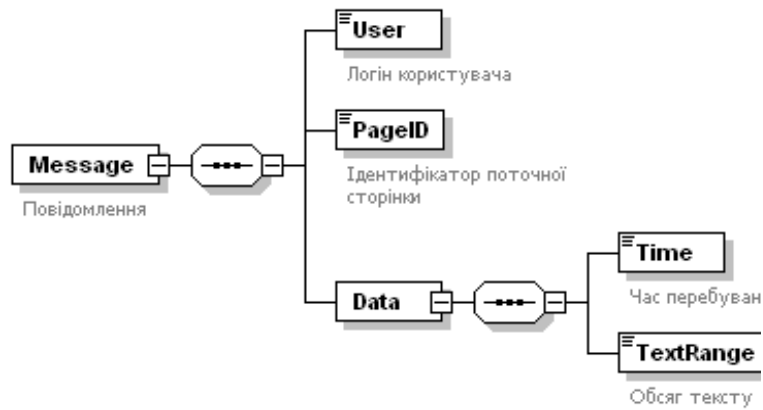


Рис. 1 – Схема плаваючих фреймів

Це розвантажує серверну частину, і дозволяє кожному отримати свої власні результати (при необхідності підлаштувати курс під себе). Запропоновані скрипти не відповідають точним специфікаціям *SCORM API*, і не включає в себе *API\_1484\_11*. Натомість, створений його аналог на базі аплета *Java*. Тому було адаптовано стандартну архітектуру *SCORM* і розроблено спеціальний **протокол обміну даних** на основі *XML* для забезпечення контролю за навчальним процесом. Ядром рішення є *Java*-апплет, який також є *SCO*-об'єктом. Він присутній на кожній сторінці ДКН, і виконує такі задачі: 1. Ініціалізація та аутентифікація; 2. Заміри показників процесу навчання; 3. Динамічна навігація; 4. Передача даних про перебіг навчання викладачу; 5. Коректне завершення курсу і збереження результатів. Для передачі даних використовується спеціальний протокол на базі документів *XML*. Розглянемо його схему (рисунок 2):



Generated by XmlSpy

www.altova.c

Рис. 2 – Схема повідомлення

Для кожної події формується повідомлення, яке містить метадані користувача. Розроблений протокол відносяться до класу документ-орієнтованих, і реалізує обмін по відкритому протоколу. Такий підхід лежить в основі *XML Web Services*. Розроблений протокол діє на прикладному рівні стеку протоколів, і використовує у якості носія даних протокол *UDP*. Загальний алгоритм дії цього протоколу можна описати наступною послідовністю: 1. Виникнення події; 2. Зняття необхідних показників; 3. Формування *XML*-документу; 4. Шифрування (необов'язково); 5. Формування пакету даних *UDP*; 6. Відправка пакету. Це повністю відповідає вимогам *SCORM RTE* та *SN*.

**Висновок.** Розроблене ПЗ є візуальною надбудовою над стандартом *SCORM* та реалізує навігацію на базі *JavaScript*, генерує *XML*-файли «на льоту», реалізує «подвійну» модель відображення контенту, «плаваючі фрейми» та зовнішнє розташування *SCO* і т.ч. дозволяє користувачу працювати на перспективу і без додаткових затрат праці переходити від локальних курсів до мережевих. Це розвантажує серверну частину, і реалізує індивідуальний підхід до студента. Для передачі даних розроблений протокол на базі *XML*.

### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Sharable Content Object Reference Model 2004 3rd Edition. Advanced Distributed Learning, October 20, 2006.
2. Гороховський О. І., Трояновська Т. І., Трухманов В. О. Комп'ютерна програма «Адаптивна система дистанційної освіти» / Свідectво про реєстрацію авторського права на твір № 26330 ПА. України - Заявлено 22.07.2008; Видано 31.10.2008.