

В.В. Герасимов, А.П. Пеньков, Г.О. Пушков

АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ НАСИЧЕНИХ ІНТЕРНЕТ-ДОДАТКІВ НА ПЛАТФОРМІ JAVA

Анотація. Проведено аналіз технології JavaFX 2, яка активно розвивається в останній час. Наведено особливості цієї технології, розглянуто перспективи її розвитку. Представлені результати порівняння сучасних технологій розробки насичених Інтернет-додатків на різних платформах.

Ключові слова: RIA, JavaFX, Flash, Silverlight, FXML, Java, Groovy, JRuby, Scala, JVM, AJAX, інтерактивність, платформа, технологія, інтерфейс.

Вступ. З розвитком комп'ютерних технологій виділився окремий напрямок — розробка насичених (або багатих) Інтернет-додатків (Rich Internet Application, RIA). Це особливий клас додатків, який містить в собі симбіоз настільних (desktop) та веб-додатків. Він має кращі властивості цих зовсім різних напрямків: легкість в установці, безпека роботи, відсутність необхідності в підтримці програми, багаті можливості інтерфейсу, інтерактивність. Основним напрямком використання цієї технології є розробка клієнтської частини великих проєктів, яка зв'язується з сервером і виконує певні бізнес-функції. Також можна реалізовувати й інші окремі додатки — невеликі ігри, презентації та ін.



Рисунок 1 - Rich Internet Applications серед технологій створення програмних систем

Існує конкуренція між технологіями, за допомогою яких розробляються RIA-додатки. До останнього часу з боку Java-технології не було достойної відповіді для Flash від Adobe, Silverlight від Microsoft, однак з релізом JavaFX 2.0 ситуація змінилася і у Java-розробників з'явився конкурентоспроможний інструмент. Тепер всі частини проекту можуть бути повністю реалізовані на одній платформі Java, що забезпечує швидкодію, підвищує безпеку та скорочує час розробки, а звідси й ціну.

Метою роботи є дослідження технології JavaFX 2 та проведення порівняльного аналізу її з іншими технологіями розробки RIA-додатків.

Основна частина. Якщо звернутися до історії, то Java-аплети були найпершою RIA-технологією, яка стала доступною ще в 1995 р. Проте аплети не отримали широкого розповсюдження. Згодом з'явилась технологія JavaFX 1.x, проте й вона повторила історію з Java-аплетами. 10 жовтня 2011 року корпорація Oracle випустила JavaFX 2.0 — вдосконалену платформу для створення інтерфейсу користувача в бізнес-додатках корпоративного класу і наступного етапу еволюції Java в якості головної клієнтської повнофункціональної платформи.

На конференції JavaOne 2010 Oracle продемонстрували довгостроковий план для JavaFX (рис. 2). Початок схеми — грудень 2011 року.

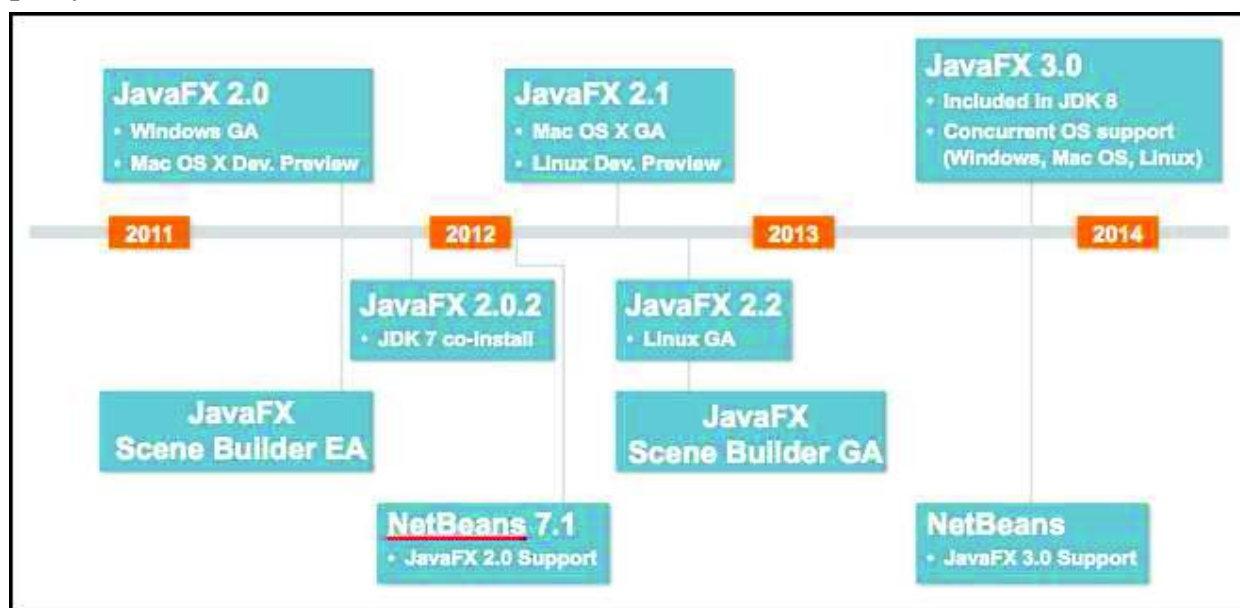


Рисунок 2 - Схема розвитку технології JavaFX

JavaFX 2.0 дає розробникам гнучкі можливості для створення додатків з використанням бажаних інструментів та мов програмування. Так в ній вводиться FXML — мова з підтримкою сценаріїв, основана на мові розмітки XML і призначена для опису користувальницьких інтерфейсів. Розробники, знайомі з Web-технологіями або іншими мовами розмітки на базі XML, легко освоюють FXML і його можливості.

JavaFX 2.0 дозволяє застосовувати популярні мови сценаріїв, підтримувані віртуальною машиною Java (JVM), такі як Groovy, JRuby і Scala, та об'єднує простоту динамічних мов з функціональністю платформи Java. JavaFX 2.0 надає Web-компонент, який працює на базі движка web-рендерингу Webkit і дозволяє розробникам комбінувати і поєднувати традиційні можливості Java і динамічні можливості Web-технологій. При створенні додатків JavaFX розробники можуть використовувати бажані інструменти Java-розробки, наприклад, NetBeans і Eclipse. При проектуванні інтерфейсу користувачів розробники можуть використовувати спеціальну програму JavaFX Scene Builder.

Майбутні релізи JavaFX будуть основними етапами для розвитку JavaFX не тільки з точки зору функціональності, але й розглядаються як інтеграція в Java SE. Ось деякі з ключових нових функцій, які, як планується, будуть частиною майбутніх релізів JavaFX:

- Тісна інтеграція з Java SE;
- Покращення UI Controls та Charts;
- Підтримка сервісів даних;
- Покращення WebView;
- Модульність;
- Підтримка доступності;
- Краща підтримка 3D графіки;
- Multi-Touch та підтримка жестів;
- Підтримка датчиків.

Порівняємо технологію JavaFX з основними її конкурентами. По кожній властивості технології буде виставлятися кількість набраних балів («+») від 1 до 5. Властивості для порівняння наступні:

- Досвід користувачів — цей критерій показує рівень інтерактивності та швидкість відгуку: human-computer interaction (HCI). На основі цього будується інтерактивна модель додатку;

- Розгортання — описує рівень комфортності при розгортанні додатку (створення релізу, інсталяція, активація, оновлення, адаптація, деінсталяція та ін.);
- Забезпечення широкої доступності — критерій описує такі речі як кількість користувачів, поширеність та доступність;
- Багатотехнологічність — критерій взаємодії або її відсутності з іншими технологіями;
- Час запуску — час, необхідний для старту роботи додатка;
- Клієнт – Сервер — порівнюються процеси на сервері або клієнті;
- Платформонезалежність;
- Підтримка розробників — цей критерій стосується інструментів розробки та мов програмування;
- SEO, доступність — можливість автоматизованого індексування та доступність для людей-інвалідів;
- Роздільна здатність екрану — критерій незалежності від розділювальної здатності екрану;
- Насиченість UI — використання стандартних компонентів, кастомізація, скіни, стилі, анімація та ін.;
- Вартість — критерій вартості розробки, ліцензії, вартість або безкоштовність використання технології.

Таблиця 1

Порівняння технологій

Властивості	AJAX	Flash	Silverlight	JavaFX
Досвід користувачів	++	++++	+++	+++ +
	AJAX, відносно досвіду користувачів, революційно замінив звичайний DHTML, однак він є дуже залежним від парадигми браузера. Silverlight порівняно з іншими підтримує меншу кількість браузерів. Flash ще не досконало взаємодіє з Adobe Integrated Runtime (AIR) для переносу веб-додатків у десктоп-додатки. Під час запуску Java-додатків дещо погіршується швидкість відгуку додатку.			
Розгортання	++++	++++	+++	+++
	Центральне розгортання серверу пропонують всі технології. AJAX не потрібно оновлювати, тому що він поставляється з браузерами. Тому він має найменший бар'єр для використання користувачами веб-додатку. Flash і JavaFX пропонують автоматичний оновлювач. Але Java при онов-			

	ленні перериває на довший час потік робіт (workflow) користувача.			
Забезпечення широкої доступності	+++++	++++	+++	+++ +
	AJAX вбудований в усі браузери. Flash є доступним у 97 % браузерів, іноді інсталується з операційною системою. Java також в більшості випадках інсталується. Silverlight менш поширений.			
Багатотехнологічність	+	+++	++++	+++ +
	AJAX не вистачає взаємодії між різними стандартами та платформами і різними мовами програмування. Flash, JavaFX та Silverlight — поодинокі технології, які пропонують єдине середовище виконання, яке забезпечує однакову поведінку на багатьох платформах. Silverlight дозволяє обирати мову для реалізації. В JavaFX також це доступно через використання таких мов JVM, як Groovy і Scala.			
Час запуску	+++++	+++++	++++	+++
	Головною рисою AJAX є те, що він одразу підтримується браузерами і запуск додатків на ньому миттєвий. Flash також дуже швидкий. Silverlight трохи повільніший. Час запуску Java потребує декількох кроків і тому він більший за інших в цьому списку.			
Клієнт – Сервер	++	++++	+++	+++ ++
	AJAX пропонує асинхронну комунікацію і зменшення навантаження на сервер, але не має сокет-з'єднання. Flash має сокет-з'єднання з AIR. Silverlight та Flash забезпечують швидку взаємодію з користувачем, але ця технологія націлена на виконання на боці клієнта. Java технологія добре підходить для серверної сторони та пропонує однорідну комунікацію між сервером та RIA-клієнтом.			
Платформонезалежність	+++++	++++	+++	+++ +
	Серед всіх Silverlight підтримує меншу кількість платформ, Flash та JavaFX є кращими за нього, але в той же час AJAX має ширші можливості майже на усіх мобільних пристроях.			
Підтримка розробників	+++	++++	++++	+++ ++
	Для AJAX розробки існують допоміжні інструменти, але їх не багато і вони не є кросплатформними. Допоміжні інструменти розробки Silverlight та Flash є платними, а інструменти розробки JavaFX є безкоштовними.			
SEO, доступність	+++	+	+	+

	AJAX пропонує найкращі показники SEO та доступності. JavaFX, Silverlight та Flash потребують використання HTML-альтернатив.			
Роздільна здатність екрану	+	++++	++++	+++ +
	AJAX може використовувати тільки браузерну піксельну технологію. JavaFX, Silverlight та Flash використовують векторну графіку.			
Насиченість UI	+	+++++	++++	+++ +
	AJAX впроваджує стилювання за допомогою CSS (каскадні таблиці стилів), але не має детального стилювання компонент та недостатньо реалізує концепцію «Look and Feel» (вигляд і поведінку). Flash порівняно з JavaFX та Silverlight в цьому плані пропонує більше.			
Вартість	+++	++	++	+++ +
	Для деяких браузерів потрібна інша реалізація додатку на AJAX і це потребує додаткових витрат. Flash та Silverlight не є безкоштовними технологіями, як і засоби для їх розробки. JavaFX та інструменти для розробки таких додатків є безкоштовними, з відкритим кодом.			
Результат	35	44	38	45

Таким чином, не дивлячись на відносно невелику швидкість запуску додатків, написаних за допомогою JavaFX, ця технологія одержує найбільшу кількість балів завдяки безкоштовності технології та інструментів розробки, великої кількості Java-спільноти та підтримки різних мов програмування. Silverlight не отримав широкої популярності і через проблеми цю технологію більше не будуть розвивати. Не дивлячись на невелику кількість набраних балів AJAX є дуже популярним, але його більш використовують для підвищення інтерактивності та комфортності при роботі з веб-ресурсами (соціальні мережі, сайти, інтернет-магазини), ніж для розробки RIA додатків. Більшість RIA-додатків розроблені на Flash: це склалося історично — інші альтернативні технології не могли на той час бути конкурентоспроможними, але на даний момент ситуація змінюється, популярність Flash падає, технологія JavaFX знаходить своїх прихильників і кількість розроблених додатків за допомогою цієї технології нестримно росте.

Висновки. Корпорація Oracle зробила великий крок вперед в напрямку RIA, що є дуже доброю новиною для Java-розробників. Технологія має багато різних реалізованих елементів інтерфейсу та

класів, які дуже прискорюють розробку додатків. Цю технологію також можуть використовувати розробники Ruby, Scala та інших мов, основаних на JVM. Великою перевагою JavaFX є те, що додаток, написаний за допомогою цієї технології, може мати декілька режимів розгортання і працювати на різних типах пристроїв, що робить цей додаток дуже універсальним і при цьому заради цієї універсальності не потрібно нічого робити додатково.

Однак JavaFX також має свої недоліки. Головним серед них є відносно великий час запуску додатку. Також через те, що версія 2.x кардинально відрізняється від версії 1.3, JavaFX 2.x можна розглядати як тільки-но створену технологію. І тому поки що дуже багато заплановано, але реалізовано не так багато, інструменти графічної розробки не дуже комфортні, мало досвідчених розробників та відносно мало інформації — тобто наявні всі риси, присутні новим технологіям. На даний момент доступною є версія JavaFX 2.2.3, в якій виправлена велика кількість недоліків та помилок, і розробники цієї технології можуть перейти до наступного кроку — впровадження нових ідей та можливостей до наступних версій JavaFX.

ЛІТЕРАТУРА

1. Florian Moritz. Rich Internet Applications (RIA): A Convergence of User Interface Paradigms of Web and Desktop Exemplified by JavaFX — Diploma Thesis / University of Applied Science Kaiserslautern — Saarbruecken, 2008 — P. 140.
2. JavaFX Documentation Home. Getting Started with JavaFX — Режим доступу:
http://docs.oracle.com/javafx/2.0/get_started/jfxpub-get_started.htm
3. JavaFX Documentation Home. JavaFX Architecture and Framework — Режим доступу:
<http://docs.oracle.com/javafx/2.0/architecture/jfxpubarchitecture.htm>
4. JavaFX Documentation Home. JavaFX Overview — Режим доступу:
<http://docs.oracle.com/javafx/2.0/overview/jfxpub-overview.htm>
5. Алексеев А. А., Савельев А. О. Технологии Microsoft для создания RIA-приложений — Режим доступу:
<http://www.intu-it.ru/departament/se/tmsria/>