

**Technologie RIA
w bankowości
elektronicznej
na przykładzie
Verax eBanking Suite**
Whitepaper



Spis treści

Streszczenie	3
1. Wprowadzenie	4
2. RIA – nowe podejście do aplikacji internetowych	5
2.1. Opis technologii	5
2.2. Korzyści dla użytkownika końcowego.....	6
2.3. Korzyści techniczne.....	8
2.4. Korzyści dla programistów.....	9
3. Verax eBanking Suite	10
3.1. Informacje o produkcie	10
3.2. Przegląd funkcjonalności.....	10
3.3. Verax Banking Suite – bezpieczeństwo.....	10
3.4. Verax Banking– integracja.....	11
3.5. Verax Banking– ładowanie oraz zależność od pasma	11
3.6. Verax Banking– skalowalność	12
3.7. Verax Banking Suite – od strony użytkownika.....	12
3.8. Wielojęzyczność	13
4. Model biznesowy	13
5. Przyszłość	14
5.1. Prognozy	14
5.2. Adobe AIR™	14
6. Podsumowanie	15
7. Źródła	15

Streszczenie

Niniejsza publikacja stanowi przegląd informacji na temat zastosowania technologii Rich Internet Application (RIA) w bankowości elektronicznej na przykładzie Verax eBanking Suite.

RIA jest zbiorczym terminem określającym technologie umożliwiające tworzenie bogatych aplikacji działających w środowisku przeglądarki internetowej. Termin ten został ukuty w 2001 przez firmę Macromedia (obecnie Adobe). Wiodącymi technologiami RIA są Adobe Flex, Microsoft Silverlight oraz Sun JavaFX. RIA znajduje coraz szersze zastosowanie w Web 2.0, a lista serwisów zbudowanych w oparciu o te technologie obejmuje w sektorze bankowości m. in. Citicorp, Morgan Stanley, Deutsche Bank, State Street Bank, Wachovia czy Raiffeisen.

eBanking Suite™ jest platformą umożliwiającą szybkie stworzenie, łatwych do integracji aplikacji bankowości elektronicznej opartej o technologię Adobe Flex™. Verax eBanking Suite składa się z trzech aplikacji przeznaczonych dla banków i domów maklerskich dostarczających nowoczesny i zaawansowany graficzny interfejs użytkownika do obsługi:

- Bankowości internetowej dla klientów indywidualnych.
- Bankowości internetowej dla klientów korporacyjnych.
- Elektronicznego obrotu papierami wartościowymi.

Odbiorcy publikacji

Publikacja, którą Państwu prezentujemy, została przygotowana przez ekspertów i specjalistów Verax Systems. Jej głównym celem jest naświetlenie najważniejszych zagadnień związanych z zastosowaniem technologii RIA w bankowości elektronicznej oraz zaprezentowanie kluczowych informacji dotyczących oferowanych produktów i technologii.

1. Wprowadzenie

Obecna sytuacja na rynku bankowym w Polsce staje się coraz bardziej skomplikowana. Walka o indywidualnego Klienta jest jedną z najaktywniej prowadzonych przez instytucje finansowe działalności. Poza oczywistym elementem tej walki, jakim są oferowane produkty, istotną rolę odgrywa sposób obsługi konta przez indywidualnego klienta.

W dziedzinie aplikacji bankowości elektronicznej istnieje wiele wyzwań, z którymi zmagają się instytucje finansowe na całym świecie. Do najważniejszych z nich należą:

- Zapewnienie bezpieczeństwa.
- Ergonomia użytkownika oraz dodawanie nowych funkcjonalności (np. wizualizacja danych, video streaming, wsparcie dla klientów końcowych itp.).
- Możliwość rozwoju poprzez nowe funkcjonalności, przy jednoczesnym zachowaniu kompatybilności wstecznej (np. poprzez tzw. „interfejs klasyczny”).
- Atrakcyjność.
- Dostępność.
- Zgodność z różnymi środowiskami i systemami informatycznymi (np. przeglądarkami internetowymi).
- **Ochrona poniesionych inwestycji** w infrastrukturę IT – zwiększanie pojemności systemu (ang. *capacity*), rozumianej jako ilość jednocześnie obsługiwanych użytkowników na jednostkę mocy obliczeniowej) przy jednoczesnej minimalizacji nakładów na rozbudowę istniejącej infrastruktury IT (łącza internetowe, serwery, routery, bazy danych itp.).

Zapewnienie bezpieczeństwa jest niezwykle istotnym elementem bankowości, a bankowości elektronicznej w szczególności. Nie tylko jest ono istotne ze względu na działalność operacyjną instytucji finansowej ale również ma istotne znaczenie dla jej wizerunku.

Zwiększona **ergonomia użytkownika** pozwala osiągnąć przewagę konkurencyjną nad instytucjami oferującymi podobne usługi. Poza oczywistą ofertą produktów i usług istotne znaczenie dla klienta końcowego ma możliwość bezproblemowej obsługi. Ponadto zwiększona ergonomia użytkownika przekłada się na mniejsze koszty pomocy telefonicznej. Ergonomia użytkownika jest też nierozdzielnie związana z **atrakcyjnością** aplikacji.

Elementem, który ma duże znaczenie dla unikalności oferty instytucji finansowej jest jej rozwój. Funkcje aplikacji, które jeszcze niedawno były unikalne i innowacyjne, dziś już są tylko standardem. Istotne jest by odpowiednie technologie wykorzystane w bankowości elektronicznej pozwalały na elastyczny **rozwój**.

Fakt, iż ogromna większość oprogramowania bankowości elektronicznej użytkowana jest za pomocą przeglądarki internetowej, jest bezpośrednio związany z jej **dostępnością**. Prawdziwe wyzwanie stanowi zapewnienie dostępu do bankowości elektronicznej dla możliwie szerokiej rzeszy odbiorców, przy jednoczesnym utrzymaniu funkcjonalności aplikacji. Co więcej, dostępność rzadko idzie w parze ze **zgodnością** z różnymi środowiskami. Uniwersalna technologia pozwala zaoszczędzić mnóstwo sił i środków, które musiałyby zostać zużyte na utrzymanie różnych środowisk.

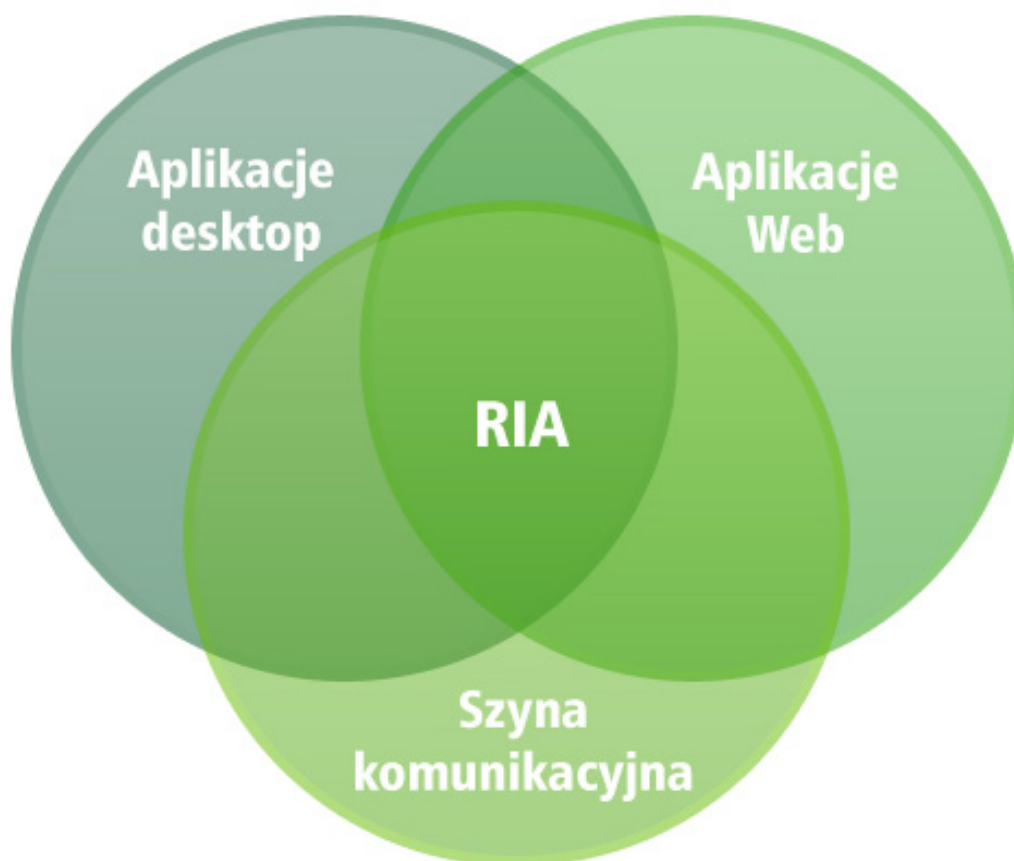
Wszystkie te czynniki składają się na zwiększenie **satysfakcji i lojalności** klienta końcowego.

2. RIA – nowe podejście do aplikacji internetowych

2.1. Opis technologii

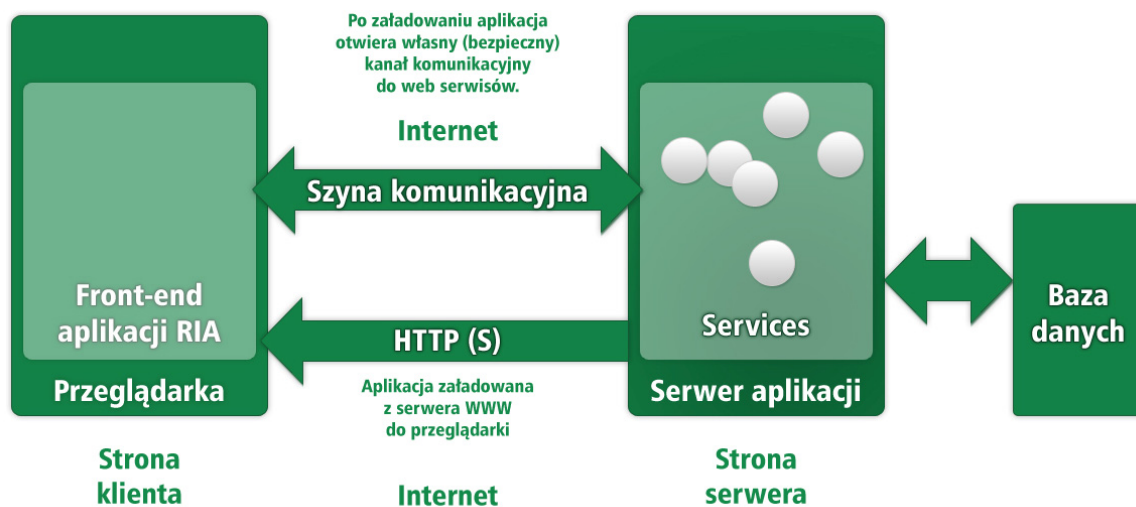
Mianem aplikacji RIA określa się aplikacje działające w środowisku przeglądarki internetowej, które posiadają cechy i zalety tradycyjnych aplikacji desktop. RIA są aplikacjami stanowymi (tj. w przeciwieństwie do klasycznych aplikacji webowych, posiadają własny stan i dane na stacji klienta) i kontaktują się usługami po stronie serwera (tzw. *back-end*), głównie w celu wymiany danych.

Tradycyjne aplikacje HTML zbudowane są w oparciu o model klient-serwer i koncepcję tzw. chudego klienta (ang. *thin client*), w której większość interakcji użytkownika wymaga zaangażowania serwera. Poprzez użycie aplikacji RIA, która może przechowywać własne dane i wykonywać instrukcje po stronie klienta, przenosząc obciążenie ze strony serwera na stronę klienta. W tym sensie, architektura RIA przypomina tradycyjne aplikacje desktop tzw. grubego klienta (ang. *fat client*).



Rysunek 1: RIA a inne technologie.

Dodatkowo, RIA ma wszelkie zalety klasycznych aplikacji webowych takie jak: łatwość uaktualnień z centralnego serwera, współpracę z *cache* przeglądarki itp.



Rysunek 2: Architektura aplikacji RIA.

2.2. Korzyści dla użytkownika końcowego

Niniejszy rozdział zawiera informacje o korzyściach wynikających z zastosowania RIA z punktu widzenia użytkownika aplikacji. W przypadku bankowości, użytkownikiem tym jest po prostu klient banku korzystający z portalu bankowości elektronicznej.

Korzyści związane z ergonomią użytkownika

Jak sama nazwa wskazuje, RIA umożliwia tworzenie „bogaty” aplikacji, czyli takich które charakteryzują się **wysoką estetyką interfejsu** użytkownika – począwszy od efektów kolorystycznych takich jak blur, czy też transparentność na animacjach skończywszy. Warto nadmienić, że jest to ważne nawet dla relatywnie prostych aplikacji. Firmy które stosują bogaty interfejs użytkownika, np. Apple (iPod, iPhone), Oracle (Metalink) czy Adobe (Portale) odniosły niebywały sukces marketingowy. Technologie RIA udostępniają programistom szereg efektów w zakresie animacji, co umożliwia **zdynamizowanie aplikacji**, tak by były one bardziej atrakcyjne dla użytkownika.

Skrócenie czasów odpowiedzi i zwiększenie doświadczenia użytkownika

Jednym z głównych problemów klasycznych aplikacji HTML jest to że działają one w oparciu o ładowanie kolejnych stron. W przypadku, gdy użytkownik chce obejrzeć nowe lub nawet te same dane w inny sposób (np. zmiana z wykresu słupkowego na kołowy) strona HTML musi być każdorazowo przeładowana (może to być po prostu następane zapytanie POST/GET lub ta sama operacja wywołana za pomocą AJAX).

Niezależnie od parametrów łącza internetowego, przeładowanie takie trwa od 1 do 5 sekund (oczywiście czas odpowiedzi samego serwera stanowi tylko część tego czasu, do tego dochodzi czas transmisji danych oraz ich rendering w przeglądarce). Warto w tym miejscu nadmienić, że satysfakcjonujący użytkowników czas odpowiedzi wynosi poniżej 1 sekundy – czasy powyżej tej wartości powodują odwrócenie uwagi użytkownika, a w efekcie niski wskaźnik doświadczenia użytkownika (ang. *user experience, UX*).

W przypadku prawidłowo zaprojektowanej aplikacji RIA czas odpowiedzi dla użytkownika może być znacząco skrócony, ponieważ sama aplikacja nie wymaga przeładowania (ładowane są tylko dane). Dodatkowo dalsze przyciągnięcie uwagi użytkownika można uzyskać za pomocą animacji, cachingu czy pre-fetchingu danych. Zalety takiego działania to:

- Zwiększenie satysfakcji klienta końcowego z aplikacji.
- Skrócenie czasów sesji klienta (zwiększenie produktywności użytkownika, a także bardziej płynne działanie).

Asynchroniczność

W modelu RIA dane mogą być przesyłane z serwera do klienta w sposób asynchroniczny, tj. dane mogą być wysyłane do klienta w trybie *push*. Oprócz niewątpliwych zalet technicznych (mniejsze obciążenie serwera, eliminacja odpytywania) daje to użytkownikowi poczucie lepszej kontroli nad aplikacją (nie muszą sprawdzać, co się dzieje, system sam mnie poinformuje) i zwiększa wygodę korzystania.

Dodatkowo za pomocą takich technik asynchronicznych jak pre-fetching, aplikacja może łączyć dane jednocześnie je wyświetlając (podobnie jak Google Maps ładuje w tle segmenty map sąsiednich do aktualnie oglądanego obszaru) dodatkowo zwiększając wydajność i doświadczenie użytkownika.

Produktywność

Ponieważ strony nie są przeładowywane, użytkownicy aplikacji RIA nie mają potrzeby "reorientacji" w nowym (choćby minimalnie zmodyfikowanym) układzie strony (warto zauważyć, że uwaga użytkowników jest też przerywana samym faktem „migotania” zawartości strony powodowanego przez jej rendering przez przeglądarkę). Umożliwia to ciągłe skupienie (ang. *focus*) użytkownika na realizowanych za pomocą aplikacji zadaniach, prowadząc do zwiększenia produktywności jego działania. Każde takie odwrócenie uwagi (ang. *attention gap*) użytkownika może powodować jego zachowanie takie jak rozpoczęcie innego zadania (np. uruchomienie aplikacji, rozpoczęcie oglądania innych stron WWW itp.), co w praktyce może powodować znaczące koszty (np. konieczność równoległej obsługi większej liczby sesji).

Integracja z mediami i Web 2.0

Technologie RIA umożliwiają pełną integrację treści multimedialnej (dźwięk, video) wewnątrz aplikacji. Treści multimedialne są przydatne nie tylko dla aplikacji typowo rozrywkowych, lecz także w „poważnych” aplikacjach biznesowych. Wybrane przykłady to:

- Systemy obsługi klienta (ang. *customer care*) korzystające z wbudowanego mikrofonu, głośników i kamery internetowej w celu połączenia się z konsultantem.
- Filmy o produkcie wyświetlane w katalogu produktów dostępnych dla klientów.
- Video-podręczniki i tutoriale.

Dodatkową zaletą RIA jest to, że treść multimedialna może współpracować z aplikacją – np. w odpowiednim momencie wyświetlić film z instrukcją, czy podświetlić odpowiednie pola formularza w zależności od wyświetlanej lub wygłaszanej instrukcji.

Kierowanie użytkownikiem (*user guidance*)

Jedną z fundamentalnych różnic w projektowaniu aplikacji RIA w porównaniu z klasycznymi aplikacjami HTML, jest fakt, że aplikacje RIA są bardziej zorientowane na "stan" a nie na "stronę". W modelu takim przejścia pomiędzy stanami są wywoływane przez poszczególne akcje użytkownika. W przypadku bankowości elektronicznej może to być:

1. Wybór operacji (np. przelew do ZUS).
2. Wybór parametrów operacji (np. typ wpłaty).
3. Wybór przelewanej kwoty.
4. Zatwierdzenie.
5. Zapytanie o potwierdzenie przelewu.

Każde przejście pomiędzy stanami może być w przypadku aplikacji RIA wspomagane technikami kierowania użytkownikiem takimi jak:

- Informacje nt. wymaganych pól (np. w postaci tooltipów, walidatorów, podświetlania, itp.).
- Animowane przejścia wizualizujące sens przejścia z jednego stanu do drugiego.

Dobrze zaprojektowana "choreografia" przejść umożliwi użytkownikom **sprawniejsze i łatwiejsze** realizowanie zadań.

Lokalność danych

Aplikacje Web, które wymieniają, przetwarzają oraz wizualizują duże ilości danych, mogą zostać znacząco zoptymalizowane poprzez zastosowanie architektury RIA. W przypadku klasycznej aplikacji każda zmiana zestawu danych wejściowych (np. przez zastosowanie filtra, zmianę porządku sortowania czy przewinięcie strony) powoduje interakcję z serwerem, co ma negatywny wpływ na produktywność. Aplikacja RIA może korzystać z danych lokalnych oraz przetwarzać je (w miarę możliwości) bezpośrednio na kliencie. Daje to efekt „lokalności danych”, który prowadzi do zwiększenia satysfakcji użytkownika korzystającego z aplikacji.

2.3. Korzyści techniczne

Niniejszy rozdział zawiera informacje o korzyściach wynikających z zastosowania RIA z punktu widzenia dystrybutora aplikacji.

Niezależność od przeglądarki i systemu operacyjnego

W przypadku Adobe Flex, technologia ta oferuje faktyczną niezależność od zastosowanej przeglądarki internetowej i systemu operacyjnego (aplikacje Flex używają Macromedia Flash Player, który jest dojrzałą technologią – trudno sobie wyobrazić portal nie stosujący Flash). Flex eliminuje problemy kompatybilności renderingu HTML oraz JavaScript pomiędzy przeglądarkami. Z powodu dużej ilości przeglądarek dostępnych na rynku (Internet Explorer, Firefox, Google Chrome, Apple Safari, Opera) oraz znaczących różnic działania pomiędzy ich wersjami (np. IE 6 a IE 7) eliminacja tego problemu, daje wymierne oszczędności na etapie testowania, a także zmniejsza prawdopodobieństwo potencjalnej niekompatybilności (np. z nowymi wersjami przeglądarek).

Łatwe uaktualnienia i dystrybucja

Aplikacje RIA są typowo dystrybuowane i ściągane w postaci pojedynczego pliku (w przypadku Adobe Flex jest to plik SWF). Eliminuje to problem z cachingiem JavaScript'u a także *de facto* zmniejsza pasmo potrzebne do korzystania z aplikacji – po pierwszym załadowaniu aplikacji pasmo potrzebne jest jedynie do wymiany danych.

Niezależność od klienta (GUI) od serwera

W przypadku technologii RIA, łatwiej niż w klasycznej aplikacji HTML wprowadzać zmiany w interfejsie użytkownika oraz przeprowadzić deployment samego front-endu bez wyłączania/modyfikacji po stronie serwera – wystarczy jedynie na serwerze WWW podmienić plik aplikacji (SWF). Fakt ten jest nie do przecenienia w przypadku bankowości elektronicznej, wymagającej dostępności w reżimie 24/7.

Powszechnie dostępny run time

Adobe Flash dostępny jest na zdecydowanej większości komputerów podłączonych do Internetu (wg Adobe wartość ta sięga 99%). Dodatkowo, instalacja runtime Flash Player przebiega dla większości przeglądarek w pełni automatycznie. Automatycznie można także wymusić uaktualnienie do konkretnej minimalnej wersji runtime. Sam Flash player zajmuje jedynie około 1,8 MB. Dla porównania środowisko uruchomieniowe JavaFx wymaga ponad 15MB, a Microsoft Silverlight 4,6MB.

Niższe obciążenie sieci

W przypadku aplikacji RIA, niższe obciążenie sieci można osiągnąć nie tylko przez sam fakt, że pomiędzy klientem a serwerem wymieniane są jedynie dane, a sama aplikacja jest ładowana jednokrotnie. Dodatkowo poważne oszczędności w ilości przesyłanych danych można osiągnąć poprzez bardziej inteligentne zaprojektowanie samej aplikacji i usług po stronie serwera, tak by pobierała dane na żądanie (tj. wtedy gdy są one potrzebne do faktycznego wyświetlenia).

Większa wydajność systemu z jednostki mocy obliczeniowej

Ponieważ systemy w architekturze RIA wykorzystują do renderingu zasoby procesora komputera klienta, a dodatkowo minimalizują ilość komunikacji z serwerem, prowadzi to znacząco do obciążenia serwera. W jednym z projektów realizowanych przez Verax Systems, przeniesienie aplikacji z klasycznej technologii HTML/AJAX/JSP na platformę Adobe Flex umożliwiło obsługę dwukrotnie większej liczby równoległych użytkowników bez wymiany sprzętu (była to platforma IBM p550 wraz z bazą danych Oracle 10i).

W związku z faktem, że współczesne komputery desktop są bardzo wydajne, daje to możliwość wykorzystania „zaoszczędzonych” cykli CPU do implementacji skomplikowanych obliczeniowo funkcjonalności takich jak np. zaawansowane algorytmy szyfrowania.

Minimalne wymagania dla komputera użytkownika potrzebne do uruchomienia aplikacji w technologii Adobe Flex™ są tożsame z wymaganiami dla aplikacji Adobe Flash Player™, czyli jest to procesor klasy Pentium 450 MHz oraz 128 MB pamięci RAM.

2.4. Korzyści dla programistów

Na zakończenie tego rozdziału, warto wspomnieć o czysto biznesowych korzyściach dla organizacji, która decyduje się rozwijać swoje aplikacje czy usługi w oparciu o technologie RIA. W Verax Systems wybraliśmy Adobe Flex jako podstawową technologię tworzenia aplikacji także z racji korzyści dla zespołu programistycznego do których zaliczyć można:

- Szybkie, graficzne projektowanie interfejsu użytkownika za pomocą narzędzia Flex Builder bez konieczności budowania aplikacji, znacząco skracające czas projektowania graficznych elementów aplikacji.
- Ścisłą separację usług od interfejsu użytkownika, co narzuca pewien reżim programistyczny i powoduje, że kod jest łatwiejszy do zrozumienia, a co za tym idzie tańszy w utrzymaniu.
- Gotowa, wydajna, asynchroniczna i dwukierunkowa szyna komunikacyjna pomiędzy klientem i serwerem (LiveCycle Data Services – LCDS).
- Bogata dostępna biblioteka komponentów stron trzecich (np. Yahoo! Maps) oraz efektów graficznych.

Według wewnętrznych szacunków Verax Systems, czas tworzenia aplikacji RIA w porównaniu z aplikacjami opartymi o HTML/AJAX/JavaScript został skrócony ok. dwukrotnie. Dodatkowo obserwujemy ok. 30% spadek nakładów poświęconych na utrzymanie systemów.

3. Verax eBanking Suite

3.1. Informacje o produkcie

Verax eBanking Suite składa się z trzech aplikacji przeznaczonych dla banków i domów maklerskich dostarczających nowoczesny i zaawansowany graficzny interfejs użytkownika do obsługi:

- Bankowości internetowej dla klientów indywidualnych.
- Bankowości internetowej dla klientów korporacyjnych.
- Elektronicznego obrotu papierami wartościowymi.

Dodatkowym elementem platformy jest aplikacja administracyjna przeznaczona do kontroli pakietu. Funkcje, które podlegają kontroli to m.in. wygląd aplikacji, dostęp do poszczególnych jej elementów dla różnych użytkowników, zarządzanie treścią reklamową.

Cała platforma jest skonstruowana tak, by wspierać architekturę SOA. Każda z aplikacji używa określonego zestawu usług (ang. services), które wykonują odpowiednią funkcjonalność i zwracają dane do aplikacji klienckiej.

3.2. Przegląd funkcjonalności

Aplikacje te składają się z graficznego interfejsu użytkownika zbudowanego w oparciu o technologie Adobe Flex/AIR i mogą być uruchamiane w przeglądarce internetowej lub jako aplikacje desktop z komputera stacjonarnego. Każda aplikacja posiada odpowiadający jej toolkit umożliwiający sprawną integrację z istniejącymi systemami back-end lub oprogramowaniem middleware.

Platforma eBanking Suite oferuje szereg funkcjonalności, które można w całości wykorzystać do tworzenia gotowej aplikacji bankowości elektronicznej. Do najważniejszych z nich należą:

- Logowanie.
- Interaktywna pomoc.
- Wizualizacja danych historycznych.
- Eksport danych do formatu PDF.
- Integracja z formularzami PDF (użytkownik może wypełniać formularze PDF on-line, bezpośrednio w aplikacji bez konieczności instalowania Adobe Acrobat).
- Filtrowanie danych.

3.3. Verax Banking Suite – bezpieczeństwo

W modelu aplikacji wykorzystywanym w eBanking Suite komunikacja odbywa się pomiędzy aplikacją uruchomioną w przeglądarce a serwerem usług. Mechanizm ten wspiera kilka metod dostępu do danych, które można konfigurować z różnymi poziomami bezpieczeństwa. Poza komunikacją po zwykłym protokole http możliwe jest wykorzystywanie technologii SOAP (Web Service) oraz specjalnego protokołu AMF.

Dodatkowo komunikacja ta może być szyfrowana protokołem SSL niezależnie od zabezpieczeń protokołu wykorzystywanego przez przeglądarkę. Technologia Adobe Flex wykorzystuje mechanizm sesji HTTP do zapewnienia identyfikacji użytkownika.

3.4. Verax Banking – integracja

Architektura platformy eBanking Suite jest zorientowana na usługi (SOA). Takie podejście pozwala na integrację aplikacji z systemem bankowym w warstwie usług. Platforma eBanking Suite wykorzystuje określony interfejs w postaci szeregu usług odpowiedzialnych za poszczególne funkcjonalności w aplikacji.

Integracja eBanking Suite z systemem bankowym polega na podłączeniu interfejsu udostępnianego przez system bankowy do wywołań usług platformy. W praktyce często usługi wymagane przez eBanking Suite są używane przez istniejące systemy bankowości elektronicznej i wystarczy je wykorzystać. eBanking Suite łatwo integruje się z usługami w języku Java oraz z usługami Web Service.

W typowym przypadku integrację rozpoczyna się od określenia podstawowych wymagań. Z reguły – w pierwszym etapie – są one tożsame z wymaganiami dla aktualnie funkcjonującej aplikacji bankowości elektronicznej. W konsekwencji – jeśli architektura używanej aplikacji jest warstwowa – wykorzystuje się warstwę usług i podłącza bezpośrednio do eBanking Suite. Podłączone usługi mogą być wykorzystane w aplikacji Flex bez dodatkowych operacji – komunikacja staje się przezroczysta z punktu widzenia programisty.

W kolejnym kroku zestawiane są menu, widoki i zawartość strony zgodnie z wymaganiami, używając gotowych komponentów eBanking Suite oraz wprowadzając do nich niezbędne modyfikacje. Na tym etapie wprowadzane są również funkcjonalności, które wynikają z możliwości Adobe Flex™, a nie wymagają zmiany usług aplikacji, np. wizualizacja danych, interaktywne filtry, etc.

W ostatnim etapie – jeśli są takowe wymagania – aplikacja wyposażona jest w dodatkowe funkcjonalności, które wymagają zmian na poziomie usług systemu bankowego. Zmiany te, ze swej natury, mogą mieć wpływ na aktualnie wykorzystywaną aplikację, dlatego ich realizacja wymaga dodatkowego planowania.

3.5. Verax Banking – ładowanie oraz zależność od pasma

Uruchomienie aplikacji zbudowanej w oparciu o eBanking Suite polega na załadowaniu przez przeglądarkę pliku aplikacji i uruchomienie go. Rozmiar pliku aplikacji jest stosunkowo niewielki i może być załadowany w ciągu kilku sekund na średniej jakości łączu internetowym. Sam proces ładowania nie pozostawia użytkownika z pustym ekranem – odpowiednia informacja wraz z paskiem postępu jest wyświetlana na ekranie.

Ponadto, aplikacja jest zachowywana w pamięci *cache* przeglądarki i przy ponownym uruchomieniu strony proces ładowania aplikacji jest pominięty.

Początkowe czasy ładowania aplikacji eBanking Suite zależą wyłącznie od rozmiaru pliku aplikacji, który pobierany jest na startcie. Niezależnie od końcowego rozmiaru aplikacji (który waha się od kilkuset kilobajtów do kilku megabajtów, głównie w zależności od ilości ładowanych grafik) Adobe Flex™ umożliwia podział aplikacji na moduły. W efekcie użytkownik musi pobrać z Internetu około 800kB – 1MB danych, by uruchomić aplikację. Na łączu o przepustowości 256kbit/s zajmie to około 30 sekund. Każde kolejne uruchomienie wykorzystuje aplikację pobraną do pamięci *cache* przeglądarki.

Gdy aplikacja jest już uruchomiona czasy ładowania danych są znacznie niższe niż w przypadku technik opartych o HTML. Tę tezę potwierdzają liczne testy, np.:

<http://www.jamesward.com/blog/2007/04/30/ajax-and-flex-data-loading-benchmarks/>

Verax eBanking zawiera możliwość realizacji mechanizmu SSO (*single sign-on*), co umożliwia wykorzystywanie aplikacji eBanking Suite **równolegle** z interfejsem klasycznym.

3.6. Verax Banking – skalowalność

Ze względu na architekturę eBanking Suite duża ilość funkcjonalności zostaje przeniesiona na komputer użytkownika aplikacji. Czynności takie jak rysowanie wykresów i grafów, konwersje danych, filtrowanie, etc. są wykonywane przy użyciu procesora klienta. W dużej skali pozwala to dość znacznie zredukować obciążenie serwerów. Wstępne eksperymenty wykazują, że zwiększenie pojemności (ang. *capacity*) systemu (rozumianej jako ilość jednocześnie obsługiwanych sesji na daną moc obliczeniową) jest około dwukrotnie większe.

Nie bez znaczenia pozostaje również fakt, że eBanking Suite – w przeciwieństwie do aplikacji opartych o HTML – potrzebuje typowo mniejszego rozmiaru sesji, co przekłada się bezpośrednio na ilość zajętej pamięci RAM w serwerach aplikacji.

3.7. Verax Banking Suite – od strony użytkownika

Do uruchomienia aplikacji w przeglądarce użytkownika wymagana jest przeglądarka z zainstalowanym odtwarzaczem Flash w wersji co najmniej 9. Badania przeprowadzone przez firmę Adobe wskazują, że Flash Player w wymaganej wersji jest obecny na 99% systemów (informacje na temat źródła zebrane są w punkcie 7). Ponadto przy uruchomieniu aplikacji wykrywana jest wersja odtwarzacza. Jeśli jest za niska odbywa się automatyczna aktualizacja (około 1,8 MB danych do ściągnięcia).

Użycie Adobe Flash Player do uruchomienia aplikacji przynosi bardzo dużą korzyść dla rozwoju aplikacji. Mianowicie nie jest istotny rodzaj i wersja przeglądarki internetowej. Aplikacja w każdej z nich wygląda tak samo, co znacznie redukuje ilość czasu przy rozwoju oprogramowania.

Uruchomienie aplikacji odbywa się przez otwarcie określonego adresu internetowego. W tym momencie z serwera pobierany jest plik, który zawiera aplikację i aplikacja jest uruchamiana. Pobrany plik jest przechowywany w cache przeglądarki, więc przy ponownym wybraniu adresu nie musi być ponownie pobierany.

W trakcie działania aplikacji specyfika eBanking Suite pozwala na pełną kontrolę nad poborem i przechowywaniem danych. W odróżnieniu od aplikacji opartych o HTML nie ma konieczności poboru danych w modelu zapytanie – odpowiedź. Dane mogą być pobierane w tle – można przewidzieć działanie użytkownika i dostarczyć dane do aplikacji zanim ich zażąda. Ponadto nie ma konieczności przetrzymywania danych w sesji HTML – dane przetrzymywane są w pamięci uruchomionej aplikacji. W konsekwencji nie ma ścisłego ograniczenia na wielkość aktywnych danych przechowywanych w kontekście aplikacji. Ponadto model odświeżania informacji jest w pełni kontrolowany przez aplikację. Przykładowo, nie ma konieczności odświeżania wartości stanu kont za każdym razem, gdy użytkownik przejdzie do strony głównej aplikacji (lub wykorzystywania mechanizmu zaawansowanego zarządzania sesją) – można ustalić określony interwał, kiedy dane są pobierane. Wspólnie, mechanizmy opisane powyżej, pozwalają na lepsze wykorzystanie usług dostarczających dane. Model wykorzystany w eBanking Suite pozwala na oszczędności w zasobach serwerów w postaci rozmiarów sesji użytkownika oraz liczbie wywołań usług.

Z właściwości modelu przesyłania danych, a także ze specyfiki samego Flash Player'a wynikają znaczne możliwości podniesienia jakości aplikacji z punktu widzenia użytkownika. Fakt, że aplikacja sama zajmuje się renderowaniem swojej zawartości powoduje, że dane dostarczane z serwera mogą być w postaci surowej. Aplikacja zajmuje się wtedy wyświetlaniem, ewentualnym przetworzeniem i wizualizacją danych. W ten sposób można rozszerzyć typowe funkcjonalności aplikacji bankowości elektronicznej o zaawansowaną wizualizację (np. interaktywne wykresy historii wydatków lub struktury posiadanych aktywów i pasywów).

Jakość aplikacji może też być podniesiona przez znacznie większe możliwości interfejsu graficznego. Przykłady takich ulepszeń można mnożyć:

- Wszelkie operacje pokazywania/chowania, przeciągania (typowo realizowane przy pomocy JavaScript) mogą być całkowicie płynne – użytkownik nie traci kontekstu i zawsze wie, co się stało na ekranie.
- Możliwe jest wyświetlanie w pełni funkcjonalnych okienek pop-up łącznie z blokowaniem akcji użytkownika (okna modalne).
- Możliwe jest pełniejsze wykorzystanie akcji użytkownika, np. przeciąganie symboli kont lub zdefiniowanych odbiorców, by zainicjować przelew.
- Łatwe wyświetlanie animowanych drzew.
- Interaktywna (w tym także multimedialna) pomoc.
- rozbudowane wsparcie dla mediów (muzyka, film).

Poza elementami związanymi bezpośrednio z reprezentacją informacji na ekranie, istotnym elementem jest możliwość wykorzystania w eBanking Suite formularzy PDF. Taka możliwość wydaje się bardzo ciekawa przy funkcjonalności składania różnorodnych wniosków. Po wprowadzeniu danych przez użytkownika generowany jest odpowiednio zabezpieczony dokument PDF i odpowiednio dystrybuowany zarówno wewnątrz instytucji, jak i dla klienta.

3.8. Wielojęzyczność

eBanking Suite jest dostarczany w wersji wielojęzycznej. Nie jest wymagana żadna dodatkowa praca programistyczna by zapewnić dodatkowe języki w aplikacjach. W każdym momencie działania aplikacji można przełączyć się między językami bez utraty kontekstu i jakiegokolwiek przeładowania strony – żadna z wykonywanych operacji nie zostaje przerwana.

4. Model biznesowy

Wybór modelu biznesowego dla rozwiązań takich jak eBanking Suite™ jest trudny. Dodatkowo bankowość i finanse jest branżą o bardzo specyficznych wymaganiach. Niektóre z wyzwań dla dostawcy obejmują następujące kwestie:

- Jak zapewnić **integrację** z systemami transakcyjnymi banku w taki sposób by mógł to robić zarówno dostawca jak i działy rozwoju aplikacji Klienta?
- W jaki sposób zapewnić bankowi możliwość **szybkiej reakcji** na ewentualnie zmiany w systemach back-end, jeżeli szybko nie może tego zrobić Dostawca?
- W jaki sposób uelastyczyć **wsparcie techniczne** wymagające dostępu do kodów źródłowych?
- W jaki sposób umożliwić bankowi **samodzielny rozwój** systemu wynikający z nowo oferowanych usług tak by zapewnić wydajność, spójność oraz zgodność nowych funkcjonalności z architekturą systemu?
- Jak zabezpieczyć dostęp do kodów źródłowych?
- Jak rozwiązywać sytuacje, w przypadku, gdy **Prawo Bankowe**, praktycznie zabrania dostępu do rzeczywistych danych o klientach pracownikom dostawcy?
- W jaki sposób **elastycznie licencjonować** system?

Dla eBanking Suite przyjęliśmy model zwany „Beekeeper Model”, przypominający ideą zamknięty open source. Model ten można streścić w następujących punktach:

- Verax Systems przekazuje Klientowi kod źródłowy całego systemu oraz development kit.
- Kod jest chroniony licencją, która zapewnia własność kodu Verax Systems.
- Opłata licencyjna jest miesięczna związana z ilością użytkowników (nie ma stałej kwoty licencyjnej).

- Klient dostaje wszelkie zmiany/uaktualnienia.
- Zmiany wprowadzane przez Klienta, konsultowane są przez architektów Verax Systems.

W takim przypadku, klient Verax Systems ma elastyczność i pełne bezpieczeństwo związaną z kwestiami integracji, wsparcia technicznego i innych wymienionych na początku tej sekcji.

5. Przyszłość

5.1. Prognozy

Zgonie z prognozami dużych firm doradczych IT takich jak Gartner, czy ZapThink, udział RIA w rynku aplikacji będzie się zwiększał. Powodem jest nie tylko ładny wygląd takich aplikacji, ale także wymierna korzyść, jaką przynoszą one organizacji.

Szacuje się, że do roku 2010 około 60% nowych projektów będzie wykorzystywało technologię RIA, przy czym w 2011 roku rynek RIA osiągnie wartość 3 miliardów dolarów.

5.2. Adobe AIR™

Obecnie jednym z nieodzownych elementów bankowości elektronicznej jest przeglądarka internetowa. Ze względu na problemy związane zwykle z instalacją tzw. „grubego klienta” pomysł dostępu do konta bankowego poprzez przeglądarkę internetową, która jest zainstalowana na każdym dzisiejszym komputerze osobistym, jest bardzo dobry.

Znikają problemy wynikające z konieczności instalacji oprogramowania. Problem różnych środowisk i systemów operacyjnych zostaje ograniczony w zasadzie do wersji przeglądarki. Dodatkowo problem aktualizacji oprogramowania w praktyce nie istnieje, ponieważ użytkownik przy każdym użyciu przeglądarki otrzymuje na ekran wynik działania najnowszej wersji oprogramowania.

Z drugiej jednak strony wykorzystanie przeglądarki nakłada określone ograniczenia w postaci braku możliwości wykorzystania zasobów lokalnych komputera – dostępu do plików, ikony powiadomień, dedykowanego okna z aplikacją etc.

Naprzeciw temu zagadnieniu wychodzi technologia Adobe Integrated Runtime™ – w skrócie AIR™. AIR™ jest rozszerzeniem, które pozwala zbudować aplikację do instalacji na podstawie aplikacji RIA – w tym wypadku aplikacji eBanking Suite.

Technologia AIR™ łączy zalety przeglądarki internetowej z możliwościami aplikacji instalacyjnej. Gdy użytkownik zechce zainstalować aplikację eBanking Suite wyposażoną w technologię AIR™ uruchamia odpowiedni plik, który instaluje aplikację na komputerze wraz z wszystkimi elementami, jak pliki, menu, skrót na pulpicie, etc. Instalacja taka może być przeprowadzona na systemach operacyjnych Windows, Linux oraz MacOS.

Przy uruchomieniu aplikacji zostaje wyświetlone okno, które współpracuje z serwerem instytucji finansowej tak, jak aplikacja w przeglądarce, jednak środowisko AIR pozwala wykorzystać lokalne zasoby komputera, takie jak ikona w zasobniku systemowym, czy inne zasoby systemowe.

Ewentualna aktualizacja aplikacji odbywa się automatycznie. Gdy tylko nowa wersja aplikacji pojawi się na serwerze, użytkownik jest o tym informowany i instalacja przebiega automatycznie. W konsekwencji użytkownik ma dostęp do banku bez konieczności otwierania przeglądarki oraz wyeliminowane zostaje ładowanie strony/aplikacji z Internetu.

Adobe AIR™ wydaje się być technologią, która doskonale spełnia rolę uzupełnienia funkcjonalności aplikacji bankowości elektronicznej dla wymagających użytkowników. Przy czym cykl działania oprogramowania ogranicza konieczność aktywności użytkownika przy utrzymaniu oprogramowania do minimum.

6. Podsumowanie

Verax Systems ma doświadczenia w bankowości jak i jedne z najdłuższych doświadczeń w Polsce w RIA (od 2005 roku). Z racji, że systemy bankowe poszczególnych instytucji finansowych znacznie różnią się od siebie, trudno jest przedstawić w jednoznaczny sposób możliwość wykorzystania eBanking Suite dla każdego zastosowania. Dlatego też zachęcamy do kontaktu z zespołem Verax Systems. Nasi konsultanci służą pomocą i są gotowi zaprezentować i wyjaśnić wszystkie aspekty RIA oraz eBanking Suite. Zachęcamy również do częstego odwiedzenia naszych stron internetowych www.veraxsystems.com.

7. Źródła

- Strona główna produktu Adobe Flex:
<http://www.adobe.com/products/flex/>
- Portfel aplikacji Adobe Flex:
<http://flex.org/showcase/>
- Porównanie czasów ładowania aplikacji w różnych technologiach:
<http://www.jamesward.com/blog/2007/04/30/ajax-and-flex-data-loading-benchmarks/>
- Dane dotyczące powszechności Adobe Flash Player:
http://www.adobe.com/products/player_census/flashplayer/
- Wymagania systemowe dla uruchomienia aplikacji w technologii Adobe Flex™:
<http://www.adobe.com/products/flashplayer/systemreqs/>
- Szacunki dotyczące RIA:
http://findarticles.com/p/articles/mi_m0EIN/is_2006_July_25/ai_n26935125/pg_1?tag=artBody;col1
<http://www.adtmag.com/article.aspx?id=17953>