

Adobe AIR

Web Application Development -seminaari

24.10.2007

Tuomas Turto, Panu Åkerman

Sisältö

- Esittely
- Vertailua muihin
- Komponentit, arkkitehtuuri
- Turvallisuusmalli
- Sovelluskehitys ja työkalut
- Jakelu, asennus

- Demo-sessio
- Ajatuksia, mielipiteitä ja kysymyksiä

Esittely

- Aikaisempi nimi Adobe Apollo
- AIR=Adobe Integrated Runtime
- “Cross-operating system platform to build and deploy Rich-Internet Applications to the desktop using existing web development technologies (Flash, Flex, HTML, JavaScript, AJAX).”
- Tämän hetkinen versio beta 2

Esittely

- “Cross-operating system”
 - Tällä hetkellä saatavana Windowsiin ja Mac OS Xään. Linux suunnitelmassa, mutta ei tule 1.0 versioon. Käytännössä Adobe odottaa täyden Flash core -tuen valmistumista Linuxille.
 - Mobiililaitetuki on “roadmapissä”(mitä ikinä se tarkoittaakaan).
- “runtime”
 - Oma ajoympäristönsä eikä tukeudu selaimen. Virtuaalikone(et) ja kirjastot.
- “to build”
 - Adobe AIR käsittää myös SDK:n ja perustyökalut. Määrittelee käytettävät ohjelmointikielet, sekä vähän prosessia ja kirjastoja.

Esittely

- “and deploy”
 - Määrittelee ja toteuttaa jakelu- ja asennusmekanismin sovellusten jakeluun webissä.
 - Tästä myöhemmin lisää.
- “Rich Internet Applications”
 - Adobe AIR korostaa sovelluksia, jotka käyttävät webiä ja palveluita siellä, vaikkakaan mikään ei estä tekemästä muunlaisia sovelluksia. Kirjastot ja teknologiat tukevat RIA-lähtökohtaa. Web-jakelu.

Esittely

- “to the desktop”
 - Kyllä, ei selainta! Natiiveja sovelluksia työpöydälle.
- “using existing web development technologies (Flash, Flex, HTML, JavaScript, AJAX).”
 - Ei uutta ohjelmointikieltä, vaan kokoelma olemassa olevia kieliä, työkaluja ja metodeja. Määrittelee jollain tasolla yhteentoimivuuden näiden tekniikoiden välillä.
 - Adobe korostaa itse tätä: ei tarvetta opetella uusia kieliä/työkaluja, vaan kehittäjä voi käyttää osaamaansa tuttua ympäristöä.
 - Adoben mukaan kehitys voi pohjautua mihin tahansa edellä mainittuun tekniikkaan.

Vertailua

- Ei selaimen hiekkalaatikkoa. Tulos on natiivi työpöytäsovellus, jota ajetaan samoilla oikeuksilla kuin mitä tahansa natiivia sovellusta.
- Työpöytäintegraatio tulee mukana, eli hyvä tuki ominaisuuksille kuten drag-and-drop, system tray ja niin edelleen.
- Offline-käyttöä korostetaan (kuten nykyään tekee myös Google)
- Flash/Flex

Vertailua

- Useita ohjelmointikielivaihtoehtoja, joita ajetaan samassa ajoympäristössä (virtuaalikoneet taitavat olla erillisiä?). Eli vähän samaa kuin MS .NET/Silverlight.
- Määrittelee myös sovelluksen asennus- ja jakeluprosessin.
- Työkalukeskeinen (samaa kuin Silverlight:ssa?)
- Varsin omaperäinen turvallisuusmalli, mikä johtuu varmasti osaksi siitä, että ei ole käytössä selaimen hiekkalaatikkoa.

Arkkitehtuuri, komponentit

- AIR on
 - Adobe AIR runtime
 - Adobe AIR SDK
 - Työkalut
- Tärkeimmät ajoympäristön komponentit
 - Virtuaalikoneet Tamarin-projektista
 - ActionScript (Flash runtime), JavaScript
 - Open source
 - ActionScript suoritetaan käännettynä tavukoodina, JavaScript tulkitaan
 - Webkit HTML engine
 - Open source. Käytetään esittämään web-sisältöä, ja web-ajoympäristönä
 - Käytetään myös Safarissa.
 - SQLite tulee mukana

Arkkitehtuuri, komponentit

- Mielenkiintoisia komponentteja kehittäjälle
 - HTMLControl
 - Käytetään peruskomponenttina web-sisällölle ja sen ajamiselle kaikissa tekniikoissa. Ajaa JavaScriptin ja renderöi HTML:n. Tarjoaa pääsyn JavaScriptissä määriteltyihin olioihin. Natiivissa ikkunassa UI-komponentti.
 - WindowContainers
 - “Application container” Flash/Flex-puolella (ja loppujen lopuksi myös JavaScriptille).
 - ServiceListeners
 - Helpottaa offline-toiminnallisuutta.

AIR-turvallisuusmalli

- Evoluutio Flashin mallista
- Sovellukset oikeudet laajemmat kuin selaimessa
- Virtuaalikonepohjainen eli muistin käyttöä hallitsee ajoympäristö
- Asennustoiminnallisuus on osa turvallisuusmallia

AIR-turvallisuusmalli

- Ajonaikaiset oikeudet määritellään tiedostokohtaisesti, perustuen niiden alkuperäiseen lähteeseen. Tiedostot jaetaan turvallisuusryhmiin, joita kutsutaan hiekkalaatikoiksi (sandboxes).
- Sandboxit
 - Applications
 - Tiedostot sovelluksen omassa hakemistorakenteessa, kaikki oikeudet
 - Remote
 - Local-trusted
 - Local-with-networking
 - Local-with-filesystem
- Sandboxit (ja tiedostot niissä) on oletusarvoisesti erotettu toisistaan.
- Koodin dynaaminen evaluointi on rajoitettua.

AIR-turvallisuusmalli

- Yhteistoiminta application- ja non-application- hiekkalaatikoiden välillä
 - Sandbox bridge, toimii gateway-komponenttina
 - Tarkoitus on, että kehittäjä voi eksplisiittisesti päättää, mitä toiminnallisuutta tarjotaan hiekkalaatikosta ulospäin, ja tarjota toiminnallisuus ilman että objekti-referenssejä välitetään.

Kehityksen raamit

- Mahdollisia lähestymistapoja on useita
 - HTML & AJAX
 - Flash
 - Flex
- Kaikissa lähestymistavoissa on paljon samoja ohjelmistoteknisiä osia
- Työkaluilla on suuri merkitys

Lähestymistapojen tekniikat

HTML & AJAX	Flash	Flex
<ol style="list-style-type: none">1. (X)HTML, CSS jne.2. Javascript3. AJAX-kirjastot	<ol style="list-style-type: none">1. Flash-animaatiot2. ActionScript 3.0	<ol style="list-style-type: none">1. MXML2. ActionScript 3.03. Flex-luokkakirjasto

- Tekniikat kohderyhmän mukaan
- Yhteistä
 - ECMAScriptin / Javascriptin käyttö
 - Flash-tavukoodi
- Flex on uusin ja mielenkiintoisin

Arkkitehtuuri



Lähestymistapojen työkalut

HTML & AJAX	Flash	Flex
<ul style="list-style-type: none">Dreamweaver CS3	<ul style="list-style-type: none">Flash CS3	<ul style="list-style-type: none">Flex Builder

- Jokaiselle tavalle sitä tukeva Adobe-tuote
- AIR-paketointi jokaisen kohdalla
lisäominaisuus

ActionScript 3

- ECMA-262-standardin 4. version toteutus
- Javascript-kieli höystettynä mm.
 - luokilla
 - nimiavaruuksilla
 - staattisella tyyppityksellä

Ohjelmien jako

- AIR-ratkaisussa ohjelmat asennetaan paikallisesti
- Ajoalusta asennetaan ensimmäisen AIR-sovelluksen yhteydessä
- AIR-sovellukset allekirjoitetaan digitaalisesti

Demot ja Esimerkki

- Työpöytäintegraatio
- Verkkopalvelujen käyttö
- Itse tehty ristinolla
 - Adobe Flex Builder
 - ActionScript
 - Paketointi

Mielipiteitä

- Uusi Java?
 - Mikä on “iso ero” Java Webstartiin. Mainoskalvojen benefit-listat ovat hyvin samanlaiset.
- Kohdeyleisö enemmänkin graafiset suunnittelijat?
- Onko tämä sittenkin vain Flex-ympäristö, ja muut häviävät aikanaan
 - Tekniikoiden yhteiskäyttö ei toimi hyvin, tosin saattaa johtua beta-vaiheesta
- Työkalukeskeisyys
 - Mainospuheista huolimatta ilman työkaluja tekeminen varsin takkuista.
- Ei loppujen lopuksi tuo juuri uutta, paketoit olemassa olevaa pienin muutoksin ja antaa sille nimen.
 - AIR-kirjasto sisältää 3 uutta luokkaa.
- Adobe: AIR=uusi paradigma. Hmmm.