

## THIN@: BUDUĆNOST POSLOVNIH APLIKACIJA

Daniel Pavlič, RIS d.o.o., Donji Sroki 40, 51000 Rijeka, (385) 98 625 892, [daniel.pavlic@ris.hr](mailto:daniel.pavlic@ris.hr), <http://www.thinetsolution.com>

### SAŽETAK

Thin@ je specijalizirano rješenje za brz razvoj poslovnih aplikacija koje se nude kao usluga putem interneta. Tehnologija je nastala kao spoj najmodernijih koncepata razvoja poslovnih aplikacija kao što je SaaS (Software as Service), BIA (Business Internet Application) i RAD (Rapid Application Development). Ideja je da za razvoj poslovnih aplikacija koristimo visoko produktivan 4GL programski jezik koji automatski generira dijelove izvornog koda i time značajno ubrzava programiranje, a da korisnik za prezentacijski sloj može birati između izgledom klasične OOB (Out-Of-Browser) aplikacije, OOB Java aplikacije ili pak Java Applet Web aplikacije. Ova tehnologija usporediva je sa Adobe AIR, Oracle JavaFx odnosno Microsoft Silverlight thin client rješenjima, s tim što je usko specijalizirana, koristi visoko produktivan RAD alat i nudi integrirano SaaS rješenje.

Thin@ is a specialized Rapid SaaS Business Internet Application Development solution. It evolved as a blend of some of the latest technologies for business application development such as SaaS (Software as Service), BIA (Business Internet Application) and RAD (Rapid Application Development). The idea is that for business application development we use a highly productive 4GL which automatically generates parts of the source code and therefore significantly accelerates programming. For the presentation layer the end user can choose between a classic OOB (Out-Of-Browser) application, an OOB Java application or a Java Applet Web application. This technology is comparable to Adobe AIR, Oracle JavaFx and Microsoft Silverlight thin client solutions, the major differences being that it is highly specialized, highly productive and has integrated SaaS support.

### UVOD

Poslovne aplikacije budućnosti korisnicima će biti dostupne putem interneta u obliku servisa ili usluge.

Prema engleskom nazivu **Software as Service** stvoren je akronim **SaaS** kojim se naglašava da se aplikacija koristi kao usluga putem interneta. Engleski naziv **Rich Internet Application** (ili skraćeno **RIA**), ukazuje da se radi o aplikaciji koja ne samo da je dostupna putem interneta, već je vizualno i sadržajno bogata. Takve aplikacije odudaraju od dosadašnjih HTML i web tehnologija, između ostalog i svojim izgledom jer više nalikuju klasičnim 'desktop' aplikacijama. No postavlja se pitanje koju od dostupnih tehnologija odabrati kao platformu za izgradnju SaaS/RIA aplikacija.

Ovim su se pitanjem bavila i najpoznatija imena u softverskoj industriji među kojima su Adobe, Microsoft i Sun (sada Oracle). Njihov je rad rezultirao implementacijom različitih ideja i nuđenjem raznih rješenja: appleta koji se izvršavaju na klijent strani unutar preglednika (Java appleti i Flash), serverskih skripti koje generiraju HTML (JSP, ASP te besplatni PHP) ili kombinacije skripti na strani servera i JavaScripta na strani klijenta (AJAX).

Sve te tehnologije bile su vođene idejom da omoguće dostupnost internet aplikacije unutar web preglednika. Pritom su čak bili zanemareni temeljni nedostaci HTML-a kao platforme za razvoj internet aplikacija. Na primjer, stalna potreba za osvježivanjem čitave stranice riješena je tek AJAX tehnologijom.

Oni koji nisu bili zadovoljni onime što HTML tehnologija nudi, mogli su svoje aplikacije pokrenuti unutar web preglednika u obliku Java ili Flash appleta. Ako se obična aplikacija usporedi s automobilom, ovi postupci sličje pokušaju da se automobilu nadgrade krila kako bi letio, a zatim mu se za brži let ugradi još i motor rakete.

Na sreću, uskoro su vodeće tvrtke shvatile da HTML automobil jednostavno nije dizajniran da bi letio te su počele s proizvodnjom svojih prvih zrakoplova – alata **Adobe AIR**, **Java Web Start** i **Microsoft Silverlight**. Kako je jedna od glavnih značajki ovih alata pokretanje aplikacija **izvan web preglednika**, umjesto termina web aplikacije primjerenije je korištenje termina **internet aplikacije**.

Pokretanje internet aplikacije izvan web preglednika omogućuje i hrvatski alat **Thin@**. Thin@ je usko specijaliziran za razvoj i korištenje poslovnih aplikacija te u odnosu na druge alate nudi niz prednosti. Između ostalog, Thin@ omogućuje korištenje programskih jezika četvrte generacije koji

višestruko ubrzavaju posao programiranja svojom sposobnošću generiranja programskog koda. Ovo je samo jedan razlog, koji uz brojne ostale, daje zaključiti da Thin@ predstavlja respektabilnu i kompetitivnu platformu za korištenje poslovnih internet aplikacija.

## 1. NAJVEĆI IZAZOVI POSLOVNIH APLIKACIJA

Kada govorimo o poslovnim aplikacijama, možemo istaknuti nekoliko glavnih izazova koji se nameću tvrtkama koje takve aplikacije razvijaju, ali i krajnjim korisnicima.

**Prvi izazov je cijena.** Poslovne korisnike prije svega zanima koliko će ih koštati ulaganje u softver ili bio koji dio informatičke tehnologije. Proizvođači softvera rješenje ovog problema vide u modelu pružanja softvera kao usluge (engl. Software as Service, SaaS) što potvrđuje i sve veća popularnost spomenutog modela kojemu se u skoroj budućnosti predviđa dominacija na softverskom tržištu softvera. Jedna od glavnih prednosti SaaS-a je niska cijena inicijalnog ulaganja u softver.

**Drugi izazov je kvaliteta.** Osim pouzdanosti samog softvera, jedan od temeljnih zahtjeva kvalitete softvera jest njegova *dostupnost*. SaaS model predviđa dostupnost aplikacije 24/7 sa bilo kojeg mjesta koje ima pristup internetu. Iako su donedavno u trendu bile web aplikacije koje se pokreću unutar web preglednika, izvanpregledničke aplikacije (engl. out-of-browser applications) sve su popularnije, a tome mogu zahvaliti tehnologijama koje to omogućuju. Prije svega to su Java Web Start, Adobe AIR te Microsoft Silverlight.

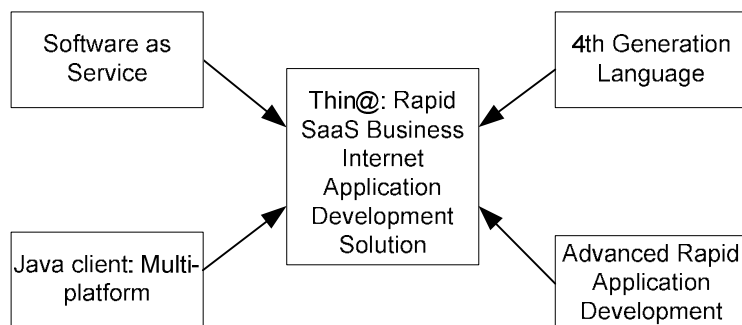
Još jedan od zahtjeva kvalitete jest *portabilnost*, odnosno mogućnost pokretanja aplikacije na različitim hardverskim i softverskim platformama, od Microsoft Windowsa, preko Linuxa i Mac OS-a, do Apple-ovog iPhone-a.

Slijede zahtjevi kvalitete kao što su stabilnost aplikacije, skalabilnost, sigurnost te brzina i interaktivnost korisničkog sučelja.

**Treći izazov je brzina isporuke.** Ukoliko softverska tvrtka gradi proizvod za široko tržište, tada se ovo odnosi samo na samu tvrtku, a utječe prije svega na troškove izgradnje proizvoda. Ukoliko se pak proizvodi softver prilagođen potrebama pojedinog korisnika, tada će brzina izgradnje predstavljati nešto što će osim softverske tvrtke itekako zanimati i samog korisnika.

Na brzinu izgradnje softvera znatno utječe odabir alata\platforme\riješenja kojeg će softverska tvrtka (u dogovoru s krajnjim korisnikom, ukoliko je riječ o softveru po narudžbi) odabrati za razvoj aplikacije. Softverskim tvrtkama je za razvoj poslovnih aplikacija najudrudnije odabrati alate koji omogućavaju brz razvoj softvera odnosno RAD alate koji koriste programske jezike četvrtke generacije.

### 1.1. THIN@ KAO ODGOVOR NA IZAZOVE POSLOVNIH APLIKACIJA



Thin@ je nastao upravo kao odgovor na potrebe i izazove modernih poslovnih aplikacija. Razvijen na temeljima 4GL programskog jezika za brz razvoj poslovnih aplikacija; udovoljava i najstrožim kriterijima SaaS-a a u isto vrijeme uspješno odgovara na zahtjeve kvalitete; prati najnovije tehnološke trendove – Thin@ zasigurno pokazuje u kojem smjeru će se kretati razvoj poslovnih aplikacija budućnosti.

## 2. TEHNOLOGIJA: POPULARNE RIA (Rich Internet Application) PLATFORME

### 2.1 Adobe AIR

Adobe Integrated Runtime (AIR) je platforma za razvoj bogatih Internet aplikacija korištenjem Adobe Flash, Adobe Flex, HTML ili AJAX tehnologija. Tako razvijene aplikacije mogu se prikazivati unutar internet preglednika kao web aplikacije ili izvan preglednika kao OOB (Out-of-browser) internet aplikacije.

Prednosti instalacije Adobe AIR aplikacije izvan preglednika uključuju pristup diskovnom prostoru i resursima lokalnog računala, dok su web aplikacije po tim pitanjima prilično ograničene. U siječnju 2009 Adobe je konstatirao da postoji preko 100 milijuna instalacija Adobe AIR-a diljem svijeta. Adobe AIR aplikacije moguće je razvijati kao HTML/AJAX aplikacije (primjerice korištenjem Adobe Dreamweaver CS4 alata), korištenjem Adobe Flash Buildera ili korištenjem Flash CS4 alata.



### 2.2 Oracle JavaFX

JavaFX je platforma za izradu RIA. Posljednja verzija (JavaFx 1.3) omogućuje izgradnju aplikacija za desktop, preglednike te mobilne uređaje. Planira se podrška za igraće konzole, blu-ray playere i druge platforme.

Za razvoj JavaFx aplikacija koristi se jezik nazvan JavaFx Script. JavaFx aplikacije mogu pokretati na bilo kojem računalu koje ima instaliran JRE (Java Runtime Environment), a na mobilnim uređajima Java ME.

Platforma se sastoji od JavaFx SDK-a, NetBeans IDE-a (korisnici Eclipsea mogu koristiti plugin zvan Project Kentai) te tzv. Production Suitea odnosno skupa pluginova za Adobe Photoshop, Adobe Illustrator isl.

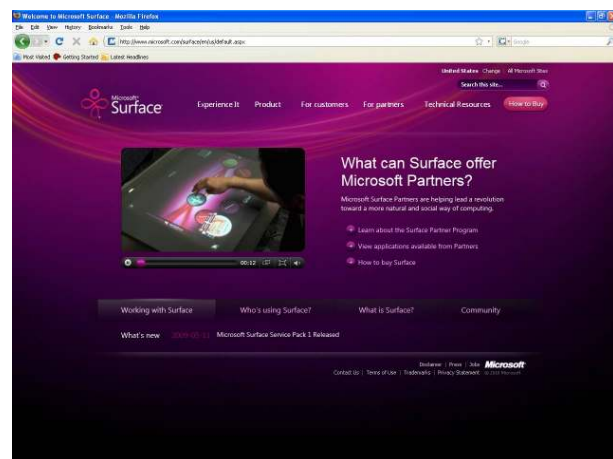


### 2.3 Microsoft Silverlight

Microsoft Silverlight je platforma za razvoj RIA koja pruža mogućnosti slične kao i Adobe Flash te integrira multimediju, grafiku, animacije i interaktivnost u jedinstveno okruženje. Silverlight je kompatibilan sa nizom internet preglednika koji se koriste na MS Windows i Mac OS X operativnim sustavima, kao i sa Windows 7 i Symbian mobilnim telefonima. Besplatan alat pod nazivom Moonlight omogućuje korištenje Silverlighta na Linux platformi.

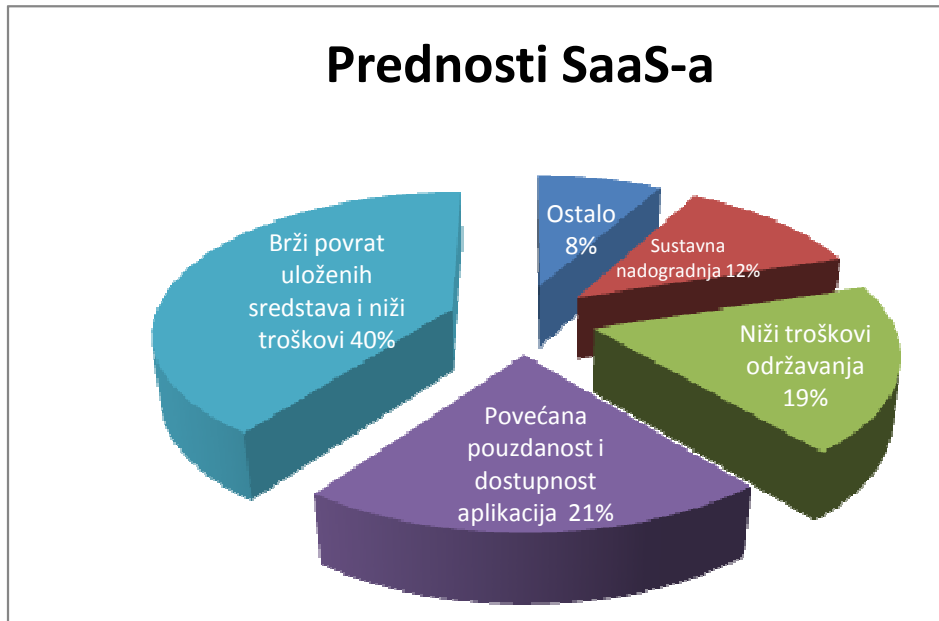
Korisničko sučelje deklarira se u XAML formatu (Extensible Application Markup Language) a za programiranje se koristi podskup .NET frameworka, dok za pokretanje Silverlight aplikacije nije potrebno imati instaliran .NET framework.

Kao i Adobe AIR te JavaFx, i Silverlight aplikacije je moguće koristiti kako unutar tako i izvan web preglednika.



### 3. SMANJENJE TROŠKOVA: SOFTWARE AS SERVICE (SAAS)

#### 3.1 Uvod u SaaS



Software as Service (SaaS) je način distribucije softvera kod kojeg pružatelj usluge (engl. Service provider) omogućuje krajnjem korisniku pristup softveru putem interneta.

Ovaj način distribucije softvera, koji prema Gartneru već sada obuhvaća 30 posto softverskog tržišta, s vremenom će postati još zastupljeniji, posebno kada je riječ o malim i srednjim tvrtkama. SaaS svoju rastuću popularnost može zahvaliti SOA tehnologijama (engl. Service Oriented architecture).

#### 3.2 Razine zrelosti SaaS aplikacije

SaaS aplikacije i njihove arhitekture mogu se, u povijesnom i razvojnom smislu, podijeliti u nekoliko razina zrelosti na osnovi ključnih karakteristika kao što su konfigurabilnost, mogućnost posluživanja više korisnika (Multi-Tenant) te skalabilnost.

Svaka viša razina razlikuje se od prethodne upravo po posjedovanju neke od gore navedenih karakteristika.

- **Prva razina – Aplikacije na zahtjev/po mjeri korisnika (Ad-Hoc/Custom)**  
Svaki Korisnik ima svoju vlastitu prilagođenu verziju aplikacije udomljene kod Nosioca usluge, te pokreće svoju zasebnu instancu aplikacije na serveru.
- **Druga razina – Konfigurabilnost**  
Na ovaj Nivo svrstavamo aplikacije koje su prilagodljive ili konfigurabilne kroz parametre, opcije ili "metapodatke", na način da ovi metapodaci prilagode izvođenje programa potrebama svakog pojedinog Korisnika.
- **Treća razina – Konfigurabilnost i Podržavanje Više Korisnika (Multi-Tenant)**  
Jedinstvena dinamički prilagodljiva instanca programa poslužuje sve korisnike (engl. Multi-Tenant).
- **Četvrta razina – Konfigurabilnost, Podržavanje Više Korisnika (Multi-Tenant) i Skalabilnost**  
Aplikacija se izvršava na serveru koji je dio tzv. farme servera. Ukoliko su potrebni dodatni hardverski resursi, farma se prema potrebi može jednostavno proširiti novim serverima, pa kažemo da je sustav skalabilan.

Thin@ zadovoljava četvrti, najviši stupanj razine zrelosti SaaS aplikacije.

### 3.3 Najčešći modeli naplate SaaS aplikacije

- **Pretplata**

Često korišten model kojeg korisnici jednostavno razumiju. Nije dobar za korisnike koji uslugu koriste samo povremeno. Također nije dobar ukoliko postoji maksimalna kvota koja onemogućava povećano korištenje usluge.

- **Naplata na temelju korištenja usluge**

Model naplate prema kojemu se usluga naplaćuje onoliko koliko je korištena.

- **Pretplata sa naplatom dodatnog korištenja**

Korisnik inicijalno kupuje predefiniрани paket za korištenje usluge. Ukoliko želi uslugu koristiti izvan paketa, korištenje se dodatno plaća.

- **„ Tekući račun“**

Korisnici ovakvog modela naplate koriste tekući račun koji se prazni korištenjem usluge, a po potrebi puni. Punjenje može biti ručno ili automatski.

- **Pre-paid bonovi**

Korisnici ovog modela kupuju tzv. pre-paid bonove koji im omogućuju fiksni iznos korištenja usluge. Prednost ovakvog modela je jednostavna distribucija širokim masama.

Thin@ platforma trenutno podržava Pretplatu, Naplatu na temelju korištenja i Pretplatu sa naplatom dodatnog korištenja.

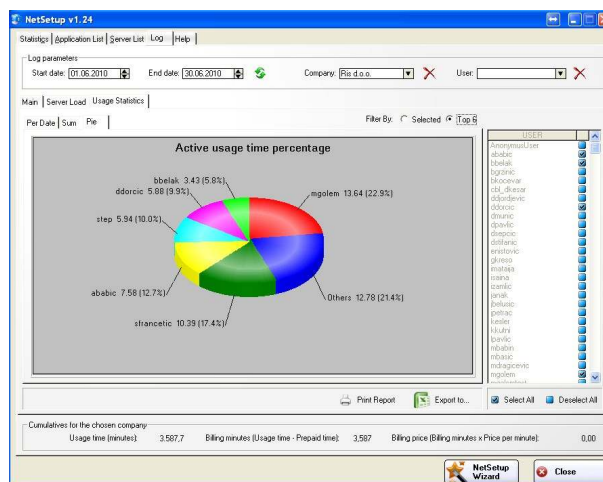
### 3.4 Thin@ podrška za napredne SaaS mogućnosti

#### 3.4.1 U obzir se uzima samo aktivno korištenje aplikacije

Thin@ koristi napredan sustav kalkulacije korištenja aplikacije koji u obzir uzima samo ukupno vrijeme u kojem je korisnik aktivno koristio aplikaciju, a ne ukupno vrijeme trajanja sesije. Također, nasumični klikovi mišem te nasumično tipkanje ne uzima se u obzir.

#### 3.4.2 Grafički prikazi i izvještaji

Thin@ nudi centralni alat za administraciju nudi grafičke prikaze (slika desno) i izvještaje opterećenja sustava i statistike korištenja sustava po firmi, aplikaciji i korisnicima. Na temelju ovih podataka moguće je uspostaviti račun za naplatu korištenja SaaS aplikacije. Također, zanimljiva mogućnost korisnicima je praćenje vlastitog aktivnog korištenja po osobi u vremenu isl.



#### 3.4.3 Jednostavna administracija SaaS sustava

- Centralizirana administracija poslužitelja
- Čarobnjaci za sve administrativne aktivnosti
- Mogućnost centralnog pregledavanja i obustave korisničkih sesija
- Integrirano praćenje korisničkih aktivnosti
- Automatska nadogradnja klijenta novom verzijom

#### 4. BRZINA RAZVOJA: CLARION 4GL PROGRAMSKI JEZIK I RAD RAZVOJNO OKRUŽENJE ZA BRZ RAZVOJ POSLOVNIH APLIKACIJA

Clarion je objektno orijentiran programski jezik četvrte generacije i razvojno okruženje za brz razvoj poslovnih aplikacija. Razvila ga je američka tvrtka SoftVelocity.

Jedna od najvećih prednosti Clariona je ta što koristi prilagodljive predloške (engl. templates) koji uz pomoć generatora aplikacija omogućuju automatsko generiranje programskog koda.

Povijesno gledano Clarion je jedan od prvih programskih jezika za MSDOS koji je imao 4GL alat za dizajn ekrana, stvaranje izvještaja i izravan pristup bazi podataka.

Clarionu je u osnovi manipulacija podataka (engl. dana-centric), što govori o njegovoj primarnoj upotrebi. Radi toga programski jezik ima naredbe koje omogućavaju izravan pristup bazi podataka. Prva verzija Clariona nastala je 1986 i razvio ju je Bruce Barrington. Posljednja verzija je Clarion 7 kojom je razvojno okruženje doživio redizajn i sada nalikuje na SharpDeveloper IDE (Integrated Developer Environment), od kojeg je Softvelocity otkupio izvorni kod.

Uskoro se očekuje izlazak .NET verzije Clariona, za što je razvijen novi programski jezik - Clarion#. Novi Clarion.NET IDE bi trebao omogućiti razvoj i Win32 i .NET aplikacija korištenjem Visual Basic.NET ili C# programskih jezika.

Programski jezik Clarion je po sintaksi i strukturi sličan Basicu, Pascalu i Cobolu, a osmišljen je tako da olakšava pisanje data-centric aplikacija. Drukčiji je od većine generičkih programskih jezika po tome što ima integrirane naredbe za manipulaciju podacima u bazi podataka, razne vrste gotovih prozora i izvještaja te naredbe za ispis podatka.

Temeljni nedostatak Clariona je njegova orijentiranost na desktop aplikacije odnosno nepostojanje kvalitetne podrške za razvoj Internet aplikacija. Thin@ uklanja ovu prepreku i uvodi moćan 4GL programski jezik u svijet poslovnih Internet aplikacija.



#### 5. THIN@ RJEŠENJE KAO ODGOVOR NA ZAHTJEVE KVALITETE POSLOVNOG SOFTVERA

- **Stabilan rad**

Kako bi osigurao stabilan i nesmetan rad SaaS sustava, Thin@ ima ugrađenu tzv. auto-reconnect opciju koja omogućuje automatsko ponovno spajanje na poslužitelj ukoliko dođe do privremenog gubitka veze.

- **Skalabilnost i balansirano opterećenje po serverima**

Thin@ podržava skalabilne SaaS aplikacije što znači da je moguće jednostavno dodati nove poslužitelje u farmu poslužitelja. Također, Thin@ Load Balancing servis vodi računa o jednakom iskorištenju resursa po poslužitelja.

- **Zaštita podataka**

Thin@ jamči apsolutnu zaštitu baze podataka. Bazi podataka nije moguće pristupiti izvana putem mreže kao što je to slučaj kod klasičnih klijent-server aplikacija. Pristup bazi podataka ima samo aplikacija koja se izvršava na aplikativnom serveru, nikad korisnik direktno.

Pored zaštite baze podataka, Thin@ koristi i 128-bitnu enkripciju pri slanju podataka putem interneta, čime se onemogućuje neželjeno čitanje podataka.

- **Sigurnost podataka**

Korisnici Thin@ SaaS tehnologije se mogu pouzdati u centralno čuvanje podataka na jednom mjestu, što uvelike smanjuje rizike vezane uz sigurnost podataka. Podatkovni centri kakve je moguće izgraditi unutar Thin@ arhitekture pružaju praktički stopostotnu sigurnost.

- **Brzo i interaktivno korisničko sučelje**

Krajnji korisnik Thin@ aplikacije ima nekoliko mogućnosti odabira korisničkog sučelja.

Ukoliko aplikaciju želi pokretati izvan web preglednika, što je trenutno trend, korisnik može birati između klasičnog Windows Win32 stila aplikacije i Java aplikacije.

Za pokretanje aplikacije koja stilom podsjeća na Windows Win32 aplikaciju potrebno je instalirati Thin@ klijent(veličine svega nekoliko MB).

Thin@ Java aplikaciju se pokreće iz web preglednika korištenjem tzv. Java Web Starta, no ona se izvršava izvan preglednika.

Ukoliko se pak aplikaciju želi koristiti kao web aplikaciju unutar preglednika, Thin@ se implementira kao Java web applet unutar kojega se izvršava aplikacija.

- **Portabilnost**

Portabilnost Thin@ aplikacija osiguravaju Java Web Start te Java applet način rada. Java Virtual Machine, koju u pravom smislu riječi možemo zvati platformom, krase mogućnost pokretanja Java aplikacija na doslovno bilo kojoj hardverskoj ili softverskoj platformi.

## 6. KORISNIČKO SUČELJE THIN@ APLIKACIJA

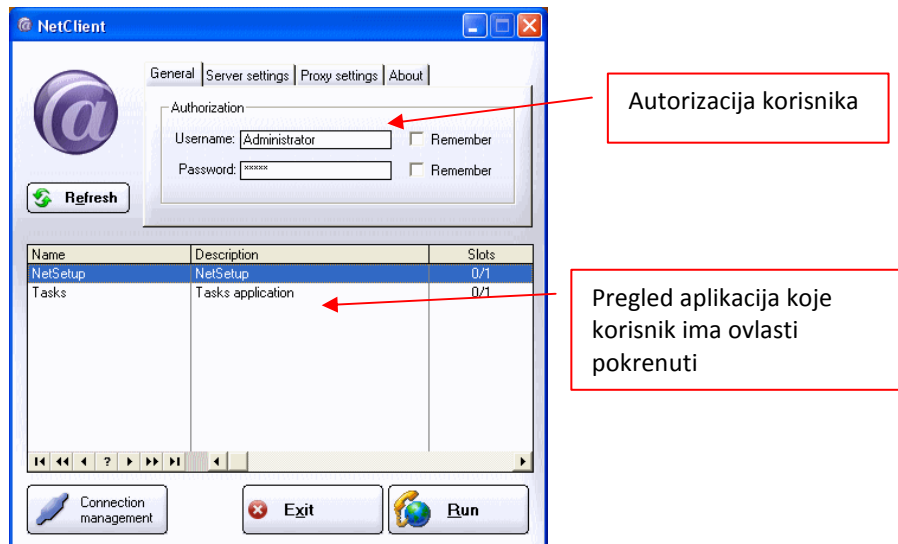
Izgled korisničkog sučelja Thin@ aplikacija ovisi o tome koju Thin@ platformu korisnik odabere.

### 6.1. Thin@ Windows Win32 stil

Osnovne prednosti Thin@ Windows Win32 načina rada uključuju i do nekoliko puta manje zauzeće računalne memorije u odnosu na ostale načine rada, kao i klijent veličine svega nekoliko megabajta, što u većini slučajeva znači inicijalno preuzimanje puno manje količine podataka sa interneta (u slučaju da korisnik već ne posjeduje Java Virtual Machine).

Osnovni nedostatak ovakvog načina rada jest nemogućnost pokretanja aplikacije na ne-Windows platformama, odnosno nedostatak portabilnosti.

Donja slika prikazuje izgled Thin@ klijenta koji je instaliran kod korisnika i omogućuje pokretanje aplikacije u Windows Win32 stilu.



## 6.2. Thin@ Java aplikacija

Thin@ Java aplikaciju moguće je pokrenuti iz web preglednika, i to putem Java Web Starta. Aplikacija se potom izvršava izvan web preglednika.

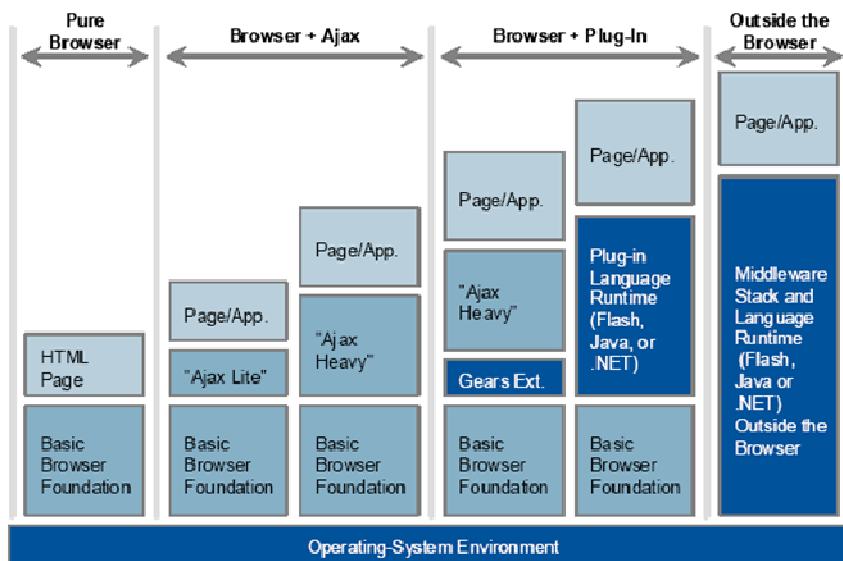
Osnovni nedostatak ovog načina rada jest veće korištenje memorijskih resursa računala u odnosu na Windows Win32 način rada.

Osnovna prednost ovog načina rada jest portabilnost, o čemu je ranije bilo riječi. Također, prednost u odnosu na Java applet način rada jest jednostavno i kvalitetno izvršavanje aplikacije neovisno o web pregledniku.



Donja slika prikazuje razvoj Internet aplikacija prema Gartneru. Vidimo da se aplikacije koje se pokreću izvan preglednika nalaze ispred aplikacija koje se pokreću unutar preglednika. Također, aplikacije koje se pokreću unutar preglednika korištenjem tzv. plug-inova nalaze se ispred Ajax aplikacija (kombinacija serverskih skripti poput PHP-a ili ASP-a te Java Scripti na strani klijenta).

Figure 1. Runtime Taxonomy for Ajax and RIA



Source: Gartner (December 2008)



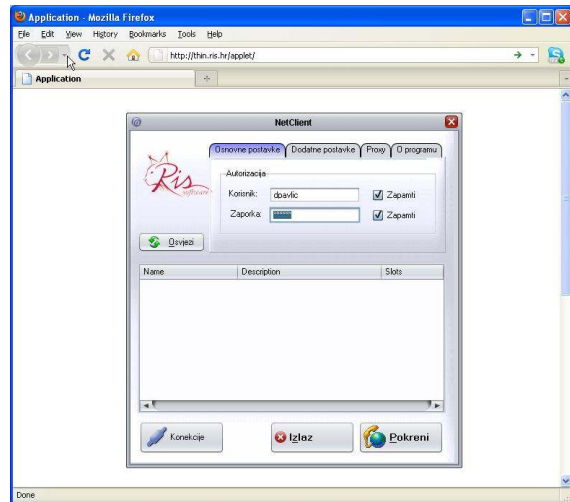
### 6.3. Thin@ Java Applet web aplikacija

Thin@ Java applet web aplikaciju pokreće se i izvršava unutar web preglednika.

Osnovna prednost ovakvog načina rada je, kao i kod klasične Java aplikacije, portabilnost. Također, kao prednost možemo spomenuti i naviknutost korisnika na web preglednike.

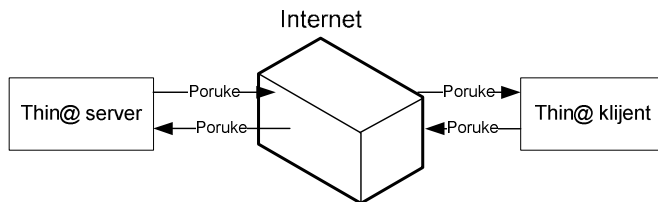
Nedostatak ovakvog načina rada jest sam preglednik, odnosno sve veći teret koji preglednici i njihova raznovrsnost predstavljaju velikim softverskim kućama, radi čega su velike softverske kuće zapravo i počele razmišljati izvan okvira web preglednika, pa danas bilježimo rastući trend trend out-of-browser tehnologija.

Slika s desne strane prikazuje izgled Thin@ Java applet web aplikacije.



## 7. TEHNIČKE KARAKTERISTIKE THIN@-A

### 7.1. Model sustava Thin@-a



Ukoliko korisnik želi koristiti izvan web preglednika kao aplikaciju koja stilom podsjeća na klasičnu Windows Win32 aplikaciju tada na svojem računalu treba instalirati Thin@ klijent aplikaciju.

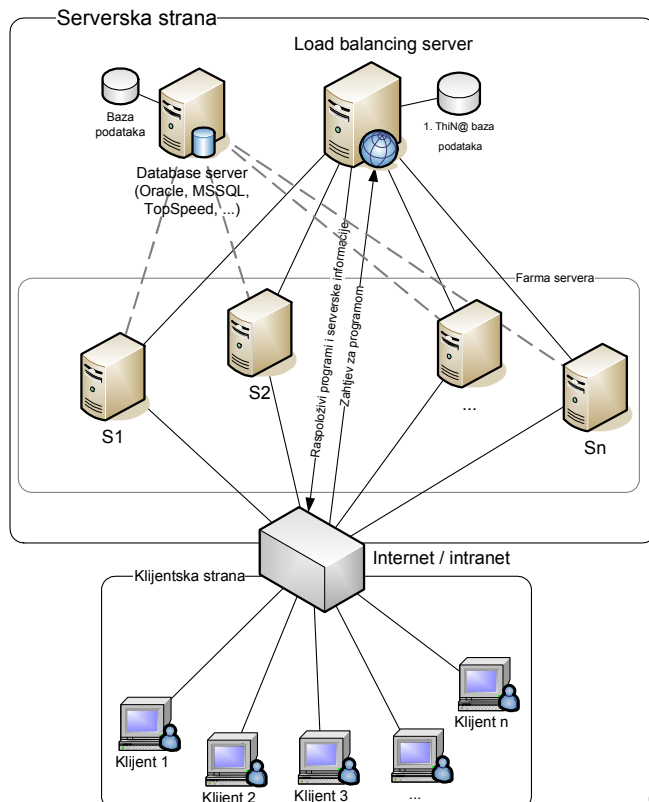
U slučaju pokretanja aplikacije izvan web preglednika putem Java Web Starta ili kao Java applet web aplikaciju korisnik ne mora instalirati dodatnog Thin@ klijenta, ali mora imati instaliran Java Virtual Machine koji u tom slučaju omogućava pokretanje Thin@ klijenta.

Instanca same aplikacije se pokreće na aplikativnom poslužitelju.

Čitava komunikacija između Thin@ poslužitelja i Thin@ klijenta se odvija u obliku poruka koje putuju internetom.

### 7.2 Model komunikacije Thin@-a

S porastom broja korisnika koji pristupaju aplikaciji rastu i hardverski resursi. Ukoliko postoji potreba, farmi poslužitelja mogu se pridodati novi poslužitelji koji potom preuzimaju nove korisnike koji ulaze u sustav.



### 7.3 Thin@ procesi na aplikativnom poslužitelju

Na aplikativnom serveru se pored same aplikacije koja koristi Thin@ izvršavaju i četiri Thin@ procesa odnosno Thin@ komponente (slika gore). Funkcije pojedinih komponenti opisane su u nastavku.

#### 7.4 Funkcije koje obavljaju pojedine Thin@ komponente

##### 1. Funkcije Thin@ NetListener-a

- prima zahtjev od korisnika
- klijentu šalje popis poslužitelja, adresu poslužitelja i mrežni port za spajanje
- pokreće aplikaciju

##### 2. Funkcije Thin@ NetMonitor-a

- nadzire sve aplikacije i ubija neaktivne
- šalje informacije o dostupnosti glavnom poslužitelju

##### 3. Funkcije Thin@ NetServer-a

- nalazi se samo na glavnom poslužitelju
- prima informacije o poslužitelju od NetMonitor-a
- šalje informacije o dostupnim poslužiteljima NetListener-u

##### 4. Funkcije Thin@ NetPort-a

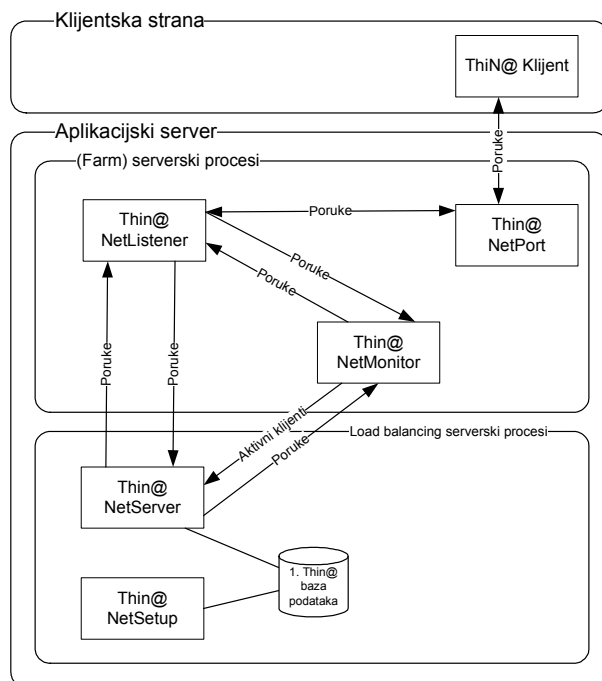
- Povećana sigurnost – klijenti se spajaju kroz jedinstveni port
- Smanjeni administrativni poslovi (otvaranje dodatnih portova na firewallu isl.)

##### 5. Funkcije Thin@ NetSetup alata za administraciju

- Omogućuje Centralno upravljanje poslužiteljima, korisnicima i aplikacijama
- Pregled i ažuriranje opcija glavnog poslužitelja
- Upravljanje korisničkim sesijama i njihova obustava

##### 6. Funkcije Thin@ NetClient-a

- Prima popis dostupnih aplikacija od NetListener-a na glavnom poslužitelju
- Šalje zahtjev za pokretanje aplikacije NetListener-u
- Prima odgovor i spaja se na dostupnu adresu
- „Crtá“ aplikacijske prozore
- Šalje zahtjev za obustavu serverskih procesa



## PRIMJERI ADMINISTRACIJE THIN@-A

### Administracija farme poslužitelja

Slijedeća slika prikazuje primjer administracije Thin@ farme poslužitelja korištenjem alata za administraciju Thin@ NetSetup.

The screenshot displays the NetSetup v1.03 application window. It features a menu bar with 'Statistics', 'Application List', 'Server List', 'Log', and 'Help'. The main area is divided into several sections:

- Grid statistics:** Shows total grid memory (2.0 GB), free grid memory (1.3 GB), average application memory (6.82 MB), and application slots (1/162).
- Selected server statistics:** Shows total server memory (2.0 GB), free server memory (1.3 GB), average application memory (6.82 MB), and application slots (1/162).
- Server List Table:** A table with columns: Server Name, IP Address, Server Memory (Used / Free, %), App. Slots (Used / Max), and Status. The first row is 'testserv' with IP 192.168.0.182, 709.3 MB / 2.0 GB memory, 35% usage, 1/162 slots, and 'Live' status.
- Application List Table:** A table with columns: Application, User, Client IP Address (External / Local), Start Time, Last Activity, and Status. The first row is 'ABBrowsers' with user 'mgolem' and client IP '192.168.0.182 \ 192.168.0.182', starting at 13:36:14 and ending at 0:10:39.

Annotations in red boxes point to specific features:

- 'Kumulativna statistika poslužitelja' points to the Grid statistics section.
- 'Statistika za označeni poslužitelj' points to the Selected server statistics section.
- 'Popis dostupnih Thin@ poslužitelja' points to the Server List table.
- 'Popis aplikacija koje se izvršavaju na poslužiteljima' points to the Application List table.
- 'Obustava označene korisničke sesije' points to the 'Terminate' button.
- 'Čarobnjaci za dodavanje poslužitelja, korisnika, aplikacija i licenci pojednostavljaju administraciju' points to the 'NetSetup Wizard' and 'Close' buttons.

At the bottom, there are buttons for 'Refresh', 'Database synchronization', and 'Terminate'. The 'NetSetup Wizard' button is highlighted with a red arrow.

## Administracija Thin@ aplikacija i autorizacije korisnika

Donja slika prikazuje primjer administracije aplikacija koje se pokreću na pojedinom serveru, kao i korisnika koji imaju pravo pristupa pojedinoj aplikaciji, te implementaciju SaaS naplate korištenja aplikacije.

**Popis registriranih aplikacija**

APPLICATION	COMMENT	LOCATION	TIMEOUT
ABBrowes	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
ABCForms	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
BrwGrid	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
BusMath	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
EIPNew	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
Features (ABC)	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
Graphs	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
NetSetup	NetSetup administration utility	C:\ThinNet\NetSetup.exe	0:30:00
People		D:\Thin@ - Related\DemoAppli	0:45:00
ReportOutput		D:\Thin@ - Related\DemoAppli	0:45:00
RTFNotepad		D:\Thin@ - Related\DemoAppli	0:45:00
RTFNotepad (ABC)	ABC Folder	D:\Thin@ - Related\DemoAppli	0:45:00
testapp		C:\ThinNet\NetSetup.exe	0:45:00

**Podaci o odabranoj aplikaciji**

Application Info

Application: ABBrowes

Comment: ABC Folder

Location: D:\Thin@ - Related\DemoApplikacije\HowTo-ABC\Browes\abcbrws.exe

Timeout: 00:45:00

Long timeout: 01:30:00

**Popis korisnika i tvrtki kojima pripadaju**

Users

Find user: [Search]

Company filter: [Company1]

USERNAME

- Daniel
- Everyone
- John
- Marko
- Michelle
- Stephanie

Username: [Marko] Password: [ ] Company: [Company1]

[Add] [Update] [Delete]

Add User Licence

**Period dozvoljenog korištenja aplikacije**

User Licences

LOGIN	FROM	TO	INST.
Everyone			1

Valid From: [ ] Valid To: [ ]

Simultaneous instances: [1]

[Update] [Delete]

**Popis licenciranih korisnika aplikacije**

Users

Code	Title	Prepaid mins	Price per min
001	Company1	30	0.05
002	Company2	0	0.20

**SaaS naplata korištenja: Mogućnost određivanja unaprijed plaćenih minuta korištenja aplikacije te cijene po minuti.**

## ZAKLJUČAK

U ovom radu izneseni su izazovi s kojima se susreću poslovne aplikacije i uloga koju Thin@ ima u rješavanju tih izazova. Predstavljene su neke od najpopularnijih RIA platformi kao što su Adobe AIR, MS Silverlight i Oracle JavaFx. U kratkim crtama je opisan model softvera kao usluge(SaaS) i načina na koji Thin@ udovoljava zahtjevima SaaS-a, te su prikazana jedinstvena obilježja Thin@-a u okvirima SaaS rješenja. Opisan je Clarion 4GL programski jezik i njegova uloga u povećanju brzine razvoja (internet) aplikacija. Opisani su načini pokretanja Thin@ izvan i unutar web preglednika, bilo kao Win32 aplikacije ili kao Java aplikacije odnosno Java appleta. Prikazane su tehničke karakteristike Thin@-a. Naposljetku su prikazani primjeri administracije Thin@-a.