

Přístup do IS z mobilních zařízení

Tomáš Tureček

Katedra informatiky, FEI, VŠB – Technická univerzita Ostrava
17. listopadu 15, 708 33, Ostrava-Poruba
Tomas.Turecek@vsb.cz

Abstrakt. Článek se zabývá problematikou přístupu do informačních systémů za použití mobilních zařízení. Zaobírá se problémy při návrhu distribuovaných aplikací, jež lze provozovat na mobilních zařízeních, jako je PDA a podobně. Jsou zde informace o technologiích, které lze k tomuto účelu použít. Dále jsou popsány problémy se synchronizací dat a je zde ukázka aplikace pro PDA, která je schopna do IS přistupovat.

Klíčová slova: Informační systém, PDA, webová služba, JAX-RPC, SOAP.

1 Úvod

S nástupem počítačů se informační systémy (IS) staly nedílnou součástí firem, institucí a také univerzit. Vždy bylo třeba nějakým způsobem uchovávat a zpracovávat různé informace, jako studijní výsledky studentů a jiná administrativní data. Většina univerzit má své vlastní IS, jež studentům umožňují přihlašovat se ke zkouškám, sledovat studijní výsledky a podobně. Většiny takových IS však bývají přístupné jen z prostředí webu, což klade na uživatele jistá omezení, jako mít k dispozici osobní počítač s internetovým prohlížečem.

V poslední době se počítače zmenšily natolik, že je lze nosit neustále při sobě. Trh nabízí mnohá zařízení, která jsou svým výkonem srovnatelná s osobními počítači a je možné na nich spouštět aplikace, jež mohou komunikovat s ostatními systémy. Mnoho lidí již dnes vlastní mobilní telefony, laptopy a PDA, tudíž je dalším logickým krokem počítat při návrhu IS s těmito mobilními zařízeními.

2 Popis problému

Jak již bylo výše uvedeno, většina IS umožňuje uživatelům přístup jen skrz webové rozhraní. To znamená, že k zobrazení informací vyžaduje webový prohlížeč a tím pádem osobní počítač. Například některé školy mívají nedostatečný počet veřejně přístupných počítačů. Většina počítačů se nalézá v učebnách, jež bývají přeplněné a během výuky jsou ostatním studentům nepřístupné. Učitelé sice mají počítače k dispozici, ale občas jsou v situacích, kdy počítač není v dosahu a přístup do IS by byl vítán. Pokud by to IS umožňoval, mohli by studenti i učitelé přistupovat k IS ze svých mobilních zařízení. V následujících sekcích budeme brát v úvahu jen PDA, ale navrhované řešení lze rozšířit i pro ostatní mobilní zařízení.

2.1 Komunikace

Představme si aplikaci spuštěnou na PDA, jež bude spolupracovat s IS. Každá operace, kterou provádíme, když přistupujeme k IS, začíná zasláním požadavku. Server požadavek zpracuje, provede požadované operace a zašle odpověď zpět. Je zde však problém společné komunikace dvou různých aplikací (aplikace na PDA a IS). Jedno z možných řešení je použít webové služby.

2.2 Webové služby

Webová služba je softwarový systém. Rozhraní a vazby jsou definovány a popsány v XML. Definice webové služby je veřejně přístupná a lze ji bez problémů vyhledat. Architekturu webové služby tvoří tři aktéři. *Poskytovatel služby* je de facto IS, který služby nabízí, dále *Žadatel služby*, což je klientská aplikace, jež do IS přistupuje a tzv. *Discover agency*, jež umožňuje vyhledání příslušné webové služby žadateli. Webová služba je serverová aplikace běžící někde na Internetu a naslouchá na nějakém portu na vzdálené vyvolání svých procedur, které poskytují dané informace, jako například ceny akcií a podobně. Při tvorbě klientské aplikace je nutné vědět, jak vyvolat procedury webové služby. Tyto informace jsou obsaženy v tzv. WSDL (Web Service Definition Language) dokumentu, který je veřejně dostupný a danou webovou službu detailně popisuje.

Implementace takové distribuované aplikace bývá náročná. Proto byly vyvinuty technologie, jako JAX-RPC, které tuto práci ulehčují.

2.3 SOAP

Simple Access Object Protocol (SOAP) využívá ke vzdálenému volání procedur XML zprávy. SOAP předepisuje jednotný formát zpráv pro přenos dat. Bývá využíván v distribuovaných prostředích pro výměnu informací.

2.4 JAX-RPC

JAX-RPC je Java API pro volání vzdálených procedur založených na XML. Umožňuje vývojářům pracujících na platformě Java vytvářet webové služby založené na protokolu SOAP. Tím, že využívá protokol SOAP a standardy definované WSDL, umožňuje spolupracovat i se systémy, které jsou založeny i na jiné platformě než Java. JAX-RPC nabízí vývojáři jistý komfort, neboť ten se nemusí ve většině případů zabírat mechanismy a protokoly na nižších úrovních.

3 Navrhované řešení

Řešení navrhované v následujících sekcích samozřejmě není řešením jediným. Popisovaná struktura odpovídá nově budovanému IS fakulty FEI na VŠB-TU

Ostrava. IS je rozšířen o webové služby, jenž klientské aplikaci umožní získávat data z IS. Struktura celého systému je navržena s ohledem na oddělení prezentace od aplikační logiky.

3.1 Server

Server je kompletně vybudován na platformě Java. Nabízí přístup ke zdrojům IS, jenž jsou přístupné prostřednictvím WWW stránek. Struktura je navržena podle vzoru Model-View-Controller.

Model je reprezentován DAO objekty (Data Access Object), jenž jsou Java Bean komponenty. Tyto objekty obsahují data a mají metody, jež z obsažených dat vytvářejí tzv. DOM objekty (Document Object Model). DOM jsou v podstatě strukturovaná data, která jsou poskytována dalším částem systému pro formátování a následné zobrazení.

View tvoří XSL šablony a transformátor, jenž transformuje DOM na základě šablony do některého z požadovaných formátů (HTML, WML, PDF, RTF atd.).

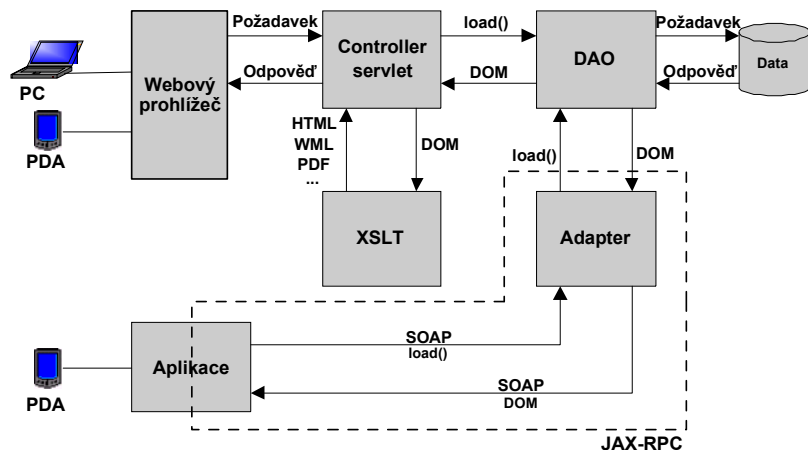
Controller je centrální část serveru, kterou tvoří servlet, jenž zpracovává požadavky a na jejich základě řídí celou činnost.

Přístup z klasického internetového prohlížeče probíhá následovně.

1. Uživatelův prohlížeč zašle požadavek.
2. Controller servlet požadavek analyzuje, najde správnou šablonu a vytvoří instanci správného DAO objektu s parametry získanými z požadavku.
3. Zavolá metodu *load()* DAO objektu
4. Metoda *load()* DAO objektu kontaktuje databázi a načte příslušná data. Z dat vytvoří instanci DOM objektu a vrátí jej jako hodnotu volané metody *load()*.
5. Controller servlet obdrží DOM objekt od DAO objektu a spustí na něj XSL transformaci s již dříve vybranou šablonou.
6. Controller servlet zašle uživateli prohlížeči odpověď – výsledek XSL transformace.

Server je dále rozšířen o webové služby. Na obr. 1 je patrný objekt Adapter, který má metody, jež jsou vzdáleně volány aplikací spuštěnou na PDA. Komunikace probíhá následovně.

1. Aplikace na PDA vyvolá metodu *load()* objektu Adapter.
2. Metoda *load()* objektu Adapter vytvoří instanci DAO objektu a zavolá jeho metodu *load()*.
3. DAO objekt, stejně jako v předešlém případě, načte data z databáze a vrátí Adapteru data ve formě DOM objektu.
4. Adapter vrací DOM objekt PDA aplikaci jako odpověď na volání své metody *load()*.
5. PDA aplikace zobrazí uživateli požadovaná data obsažená v DOM objektu.

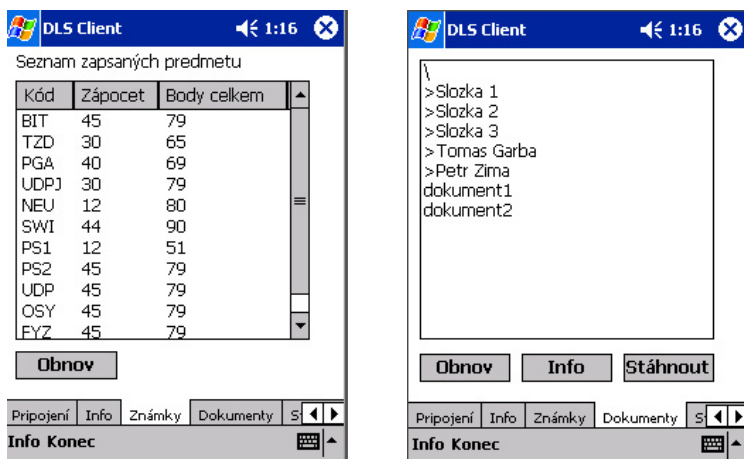


Obr. 1. Schéma IS FEI VŠB-TU Ostrava.

3.2 Client

Klienty jsou z 90% internetové prohlížeče, které komunikují se serverem přes HTTP protokol. Jako odpověď dostávají výsledek XSL transformace. Nejčastěji jde o HTML dokumenty (HyperText Markup Language). Může jít o prohlížeč provozovaný na osobním počítači, PDA, mobilním telefonu nebo na jiném zařízení. Správné formátování výstupu pro příslušné zařízení lze provést detekcí typu prohlížeče na serveru a následným výběrem příslušné šablony.

Druhým typem klienta je PDA aplikace. V tomto případě je využito modelu JAX-RPC. Aplikace volá vzdálenou metodu load() objektu Adapter. Zvoláním této metody získá aplikace DOM objekt, který následně vhodně zobrazí uživateli.



Obr. 2. PDA aplikace komunikující s fakultním IS.

3.3 Prohlížeč vs. PDA aplikace

Aby aplikace mohla komunikovat s webovou službou, musí mít přístup na Internet. Jelikož jde o mobilní zařízení a jeho hlavním účelem je možnost být spojen s IS odkudkoliv, musíme brát v úvahu, že uživatel může být připojen prostřednictvím mobilního telefonu. V takovém případě se většinou platí za množství přenesených dat. Zde má aplikace oproti prohlížeči obrovskou výhodu. Přenášejí se jen chtěná data a ta PDA aplikace vlastními silami zobrazuje. Prohlížeč vyžaduje navíc i formátovací informace, které mu sdělují, jak má uživateli informace zobrazit. Aplikace není omezena zobrazovacími možnostmi prohlížeče. Vše záleží na fantazii tvůrce.

Kromě zmíněných výhod nabízí PDA aplikace možnost tzv. offline přístupu. Jde v podstatě o to, že lze na PDA provozovat databázi, jež obsahuje data z IS a v případě online připojení synchronizuje databázi s databází IS. Tento offline přístup uživateli umožní pracovat s IS i v momentě, kdy není připojen. V tom okamžiku uživatel provádí operace nad vlastní databází, a když se spojení obnoví, provedou se v IS změny, které uživatel mezi tím udělal, tzn. obnoví se databáze.

Prohlížeče mají oproti popisované PDA aplikaci jednu velkou výhodu. Vytvořit prostředí za pomoci HTML dokumentů je mnohokrát jednodušší a univerzálnější, než vytvořit podobné prostředí v Javě nebo C#. Ale na druhou stranu lze na trhu nalézt množství CASE nástrojů, jež umožňují “naklikat” prostředí téměř stejně snadno jako vytvořit odpovídající HTML dokument.

4 Synchronizace dat

V předešlé kapitole byla zmíněna možnost offline přístupu do IS. Takový offline přístup vyžaduje, aby alespoň část informací z IS byla uloženo na zařízení, jež tento přístup umožňuje. Na zařízení provozovaná databáze obsahuje část dat z databáze IS. V okamžiku spojení s IS dojde k výměně informací mezi databázemi tak, aby obsahovaly totožné informace. Taková synchronizace dat není triviální záležitostí. Například je nutné identifikovat, které ze záznamů v tabulkách jsou aktuální a na základě této identifikace provést potřebné operace. V publikacích jsou popsány možnosti replikace databází, kdy se kopie databází nachází na různých místech a k synchronizaci dochází v pravidelných intervalech. PDA má však omezenou kapacitu a s ohledem na množství přenášených dat je nutno přijít na nějaký úsporný způsob. Zde přichází na řadu otázka budoucího výzkumu, kdy se bude třeba zaměřit na problematiku synchronizace dat jak na úrovni databáze (tabulek a záznamů), tak na vyšší úrovni – úrovni systému.

5 Závěr

Článek popisuje současnou práci v oblasti IS a nabízí řešení nových požadavků, které na IS kladou moderní mobilní zařízení. Počet uživatelů, jež vlastní nebo mají tato zařízení k dispozici, neustále roste a v budoucnu budou IS muset s těmito zařízeními

počítat. Nabízené možnosti offline přístupu s následnou synchronizací řeší zatím poměrně drahou komunikací.

Práce na zmíněných problémech je stále stále předmětem výzkumu. Je zde ještě mnoho problémů, které bude třeba rozebrat a vyřešit.

Reference

1. Garba T. *Rozhraní DLS pro mobilní zařízení*. Diplomová práce, Ostrava, 2003.
2. Wigley A., Roxburgh P. *Building .NET Applications for Mobile Devices*. Microsoft Press, Redmond, 2002, ISBN 0-7356-1532-2.
3. Drayton P., Albahari B., Neward T. *C# in a Nutshell*. O'Reilly & Associates Inc., Sebastopol, 2002, ISBN 0-596-00181-9.
4. Farley J., Crawford W., Flanagan D.: *Java Enterprise in a Nutshell*. O'Reilly & Associates Inc., Sebastopol, 2002, ISBN 0-596-00152-5.

Annotation.

Access to IS via mobile devices

The article describes modern demands to cooperation between information systems and mobile devices. It ruminates problems with coding a distributed application which can be launched on mobile devices like PDA. The article contains information about Web services which cooperate with other applications and about few technologies that form a base of these services.

Publikace ve kterých byly prezentovány dosažené výsledky

1. Beneš M., Běhálek M., Tureček T. Správa a prezentace dokumentů v informačním systému fakulty. sborník konference *Technologie pro e-vzdělávání*. Praha 2003. ISBN 80-01-02761-9.
2. Tureček T. Remote access to information system via PDA. sborník konference a soutěže Student EEICT 2003. ISBN 80-214-2401-X
3. Tureček T., Beneš M. Remote access to information system via PDA. sborník konference *ISIM'03*. Brno 2003. ISBN 80-85988-84-4.
4. Tureček T., Beneš M. Informační systém pro správu WWW jako prostředek pro sdílení dokumentů mezi skupinami. sborník konference *Technologie pro e-vzdělávání*. Praha 2002. ISBN 80-01-02538-1.