

tvom vytvoreného objektu *XMLHttpRequest* nadviaže spojenie so serverovou časťou označenou ako skript *brána* (obr. 1). Po úspešnom získaní odpovede vytvára z prijatej štruktúry XML zodpovedajúce elementy HTML s ich obsahmi, ktoré potom vhodne publikuje do príslušného pomenovaného elementu (funkciou *innerHTML*).

Tento prenos údajov zo servera môže byť orientovaný do jedného alebo viacerých elementov HTML. Podobne jeden objekt *XMLHttpRequest* môže obstarávať rôzne režimy komunikácie, alebo každej komunikácii môže byť priradený samostatný objekt *XMLHttpRequest*, popr. môže byť využívané celé pole týchto objektov. To všetko závisí na rozložení vstupných prvkov stránky, na koncepcii ich funkcií a na koncepcii asynchrónnej komunikácie medzi klientom a serverom. V našom prípade komunikáciu zabezpečuje jeden objekt *XMLHttpRequest*, ktorý odosiela na server vždy jednu trojicu vstupných parametrov a prijíma jednu alebo viac trojíc údajov z tabuľky.

Súbor *klienttempajax.js* je tvorený skupinou funkcií, ktoré budú popísané v poradí, v akom sa nachádzajú v súbore. Funkcia *createXmlHttpRequestObject()* vytvára inštanciu objektu *XmlHttpRequest* podľa typu prehliadača, ktorá zabezpečuje komunikáciu so skriptom *brána*. Funkcia *createRequestString()* zabezpečuje jednoduché spojenie troch reťazcov – parametrov volania skriptu *brána*. Funkcia *process()* vykonáva volanie skriptu *brána* (súbor *tempbrana.php*) a pri zmene jeho stavu (príchode odpovede webovej služby) volá funkciu *handleRequestStateChange()*, ktorá prekontroluje úspešnosť komunikácie a v prípade jej správnosti volá funkciu *handleServerResponse()*. Táto funkcia „rozoberá“ prijatú správu vo formáte XML s využitím DOM (*Document Object Model*) a vytvára z jej častí (v našom prípade *<cas>*, *<temp>*,

*<press>*) príslušnú štruktúru v HTML (v našom prípade *<table>*). Následne je vytvorený prvok HTML zapísaný do príslušnej položky formulára pomocou funkcie *innerHTML* (*myDiv.innerHTML*). Posledná funkcia *removeTable()* má za úlohu „zmazať“ zapísané hodnoty z formulára.

Je potrebné poznamenať, že klientsky skript dostáva výsledok webovej služby vždy v tvare XML štruktúry odoslanej serverom (napr. pre funkciu *posledné* (aktuálne) hodnoty má tvar *<response><data><cas>7</cas><temp>85.00</temp><press>130.00*

```
<?php //klienttempbrana.php
// load configuration file
require_once('error_handler.php');
header('Content-Type: text/xml');
// make sure the user's browser doesn't cache the result
header('Expires: Wed, 23 Dec 1980 00:30:00 GMT'); // time in the past
header('Last-Modified: ' . gmdate('D, d M Y H:i:s') . ' GMT');
header('Cache-Control: no-cache, must-revalidate');
header('Pragma: no-cache');
// read the action parameter
$rezim = $_GET['rezim'];
$params = $_GET['param1'];
$params2 = $_GET['param2'];
// we call the ws
$sm = new SoapClient(
('http://lipko.tuke.sk/~horovcak/php_ws/wsdl/temp.wsdl');
$xml=$sm->getTemp($rezim,$param1,$param2);
echo $xml;
?>
```

Obr. 7. Spojovací skript brána (*klienttempbrana.php*)

*</press></data></response>*), pre ktorú je potrebné vhodným spôsobom zabezpečiť zodpovedajúce zobrazenie. V našej ukážke je zobrazenie riešené s využitím kaskádového štýlu (súbor *styles.css*).

### 3.3 Spojovací skript brána

Spojovací skript *brána* zabezpečuje vlastné volanie webovej služby (konzumáciu) na základe znalosti jej rozhrania, ktoré poskytuje súbor WSDL služby (obr. 7), takže je vlastne klientom webovej služby. Výsledok volania webovej služby – funkcia *getTemp()* s príslušnými parametrami, získava skript v premennej *\$xm* (vo formáte XML), ktorá je následne zapisovaná na obrazovku klienta úlohy, to znamená je vstupom súboru v JavaScripte.

Pritom je dôležité správne nastavenie prehliadača (typ súboru *text/xml*, nedovoliť „kešovanie“ vzhľadom na kontinuálnu komunikáciu medzi formulárom a WS). Klientska časť webovej služby sa tak po doplnení o jednotlivé nastavenia prehliadača (pomocou hlavičiek *header*) stáva serverom JavaScriptu, ktorý zabezpečuje obojsmernú komunikáciu s využitím objektu *XmlHttpRequest*. Skript ďalej zabezpečuje preberanie vstupných parametrov (*\$\_GET[]*), vytvorenie komunikácie so serverom WS (premenná *\$my*), volanie príslušnej funkcie WS s danými parametrami (premenná *\$xm*) a jej odovzdanie na klienta Ajax (echo *\$xm*).

## 4. Záver

V rámci zostavenej ukážky sme sa pokúsili naznačiť možnosti a výhody servisne orientovanej architektúry. Použitie Ajaxu zjednodušuje a zrýchľuje úlohu z používateľského hľadiska, pripojenie jednej alebo viacerých webových služieb do aplikácie je veľmi jednoduché. Naopak najmä prácnosť manuálneho zostavenia a ladenia súboru v JavaScripte narastá. Webové služby je možné kombinovať na úrovni úlohy, ako aj priamo na úrovni webovej služby. Ukážka zostavenej úlohy v prostredí PHP5 (technológia Ajax v spojení s WS) je dostupná na adrese: [http://omega.tuke.sk/pavel.horovcak/php\\_ws/ajaxwsom/temp/klienttempajax.php](http://omega.tuke.sk/pavel.horovcak/php_ws/ajaxwsom/temp/klienttempajax.php)

Úloha bola vypracovaná v rámci riešenia projektov VEGA 1/4194 /07 (L), VEGA 1/0194/08 (S) a VEGA 1/0365/08 (T).

doc. Ing. Pavel Horovčák, CSc.,  
Ústav riadenia a informatizácie  
výrobných procesov, fakulta BERG,  
Technická univerzita v Košiciach  
([pavel.horovcak@tuke.sk](mailto:pavel.horovcak@tuke.sk))

Tento článok, ako i predchodzí s názvom *Architektúra SOA, Ajax a webové služby*, lektoroval Petr Klán, Ústav informatiky AV ČR, v. v. i., a Fakulta strojná, České vysoké učení technické v Praze.

## ► Těžaři ropy a plynu rozšiřují možnosti využití RFID

V podmínkách mimořádně vysokých cen surové ropy a prudkého růstu poptávky po ní po celém světě hledají těžbařské firmy způsoby, jak racionalizovat svůj vnitřní chod. Jednou z vhodných technik se jeví RFID, umožňující automatizovat či zdokonalit vše – od nákupu různých položek po zvýšení bezpečnosti a spolehlivosti provozního zařízení.

V čele snah o takovou podporu průmyslu nyní stojí *Oil & Gas RFID Solution Group*, sdružení expertů v dané oblasti, akademických

výzkumníků, dodavatelů techniky, představitelů předních ropných společností, organizací a firem jako Texas A&M University, University of Houston, Avery Dennison, Merlin Concepts & Technology, Shipcom Wireless, Motorola Inc atd. Tato širokospektrální skupina odborníků pomáhá prospektorům, vrtářům i producentům ropy a plynu nacházet způsoby efektivního použití techniky RFID k řešení problémů jejich denní praxe, definovat potřebné specifikace, datové modely a aplikační struktury a zavádět vyzkoušené pracovní postupy. Velká pozornost je přitom věnována krajně nepříznivým výrobním podmínkám, jímž je technika RFID

vystavena zejména ve vrtech a jejich blízkosti (přítomnost např. kyseliny chlorovodíkové, sirovodíku, barytu atd.). S uvedenou skupinou aktivně spolupracuje také EPC Global, mezinárodní standardizační orgán pro oblast RFID.

Cílem všech zúčastněných je nabídnout ropnému průmyslu nové systémy RFID schopné uchovávat v krajně nepříznivých podmínkách větší objemy dat, komunikují na větší vzdálenosti a přiměřeně nákladně. Ty pak otevřou nové možnosti využití techniky RFID v oblastech evidence zařízení a řízení výrobních i logistických procesů. [ARCwire, 25. července 2008.] (sk)