

Vývoj pro OpenCms aplikace KIV

Josef Krupička, říjen 2006

Přemek Brada, únor 2009

Web KIV je implementován ve webovém redakčním systému OpenCms (www.opencms.org).
V této příručce jsou základní informace o tom, jak se pro něj píšou moduly čili aplikace. Psáno pro OpenCms verze 6, mělo by být platné i pro OpenCms řady 7.

Další návody a informace najdete na <http://wiki.kiv.zcu.cz/OpenCMS/HomePage>.

Obsah

1. Vývoj modulů	2
Princip modulů.....	2
Vytvoření modulu.....	2
Import a export modulu.....	3
Action Class.....	3
OpenCms taglib.....	3
OpenCms API.....	4
CmsJspActionElement.....	4
Další důležité třídy.....	4
2. KIV moduly.....	5
3. Rady pro vývoj.....	6

1. Vývoj modulů

Princip modulů

Ve většině CMS je modul chápán jako samostatná komponenta, kterou je možno přidat na stránku, kde bude poskytovat přesně dané funkce (anketa, novinky, diskusní fórum). Modul v OpenCms je mechanismus, kterým můžete zaobalit vybraná data (JSP stránky, obrázky, šablony, normální stránky, atd.), která můžete přenášet mezi různými instalacemi OpenCms. Například dokumentace, demo stránky, WYSIWYG editory jsou distribuovány ve formě modulů, které můžete importovat do Vaší instalace.

Na modul lze také nahlížet jako na klasickou webovou aplikaci¹, která používá OpenCms knihovny tagů a API. Není tudíž potřeba implementovat nějaké API, aby byl váš modul použitelný. Všechny standardní moduly OpenCms se nacházejí v adresáři `/WEB-INF/packages/modules`, kde jsou zazipované.

Vytvoření modulu

V administrátorské části Workplace můžete vytvářet, editovat, mazat, importovat a exportovat moduly. Stisknete *New module* a objeví se následující formulář:

Administration View > Module Management > New Module

New Module

Module information

Package name:

Module name:

Module description:

Module version:

Module group:

Action class:

Module creator

Author name:

Author email:

Module folders

Create Modulefolder:

Create templates subfolder:

Create elements subfolder:

Create resources subfolder:

Create classes subfolder:

Create lib subfolder:

Ok Cancel

¹ Zde je míněna webová aplikace, která používá pouze JSP stránky a vlastní knihovny tagů. Se servlety a webovými frameworky je problém a autorovi se nepovedlo je zprovoznit.

U každé položky ve formuláři je kontextová nápověda v angličtině, která popisuje její funkci. Po stisknutí *Ok* je vytvořen nový modul, který je uložen v adresáři */system/modules*. Během vytváření nového modulu je vytvořena základní struktura, která se skládá z několika adresářů:

- *classes* – zde jsou uloženy především properties soubory, které slouží např. k lokalizaci.
- *lib* – obsahuje knihovny, které tento modul potřebuje. Během importování modulu budou nakopírovány do classpath OpenCms
- *templates* – JSP soubory, uložené v tomto souboru se objeví ve výběru šablon, při vytváření nové stránky
- *resources* – slouží k ukládání obrázků, css nebo javascriptových souborů

Nejste omezeni jen těmito adresáři a lze přidávat další adresáře. K modulu můžete připojit i další zdroje z VFS, které se nacházejí mimo jeho adresářovou strukturu. K tomuto slouží formulář *Module resources*. Pokud vytváříte více modulů, které na sobě nějak závisí, je možné nastavit potřebné závislosti ve formuláři *Module dependencies*.

Import a export modulu

Volbou *Export module* z administrátorského rozhraní se vytvoří zip soubor, který obsahuje všechny objekty tohoto modulu. V souboru *manifest.xml* jsou uloženy vlastnosti všech objektů. Exportovaný modul je uložen v adresáři *WEB-INF/packages/modules*. Při každém exportu dojde k automatické inkrementaci verze.

Modul lze importovat dvěma způsoby:

- nahrát patřičný soubor do adresáře *WEB-INF/packages/modules* a zvolit volbu *Import module from server*. Zobrazí se seznam modulů uložených právě v tomto adresáři.
- zvolit volbu *Import module with HTTP*, kde zadáte cestu k zazipovanému modulu

Pokud modul obsahuje nějaké javovské knihovny, properties resp. třídy je potřeba po exportu restart OpenCms aplikace.

Action Class

Pokud chcete provádět nějaké specifické operace během určitých událostí je možné vytvořit třídu implementující rozhraní *org.opencms.module.I_CmsModuleAction*. Tuto třídu pak musíte zadat do parametru modulu *Action class*. Události, na které je možné reagovat:

- *initialize, shutdown* – bude zavolána při startu resp. ukončování OpenCms.
- *uninstall, update*

OpenCms taglib

Dostupné tagy:

- *cms:user* – poskytuje data o právě přihlášeném uživateli (login, osobní údaje, ...)
- *cms:property* – umožňuje přístup k vlastnostem aktuálního souboru (jeho rodičovských adresářů).
- *cms:link* – vytvoří URL pro daný objekt z VFS. Nemusíte vkládat jméno aplikace a servletový kontext ručně. Také je takto vytvořený odkaz správně ošetřen při statickém

exportu.

- *cms:include* – slouží k dynamickému vkládání souborů z VFS.
- *cms:label* – zobrazí lokalizovaný řetězec z OpenCms properties souborů.
- *cms:template* – tímto tagem je možné rozdělit JSP soubor na několik částí, které pak lze vložit do jiného JSP použitím tagu `<cms:include>`.
- *cms:info* – zobrazí řadu informací o OpenCms a prostředí (Java, OS), ve kterém běží. Podstatné jsou informace o aktuálním požadavku (normální URL, OpenCms URL, ...).
- *cms:editable* – nastavuje možnost přímého editování HTML stránek a XML contentu.
- tagy pro práci s XML daty – tyto tagy slouží k získávání dat ze XML content souborů.

Podrobnější informace a příklady použití najdete v modulu *Alkacon documentation*. Řada důležitých funkcí však v taglib není a je nutno je volat ze scriptletů.

OpenCms API

CmsJspActionElement

Všechny funkce z taglib, jsou dostupné i ve scriptletech. Stačí vytvořit instanci třídy *CmsJspActionElement*.

```
org.opencms.jsp.CmsJspActionElement cms =new CmsJspActionElement(pageContext, request, response);
```

Další dostupné funkce:

- vytváření navigace – metodou *getNavigation()* získáte instanci třídy *CmsJspNavBuilder*, která obsahuje metody pro vytváření vlastní navigace (menu, breadcrumbs, sitemap, ...).
- přístup k základním funkcím OpenCms – metodou *getCmsObject()* získáte instanci třídy *CmsObject*, která obsahuje metody pro většinu podstatných funkcí (CRUD operace nad objekty, publikování projektu, práce s vlastnostmi, ...).

Další důležité třídy

Pro mnohem podrobnější informace Vás odkazují na obsahově bohatý JavaDoc. Toto je pouze stručné seznámení.

- *CmsObject* – základní třída, která poskytuje většinu podstatných funkcí OpenCms (v dokumentaci je uveden termín „initialized shell“):
 - CRUD operace nad objekty a jejich vlastnostmi
 - nastavování oprávnění, zámku, uživatelů, skupin, ...
 - práce s projekty a spousta dalších funkcí
- *CmsRequestContext* – obsahuje informace o aktuálním požadavku (url, v jakém projektu se nachází, identitu přihlášeného uživatele)

Opět podrobnější informace najdete v programátorské dokumentaci OpenCms.

2. KIV moduly

TBD aktualizovat

3. Rady pro vývoj

Velmi užitečnou vlastností OpenCms je možnost synchronizace dat mezi VFS a normálním FS. Doporučuji přečíst dokument [OpenCms_synchronizace.pdf](#). Dále je nutností stáhnout zdrojové kódy OpenCms, ze kterých se toho o fungování systému naučíte nejvíce. Vzhledem k povaze OpenCms modulů se vyplatí mít kvalitní editor JSP souborů.

K dispozici jsou i zajímavá rozšíření OpenCms, která vývoj nových modulů usnadní:

- FTP přístup k VFS (*opencmsftpd*). Bohužel stránka tohoto projektu během psaní tohoto tutoriálu nefunguje, ale zdrojové kódy a binární verze existuje. V mailing listech lze najít návody, jak toto rozšíření nainstalovat.
- WebDAV přístup k VFS. TBD doplnit
- Eclipse plugin, který umožňuje připojit se k lokální instalaci OpenCms a provádět změny ve VFS. Domovská stránka projektu: <http://www.redstardevelopment.nl/opencms/opencms/en/opencms/plugin/>.

Vzhledem k povaze většiny uvažovaných modulů (CRUD aplikace nad KIV databází), lze moduly vyvíjet jako samostatné webové aplikace, které do OpenCms vložíte později. Důvodem je především rychlejší vývoj. Restart Tomcatu s nainstalovaným OpenCms zabere několik vteřin (10s a více).