

Jazyčky na síti

Chcete zvládat rychlý vývoj webových aplikací v programovacím jazyku, který vám sedne do ruky, udělá spoustu práce za vás a má k dispozici užitečné rozšiřující moduly, například práci s SQL databází, čtení dat z formulářů a uživatelské relace neboli sessions? Chip vám poradí, jak na to – představíme vám tři takové programovací jazyky, možná mezi nimi najdete svého favorita.

Světlu webových aplikací vládou skriptovací programovací jazyky – na půdě Windows je to Visual Basic skrytý v ASP stránkách, na půdě Linuxu a serveru Apache je to nejčastěji Perl nebo PHP. My se zaměříme na jazyky Perl, PHP a nastupující jazyk Python; zájemce o Visual Basic a ASP stránky odkážu na seriál kolegy Dvořáčka v loňském Chipu. Nejdříve se budeme věnovat obecné motivaci a historickému úvodu, k ukázkám kódu a technickým detailům se dostaneme v některém z příštích Chipů.

Skriptovací jazyky se od klasických programovacích jazyků typu C, C++ odlišují. Jejich návrh směřuje k tomu, aby programátor zvládl vyřešit danou úlohu co možná nejrychleji, na rychlosti běhu výsledné aplikace tolik nezáleží. K popularitě skriptovacích jazyků přispěl růst výkonnosti počítačů. Z toho vyplývají společné vlastnosti skriptovacích jazyků – programátora odstiňují od detailů operačního systému “hezkým” abstraktním rozhraním, starají se o správu paměti, pohodlně pracují s řetězci, poli a asociativními poli. Program ve skriptovacím jazyku je díky tomu několikrát kratší než funkčně podobný program v C/C++, programátor se nestará o každý bit a bajt a základní datové struktury, ale o funkčnost aplikace jako takové.

Skriptovací jazyky jsou většinou interpretované, a nikoli kompilované, takže programátor bez dlouhé kompilace projektu vidí důsledky změn v kódu. Typickým způsobem vývoje aplikace ve skriptovacím jazyku je inkrementální vývoj stylem pokus – omyl. Vývojář napíše kousek kódu, hned jej bez kompilace vyzkouší, opraví případné chyby a krok za krokem postupuje dál. Výsledný program je ihned přenositelný na jiný operační systém, aplikaci vytvoříte na svém domácím péčičku s Windows a pak ji nahrajete na unixový webový server, žádný problém. Interpretery populárních skriptovacích jazyků jsou k dispozici na většině operačních systémů – Perl, PHP i Python najdete na Windows, na Linuxu/Unixu i Macintoshi, spolupracují s různými webovými servery – určitě s Microsoft IIS a Apache, dvěma “největšími”.

Skriptovací jazyky jsou kvůli svým vlastnostem vhodné pouze pro některé typy úloh – operační systém v nich nenapíšete, to je stále úkol pro C či C++, naproti tomu pro webové aplikace jsou tyto jazyky jako stvořené. Webové aplikace musejí být hotovy rychle, musejí být flexibilní k provádění drobných změn, hodně pracují s textovými daty, měly by být přenositelné na jiné operační systémy pro případ změny webového serveru – aplikační oblast jako stvořená pro skriptovací jazyky.

Nadále se budeme věnovat jazykům Perl, PHP a Python, ale je škoda nezmínit jejich zajímavé kolegy, také použitelné pro webové aplikace, byť zatím méně rozšířené: TCL, Rebol, Pike, Ruby, Frontier, Guile, Haskell a třeba ještě Cold Fusion.

Perl

Klasikem mezi skriptovacími jazyky je právě Perl. Jeho první verze spatřila světlo světa už v roce 1987, od té doby se Perl vyvíjel, přijímal podněty z okolí, databáze, objektově orientované programování, web, a dnes je i přes nástup mladších jazyků patrně nejpoužívanějším ve své kategorii. Pan Larry Wall, autor Perlu a dodnes jeho hlavní architekt, má smysl pro humor a komunita vývojářů Perlu se umí bavit. Poezie psaná v Perlu, konference o Perlu pořádaná na palubě parníku plujícího kolem Aljašky a hesla typu “Perl podporuje tři vlastnosti programátora: lenost, zbrkllost a pýchu”, to je rozptýlení v těžké programátorské práci.

Vývoj Perlu předznamenává jeho podobu – Perl není dělán proto, aby byl samoučelně hezký, ale proto, aby byl účelný. Jeho syntaxe tedy neoplývá krásou abstraktních konstrukcí, ideje si vypůjčuje z C, C++, Ady, z jazyků unixové příkazové řádky, z archaického jazyka AWK, ale programy se v něm dají vyjádřit stručně a stylově.

V Perlu lze programovat různými styly – podobně jako v klasickém C, čistě objektově, nebo podobně jako na unixové příkazové řádce; každý si vybere.

K popularitě Perlu přispěla zásobárna rozšiřujících modulů CPAN (Comprehensive Perl Archive Network). Dá se v ní najít spousta vyřešených úloh, které prostě vezmete a upravíte pro svůj případ, licenční politika autorů to dovoluje. Chcete si napsat vlastní webový mail? Proč složitě

studovat poštovní protokoly POP3, IMAP, SMTP a normu MIME, když existují hotové moduly v Perlu, jejichž služeb můžete využít. Takových modulů jsou tisíce – přístup k SQL databázím, generování GIF obrázků, spolupráce s webovým serverem; to je krátký výběr z řady.

Dosud jsme mluvili o hezkých stránkách Perlu, v rámci objektivitu však musím zmínit i druhou stranu věci. Perl není příliš vhodný jako první programovací jazyk, velká flexibilita jeho syntaxe může začátečníkům dělat potíže. Vzhledem k množství jeho rysů a idejí převzatých z několika jazyků není vždy jasné, odkud začít se studiem. Přejít na Perl se tedy obvykle doporučuje programátorům znalým jiného programovacího jazyka, například C/C++.

V neposlední řadě, Perl nebyl navržen jako jazyk speciálně pro webové aplikace, ale jako univerzální jazyk. S webem si díky svojí flexibilitě rozumí výborně, ale pokud chcete vyvíjet pouze webové aplikace, může vám to v některém jiném jazyku připadat jednodušší.

Na závěr ještě jeden klad Perlu – je velmi rozšířený, programátor v Perlu poměrně dobře najde práci a firma poměrně dobře najde programátora v Perlu, to platí alespoň na světovém webu.

PHP

Mladík mezi skriptovacími jazyky, narozen v roce 1994, ale velmi rychle roste a vyvíjí se, ohrožuje suverénní pozici Perlu. Velmi populární na českém internetu, z českých internetových společností na něj sázejí třeba Seznam, Centrum a Mobil Server. Rychlé charakteristiky mohou znít “jednodušší C++” nebo “Perl pro normální lidi”.

PHP navrhl jeho autor Rasmus Lerdorf jako jazyk pro tvorbu jednoduchých webových aplikací typu počítadlo nebo návštěvní kniha. Skvělým krokem autora bylo propojení s tehdy populární databází mSQL (někdy kolem roku 1995), základna vývojarů rostla, jazyk dostával nové rysy a hlavně rozšiřující moduly – důležitý rys, který vývojarům šetří práci, a ti na oplátku píšou další moduly... Časem se jazyk PHP dostal na další platformy, přibyla podpora objektového programování, spolupráce s databázemi včetně MySQL na Linuxu a normy ODBC na Windows, objevily se učebnice a knihy, jazyk PHP se stal dospělým.

Výhodou PHP je jeho dobrá čitelnost a srozumitelnost, PHP je vhodný i jako první programovací jazyk pro výuku programování. Člověk, který trochu umí některý programovací jazyk, dokáže po chvíli pokusů a omylů upravovat a tvořit funkční kód v PHP. Jazyk je od začátku navržen pro tvorbu webových aplikací a spousta věcí z této oblasti v něm funguje jednoduše a logicky. Jeden příklad za všechny: vstup z formuláře je ve skriptu automaticky k dispozici v proměnných stejného jména, jako je název vstupního prvku ve formuláři.

Jazyk PHP býval velmi pomalý (třeba ve srovnání s Perlem), to však od verze 4 neplatí. Společnost Zend, založená autory PHP, přepsala interpreter s ohledem na rychlost a přenositelnost, pracuje se na dalších zlepšeních, například kompilátoru a vizuálním debuggeru jazyka.

V rámci objektivitu nejen chvála – PHP zatím nemá moc dobře vyřešenu distribuci rozšiřujících modulů ani jejich jednotné rozhraní, chybí obdoba CPAN z Perlu a programátoři musejí ručně vkládat hotové kousky kódu do svých programů a podle potřeby upravovat. Na řešení se pracuje, norma je navržena a zbývá doufat, že se ujme v praxi.

Python

Nejméně známý jazyk z této trojice. Začal vznikat v roce 1989 ve výzkumném ústavu v Amsterdamu, jeho autorem je Guido van Rossum. Správně tušíte, název autor vybral podle své oblíbené skupiny humoristů Monty Python's Flying Circus, jednoho z příspěvků televize BBC světové kultuře.

Vědecké zázemí autora se nezapře, navrhl velmi hezký a čistě objektově orientovaný skriptovací jazyk, který ovšem dlouho zůstával ve stínu jiných. Situace se změnila o mnoho let později, když firma Digital Creations uvedla aplikační server Zope, který je napsán právě v Pythonu – výkřiky nadšení recenzentů typu “splněný sen vývojarů webových aplikací” přiměly mnoho lidí, aby se s Pythonem seznámili a zjistili, že to je velmi dobrý skriptovací jazyk. Komunita kolem Pythonu začala růst, objevily se užitečné rozšiřující moduly, byly napsány články v odborném tisku a na odborných webech, Python začal být brán vážně.

Python je velmi dobře čitelný a pochopitelný “na první pohled”. Nemá žádné závorky oddělující bloky kódu, strukturu programu určuje odsazení od okraje – zní to zvláště, ale funguje to docela dobře a hezky to vypadá.

Dnešní Python nabízí spoustu rozšiřujících modulů, rozumí si s databázemi, spolupracuje s různými webovými servery, s Javou, s ActiveX, rozumí internetovým protokolům – samá pozitiva. A nějaké minusy? Snad jen menší popularita ve srovnání s dvěma dříve jmenovanými a skoro až moc nesmiřitelná nálada fanoušků Pythonu vzhledem k jiným jazykům.

Jan Stoklasa
jstoklasa@writeme.com