

9 Python a vývoj webových aplikací

Již ve třetí kapitole jsme se zmínili o tom, že programovací jazyk Python je možné využít i pro vývoj webových aplikací. Již tradičně a po velmi dlouhou dobu se Python v této oblasti skutečně používá, a to primárně pro tvorbu *back endu*, tedy (zjednodušeně řečeno) té části webové aplikace, která běží na serveru a s níž komunikují webové prohlížeče klientů. Ovšem nyní nás bude zajímat především nasazení Pythonu na *front endu*, což je část webové aplikace běžící přímo v rámci webových prohlížečů. V této oblasti se stále nejvíce setkáme s JavaScriptem popř. s technologií *WebAssembly* (zkráceně *WASM*).

Ovšem i fakt, že webové prohlížeče podporují pouze JavaScript, nás nemusí omezit pouze na tento programovací jazyk. Existují totiž takzvané *transpřekladače* (*transcompiler*, *transpiler*) umožňující transformaci zdrojového kódu z jednoho programovacího jazyka do jazyka jiného. V tomto konkrétním případě tedy o transformaci mezi Pythonem a JavaScriptem. A alternativně je možné samotný interpret Pythonu přeložit do WebAssembly.

9.1 Lze Python použít pro webový front-end?

Na tomto místě je vhodné si položit otázku, jestli je vůbec možné a praktické použít programovací jazyk Python přímo na webovém front-endu a pokud to je možné, jakou za to zaplatíme cenu. Ve skutečnosti existuje hned několik projektů, které se více či méně úspěšným způsobem snaží o to, aby byl Python na front-endu skutečně použitelný, a to i pro reálně provozované aplikace. Mezi tyto projekty patří například:

1. *Brython*. Jedná se o transpřekladač zdrojových kódů z Pythonu do JavaScriptu, přičemž samotný překlad (resp. přesněji řečeno transpřeklad) probíhá na pozadí. To v praxi znamená, že přímo v kódu webové stránky je umístěn (či načítán) zdrojový kód napsaný v Pythonu. Velkou předností Brythonu je, že se jakékoli úpravy tohoto kódu ihned projeví v aplikaci po znovunačtení stránky (F5), jednou z nevýhod pak pomalá prvotní inicializace aplikace z důvodu jejího transpřekladač.
2. *Transcrypt* je taktéž transpřekladačem Pythonu (konkrétně Pythonu 3) do JavaScriptu. Samotný transpřeklad je napsaný velmi dobře – zhruba platí, že velikost vygenerovaného kódu v JavaScriptu odpovídá velikosti původního kódu napsaného v Pythonu (případné zvětšení je o jednotky procent). Musíme však počítat s tím, že je nutné načíst i jádro Transcryptu, jehož velikost je přibližně 20 kB (tedy čím větší je kód samotné aplikace, tím (poměrově) menší budou režijní náklady). Zajímavá a užitečná je i podpora *type hintů* (informací o datovém typu parametrů, proměnných atd.), což bylo téma, kterému jsme se věnovali v páté kapitole.
3. *pyjaco* je projekt, který pracuje na velmi podobném principu jako výše zmíněný projekt *Transcrypt*.

4. *Skulpt* je naproti tomu projekt, který se spíše podobá *Brythonu*, protože taktéž umožňuje provádět transpřeklad na pozadí. Programátor tedy pracuje pouze se zdrojovým kódem vytvořeným v Pythonu a nevidí ani nemusí vidět výsledek transpřekladu.
5. *fiddlesalad* je velmi zajímavým nástrojem, který názorně ukazuje, jak vlastně transpřeklad funguje (a že jeho výsledek může být čitelný). Tento nástroj umožňuje zápis zdrojového kódu v Pythonu, který je ihned (již v době zápisu) transpřekládán do JavaScriptu a současně i spouštěn.

Poznámka: povšimněte si, že všechny projekty zmíněné v předchozím textu jsou skutečně transpřekladači. Další variantou (kromě realizace transpřekladače) je interpretace zdrojového kódu Pythonu ve webovém prohlížeči s využitím interpretu psaného v JavaScriptu popř. použitím WebAssembly. To nás přivádí k dalšímu zajímavému projektu, který se jmenuje PyScript. Namísto transpřekladu do JavaScriptu se používá klasický (standardní) CPython přeložený do WebAssembly (WASM). Prakticky stejným způsobem se do WebAssembly může přeložit i MicroPython.

Další více či méně zajímavé a užitečné projekty, které se snaží Python přenést do webového prohlížeče, můžete najít na stránce <https://wiki.python.org/moin/WebBrowserProgramming>, popř. v tabulce umístěné na adrese <http://stromberg.dnsalias.org/~strombrg/pybrowser/python-browser.html>.

9.2 Transpřekladače do JavaScriptu

Jednu z dnes velmi populárních technik umožňujících použití prakticky libovolného programovacího jazyka pro tvorbu aplikací běžících na straně webového prohlížeče, představuje použití výše zmíněných transpřekladačů (taktéž source-to-source compiler) zajišťujících transformaci programu napsaného ve zdrojovém programovacím jazyce do funkčně identického programu reprezentovaného v JavaScriptu. Transpřekladač se většinou spouští jen jednou, a to například na vývojářském počítači. Samotní klienti, kteří k webové aplikaci přistupují, již mají k dispozici JavaScriptový kód, který je spouštěn v jejich prohlížeči. To jinými slovy znamená, že vývojář může použít jakýkoli vhodný programovací jazyk, zatímco uživatel si vystačí s běžným webovým prohlížečem podporujícím JavaScript (což jsou všechny mainstreamové webové prohlížeče).

Poznámka: ve skutečnosti není technologie transpřekladačů žádným způsobem svázána právě s JavaScriptem, protože se používala (a používá) i pro další manipulace se zdrojovými kódy.

9.2.1 Příklady transpřekladačů

Z praxe můžeme uvést například následující projekty založené na transpřekladači. Některé z nich je možné použít přímo v prohlížeči, jiné provádí překlad do JavaScriptu na příkazovém řádku, existují i kombinace obou způsobů (příkladem může být projekt WISP podporující oba režimy):

Jazyk	Poznámka
CoffeeScript	přidání syntaktického cukru do JavaScriptu
JSweet	překlad programů z Javy do JavaScriptu popř. do TypeScriptu
Transcrypt	překlad Pythonu do JavaScriptu
ClojureScript	překlad aplikací psaných v Clojure do JavaScriptu
Kaffeine	rozšíření JavaScriptu o nové vlastnosti
RedScript	jazyk inspirovaný Ruby
GorillaScript	další rozšíření JavaScriptu
ghcjs	transpřekladač pro fanoušky jazyka Haskell
wisp	zjednodušená, dnes již nevyvíjená varianta ClojureScriptu
Babel	překlad novějších variant JavaScript (ES2015) a TypeScriptu do starší verze JS
GopherJS	překladač programů naprogramovaných v jazyce Go do JavaScriptu
Fable	transpřekladač jazyka F# do JavaScriptu (přes Babel)
ScriptSharp	transpřekladač z jazyka C# do JavaScriptu
Dart	transpřekladač z jazyka Dart do JavaScriptu
Coconut	transpřekladač jazyka Coconut do Pythonu

Poznámka: seznam všech (či alespoň většiny) známých transpřekladačů do JavaScriptu naleznete například na stránce <https://github.com/jashkenas/coffeescript/wiki/List-of-languages-that-compile-to-JS>.

9.3 Brython

V této podkapitole se ve stručnosti seznámíme s projektem pojmenovaným *Brython*. Bude nás zajímat především to, jak se vlastně Brython (tedy vlastní transpřekladač z Pythonu do JavaScriptu) vloží do webové aplikace, jakým způsobem se inicializuje, kam a jakým způsobem se uloží zdrojové kódy front-endu psané v jazyce Python a v neposlední řadě je nutné se seznámit i se způsobem, jakým Brython pracuje s DOMem (*document object model*) webové stránky i se systémem událostí (*events*). Bez podpory těchto dvou nezbytných částí webové aplikace by totiž použití Brythonu pro