

Transparentní programování grafických procesorů v Groovy

Předmětem projektu je návrh a implementace knihovny pro transparentní programování moderních paralelních grafických procesorů (GPU) přes javovské OpenCL API (<http://www.khronos.org/ocl/>) v programovacím jazyce Groovy. Výsledkem projektu by mělo být rozšíření existující open-source knihovny *GPars*, které umožní využití paralelních schopností GPU přímo z Java a Groovy kódu, bez nutnosti programovat v C.

Kontext projektu

Groovy (<http://groovy.codehaus.org/>) je dynamický skriptovací objektově-orientovaný jazyk postavený nad JVM, jehož syntaxe vychází z Javy. V současné době patří mezi tři nejrozšířenější alternativní jazyky pro JVM (Scala, Groovy, Clojure) a zaměřuje se primárně na vývoj webových aplikací (Grails), tvorbu doménově-specifických jazyků a skriptování. Pro svoji syntaktickou blízkost k jazyku Java nabízí Groovy zkušeným Java programátorům bezbolestnou cestu k osvojení nových konceptů známých z jiných jazyků, jako je třeba funkcionální programování či meta-programování.

Open-source knihovna *GPars* (<http://gpars.codehaus.org/>) rozšiřuje Groovy o moderní koncepty concurrent programování, jako jsou actors, dataflow, paralelní kolekce, agenty a jiné, které dovolí Groovy a Java programům využívat schopnosti dnešních vícejádrových procesorů.

Moderní grafické karty jsou osazeny výkonnými GPU, jejichž doménou je masivní paralelní zpracování kolekcí číselných hodnot. OpenCL je otevřený standard pro multiplatformní programování těchto procesorů, které dovoluje Java kódu spouštět funkce k vykonání na GPU. Tyto funkce ovšem musí být naprogramovány v jazyce C a přeloženy C překladačem, což komplikuje vývoj aplikací v Javě.

Tento softwarový projekt by měl doplnit *GPars* o modul, který umožní programovat funkce pro GPU v Groovy a rovněž tyto funkce pohodlně z Groovy či Java kódu volat.

Projekt může navázat na dřívější prototypové práce *GPars* týmu, které využívaly Groovy AST transformace, knihovny *JOCL* (<http://www.jocl.org/>) a *APARAPI* (<https://code.google.com/p/aparapi/>).

Projektový tým

Vedoucí: Martin Děcký, MFF UK (decky@d3s.mff.cuni.cz)

Konzultant: Václav Pech, JetBrains (pech@d3s.mff.cuni.cz)

Řešitelský tým:

- TBD

Cíle projektu

- Návrh a implementace překladu Groovy kódu do jazyka C (primárně využitím Groovy AST transformace či knihovna *APARAPI*). K dispozici je funkční prototyp.
- Návrh a implementace API pro volání funkcí vykonávaných na GPU. Toto API by mělo navazovat na stávající Groovy/*GPars* API pro práci s kolekcemi.
- Výstup projektu bude začleněn do kódu knihovny *GPars* a uvolněn pod shodnou open-source licenci (Apache 2).
- Vzhledem k mezinárodní povaze projektu *GPars* a jeho komunity budou veškeré výstupy, tj. kód i dokumentace, provedeny v anglickém jazyce.

Analýza náročnosti a rizik

Projekt klade na řešitele především tyto nároky:

- Osvojení programovacího jazyka Groovy (včetně mechanismu AST transformací) a knihovny *GPars*.

[1 měsíc]

- Nastudování mechanismu fungování GPU a seznámení s OpenCL API. [1 měsíc]
- Navržení vhodné softwarové architektury modulu a jeho implementace. [4 měsíce]
- Začlenění modulu do knihovny GPars a sladění API. [1 měsíc]
- Dokumentace, sepsání uživatelského manuálu. [2 měsíce]

Groovy je rozšířený programovací jazyk podporovaný společností VMware a širokou komunitou přispěvatelů a uživatelů. Jeho funkcionality pro compile-time meta-programování je již dostatečně vyzrálá, čímž poskytuje dostatečně stabilní základ pro úspěšnou realizaci projektu.

GPars je ve své současné verzi 1.0 stabilní systém, který nepředstavuje rizika po stránce spolehlivosti ani dokumentační. Člen GPars týmu bude k dispozici pro konzultace během celého průběhu projektu.

OpenCL je standard podporovaný společnostmi jako Intel, Apple, AMD, ARM, IBM a dalšími.

Open-source knihovny JOCL a APARAPI procházejí bouřlivým vývojem a mohou představovat určité riziko ohledně stability a funkcionality. Toto bude nutno zohlednit při volbě technologií použitých během implementace projektu.

Lze říci, že celková náročnost a rizika projektu odpovídají očekávanému workloadu pro pětičlenný řešitelský tým a devítiměsíční dobu vypracování.