

Optimalizace uspořádání výpočetních uzlů pro potřeby distribuovaného počítání

Vedoucí projektu

Mgr. Daniel Toropila <daniel.toropila@ruk.cuni.cz>

Mgr. Jindřich Stejskal <jindrich.stejskal@ruk.cuni.cz>

Řešitelský kolektiv

4 nebo 5 studentů.

Termín dokončení

9 měsíců od zahájení projektu.

Anotace

Jedním z hlavních problémů při gridovém (distribuovaném) počítání je efektivní využití dostupných výpočetních zdrojů a optimální rozložení úloh. Častokrát se totiž ukazuje, že pro distribuovatelné úlohy bez ideální granularity nemá smysl zapojovat do jejího počítání víc než dopředu známý počet výpočetních jednotek (tento počet je specifický pro konkrétní úlohu). Má však smysl vybrat z dostupných počítačů takové, které budou schopné spolu rychle komunikovat, případně sdílet nějaká data. To ale nemusí být v případě **heterogenního síťového prostředí** (firewally, NAT) a omezených zdrojů jednoduchým úkolem.

Základní entitou projektu je **uzel**, což je program, který umí komunikovat po síti s ostatními uzly. Cílem projektu je vytvoření a **implementace algoritmů pro uspořádávání těchto uzlů do uživatelem zvolených virtuálních topologií (skupin) v reálném síťovém prostředí**. Struktura by měla být co neoptimálnější v závislosti na zvolené topologii, konkrétní situaci v síti, počtu dostupných uzlů a dalších požadovaných vlastnostech (propustnost dat, minimální konfigurace, atd.). Struktura se navíc musí **dynamicky přizpůsobovat změnám**, jako je například odpojení uzlu, připojení nového uzlu. Uzly musí být schopné komunikovat přes firewally, měly by mít schopnost detekce vzájemné viditelnosti v privátních sítích, atd..

Základem pro vytvoření takových algoritmů je zjištění co možná nejvíc informací o připojených počítačích. Důležitou součástí projektu je proto **automatická detekce reálné síťové topologie**, jejich vlastností a vzájemné viditelnosti uzlů. Na základě těchto znalostí pak může proběhnout dynamická organizace výpočetních jednotek do požadované virtuální struktury.

V rámci projektu bude též vytvořena grafická komponenta pro vizualizaci stávajících topologií a grafické uživatelské rozhraní pro správu topologie uzlů.

Další požadavky

- Platforma **Java**
- Schopnost integrace výsledného produktu do systému pro distribuované počítání DICOS vyvíjeného na Ústavu výpočetní techniky UK