CESNET

MetaCentrum

datové služby

Miroslav Ruda, Zdeněk Šustr

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

Agenda

- Národní gridová infrastruktura
  - přehled služeb MetaCentra
  - aktuální stav
  - výpočetní grid
  - cloudové prostředí
- MapReduce výpočty

Národní gridová infrastruktura

Distribuované prostředí pro sdílení výpočetních zdrojů

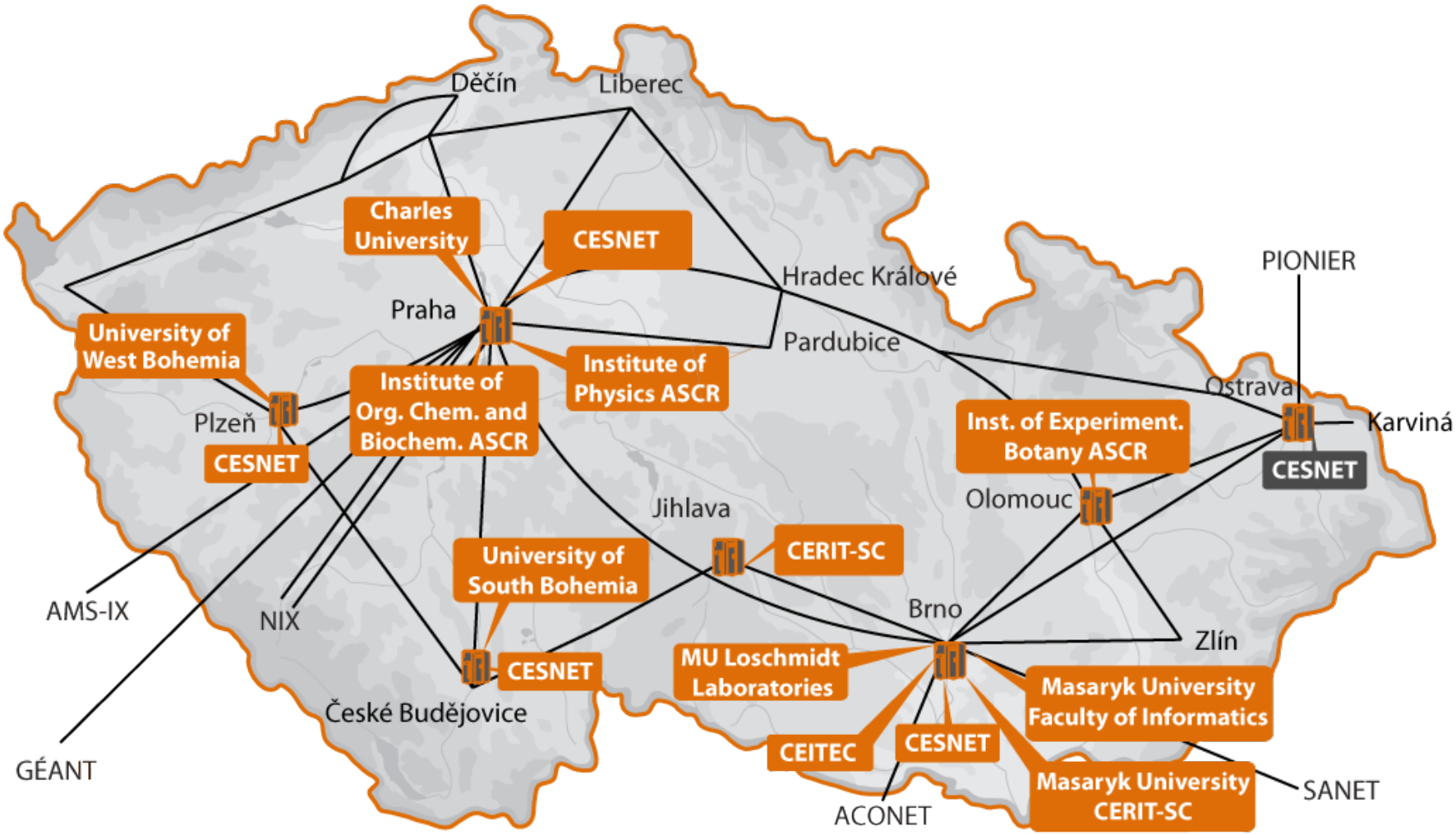
- motivací je optimální využití zdrojů (nejen hardwaru)
  - přenesení nárazové zátěže na volnější zdroje
  - a využití jiných zdrojů při výpadku
  - poskytnutí vlastních dočasně volných zdrojů
- možnost pořídit si cluster jen na standardní kapacitní požadavky
- spolupráce týmů z různých organizací, sdílení dat

MetaCentrum

- součást e-infrastruktury spravované CESNETem
  - jednotný účet, propojené služby
- pracuje v koordinaci s dalšími projekty IT4i a CERIT-SC
- přidáváme podporu zpracování rozsáhlých dat, nových typů výpočtů

MetaCentrum

- clustery a výkonné servery, úložné kapacity
  - Plzeň, Praha, Brno, Ostrava, ČB
- aktuálně 10 000 jader, 4 000 vlastní CESNET
  - clustery CEITEC, CERIT-SC, ZČU, JČU, UK, MU
- přes 1 PB diskových prostor na zpracování dat
- centrální správa uzlů, účtů, úloh, aplikačního softwaru
- dalších 4 000 jader a 2 PB přes FZU do EGI (LHC)
- "placení" formou publikací s poděkováním
  - přístup všem akademickým pracovníkům a studentům bez omezení, bez podávání projektů
  - publikace využívány pro určení priority uživatele
- výpočty gridové, cloudové, MapReduce


Děčín
Liberec
Charles University
CESNET
Hradec Králové
PIONIER
Praha
University of West Bohemia
Pardubice
Institute of Physics ASCR
Institute of Org. Chem. and Biochem. ASCR
Ostrava
Plzeň
Karviná
Inst. of Experiment. Botany ASCR
CESNET
CESNET
Olomouc
Jihlava
CERIT-SC
University of South Bohemia
AMS-IX
NIX
Brno
Zlín
MU Loschmidt Laboratories
CESNET
Masaryk University Faculty of Informatics
České Budějovice
CEITEC
CESNET
Masaryk University CERIT-SC
SANET
GÉANT
ACONET

Novinky

CESNET

- zprovozněna třetí generace GPU clusteru
  - 30 uzlů 2× NVIDIA Tesla K20
- tento rok přibude
  - 400 TB home v Brně
  - 4 SMP servery (každý 4×8 jader, 512 GB RAM)
  - 18 uzlů cloudu (každý 2×8 jader, 256 GB RAM)
  - Hadoop cluster (27 uzlů)

CERIT-SC

- druhý SGI UV v Brně
  - celkem 384 jader, 6 TB RAM

Připojeny clustery na ČVUT, FZU, uzel Elixiru na JČU

Výpočetní grid

- využívá většinu zdrojů, orientace na dávkové úlohy, hromadné zpracování, dlouhé vědecké výpočty
- různý typ výpočetních uzlů
  - klasické dvouprocesorové uzly orientované na HTC úlohy
  - uzly s GPU kartami (30 uzlů, každý 2× NVIDIA Kepler K20)
  - větší SMP uzly (4 procesory, 512 GB RAM)
  - 2× SGI UV2 (6 TB RAM každý)
- souborové systémy rozdělené podle využití uložených dat
  - lokální scratch disky pro dočasná data (HTC úlohy, <1 TB)
  - 3× sdílený scratch filesystém, viditelný přes jeden cluster – paralelní zpracování větších dat (10 TB)
  - home v každém městě – semipermanentní, zálohovaná data (100 TB)
  - projektové adresáře – semipermanentní, pro sdílení v rámci projektu
  - zpřístupněná data ze souborových serverů vlastníka clusteru
  - přímo zpřístupěné souborové systémy DU

Výpočetní cloud

- určeno pro vědecké výpočty nebo služby související s těmito výpočty
  - ne pro obecné hostování webových služeb
- výpočetní uzly s možností využití vlastního systému
  - uzly totožné s MetaCentrem – pro potřeby ladění softwaru
  - uzly dodané mezinárodní VO ve které jsou uživatelé zapojeni – dodaný obraz, předinstalovaný software
  - uzly s vlastní instalací systému a vlastního softwaru
- webové služby pro zadání úloh, permanentní část výpočetního prostředí
  - Galaxy portal nabízený MetaCentrem (autentizace prostředky MetaCentra, úlohy běží v gridu)
- uzly mohou být uzavřené do VLAN, mít pouze privátní IP adresy
- VLAN umožňuje i L2 sítě, propojení do domácí sítě, zahrnutí gridových uzlů

Map Reduce výpočty v MetaCentru

- nové prostředí, nabízené dočasně v cloudovém prostředí
- na konci tohoto roku bude provozováno na vlastním dedikovaném hardwaru
- opět integrováno s MetaCentem (přihláška, autentizace, accounting)
- Apache Hadoop prostředí (MapReduce, YARN, Hive, Pig, HBase, . . .)
- 3 servery sloužící jako front-end, metadata, management
- 24 výpočetních serverů (dual-socket, 128 GB RAM, 12×4 TB v každém uzlu)
- celková kapacita HDFS filesystému 1 PB

Map Reduce výpočty – motivace I

- MapReduce přístup publikován firmou Google (2003/2004)
- typ výpočtů rozpoznán jako klíčový i dalšími firmami s potřebou prohledávat velká nestrukturovaná data
  - logy (webových) serverů
  - textové soubory (viz náš příklad počítání slov)
  - XML/JSON soubory, dump databáze
  - bioinformatická data
- volně dostupná implementace HDFS a MapReduce pod hlavičkou organizace Apache (přispívají i firmy typy Facebook, Twitter, IBM)
- vznikají i formy poskytující komerční podporu (Cloudera, Hortonworks)
- postupně vznikají další nadstavby (Pig, HBase, Hive, YARN)

Map Reduce výpočty – motivace II

- motivace – tradiční servery nejsou schopné zpracovat data v potřebném rozsahu,
  - posilování serverů naráží na finanční limity,
  - nedostatek paměti
  - rychlost I/O na serveru
- nabízí se možnost data distribuovat přes sadu levnějších uzlů clusteru
  - zaručí se škálovatelnost nákladů
  - data se zpracovávají paralelně po částech
- potřeba koordinace, zaručení stability a konzistence při výpadku uzlu clusteru
- potřeba výpočetního prostředí ve kterém lze jednoduše takové úlohy zpracovávat

Map Reduce výpočty – HDFS

- filesystém pro uložení dat
  - data jsou při uložení do Hadoop clusteru rozdělena na menší kousky, ty jsou rozhozené přes jednotlivé uzly clusteru
  - soubory nelze jednoduše přepisovat, pouze prodlužovat
- při samotném výpočtu se data již nestěhují – úlohy se stěhují za daty
- systém sám zaručuje, že je uloženo několik kopií každého kusu dat
  - při výpadku uzlu sám zajistí dodělání chybějících kopií
- uzly mezi sebou komunikují minimálně, uživatel programuje v prostředí, kde neřeší výpadky, síťovou komunikaci ...
- pro nalití dat do HDFS existují i specializované nástroje
  - (Flume – logy, Sqoop – data z databází)

Map Reduce výpočty

- typická úloha se skládá z Map a Reduce fází, jazyk Java
- komunikace probíhá přes dvojice (klíč, hodnota)
- úloha pracuje nad jedním blokem dat
  - Master úlohy koordinuje, plánuje tam kde leží zpracovávaná data, ve vlastní režii řeší výpadky apod.
- Mapper zpravidla čte (jméno souboru,data), produkuje seznam (klíč, hodnota)
- Reduce proces dostane dvojice s jemu přiřazeným klíčem (setříděné podle klíče), produkuje výsledný (klíč, hodnota) do HDFS

```
Map(input-key,input-value)
  {foreach word w in input value; emit(w,1)}
Reduce(key, list of value)
  {foreach value sum=sum+value; emit(key,value)}
```

Hadoop ecosystem

- Hive – SQL-like interface pro dotazy nad Hadoop daty
- Pig – skriptovací jazyk pro tranformaci velkých dat v Hadoopu
- HBase – sloupcová databáze nad HDFSA (vybírá jen řádky, omezený jazyk pro dotazy, jeden klíč, vstup pro MapReduce)
- Flume – nástroj pro průběžné produkování dat
- YARN – Haddop 2.0, workflow z MapReduce úloh

```
A = load './input.txt';
B = foreach A~generate flatten(TOKENIZE((charar)$0)) as word;
C = group B by word;
D = foreach C generate COUNT(B), group;
store D into './wordcount';
```

Děkuji za pozornost

http://www.metacentrum.cz