ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE
PROVOZNĚ EKONOMICKÁ FAKULTA

Obor Systémové Inženýrství

BAKALÁŘSKÁ PRÁCE

Téma: REALIZACE VÝPOČETNÍHO CLUSTERU

Vypracoval: Jan Černý
Vedoucí bakalářské práce: Ing. Richard Černý, CSc.

Praha 2007 ©

Čestné prohlášení

Prohlašuji, že bakalářskou práci na téma „Realizace výpočetního clusteru“ jsem vypracoval samostatně za použití uvedené literatury.

Poděkování

Tímto děkuji panu Ing. Richardu Černému, CSc. za odborné vedení a připomínky k bakalářské práci. Déle chci poděkovat společnosti SPRINX Systems, a.s. za možnost podílet se na projektu superpočítače Amálka a za poskytnutí potřebného hardware.

Souhrn:

Tato práce se věnuje rozvíjejícímu se odvětví z oblasti výpočetní techniky, výpočetním clusterům.

V úvodu práce jsou vysvětleny jednotlivé typy výpočetních clusterů. V dalších kapitolách jsou podrobněji rozebrány důležité součásti clusteru. Nejprve se jedná o části hardwarové, dále pak o softwarové. Z hardwarových částí se jedná o procesory, paměti, sítě a disková pole. Ze softwarové části jsou to pak operační systémy a programové vybavení potřebné k paralelním výpočtům.

Praktická část práce se zabývá instalací výpočetního clusteru. Instalace probíhá od instalace a konfigurace operačního systému po instalaci a konfiguraci programu pro distribuci úkolů.

Klíčová slova:

výpočetní cluster
vysoce výkonný cluster
paralelní výpočty
Linux
HPC

Summary:

This work is about sector of computer technology, high performance computers.

At the beginning of this work are explained types of computer clusters. In next chapters are conquest explained important parts of clusters. First are parts from hardware, next from software. Hardware parts are processors, memory, ethernet and disk arrays. From the software it is operation systems and programs, which are neccessary to parallel computing.

The practical part of this work is installation high performance cluster. Installation is running from installation operation system to installation and configuration program to distribution tasks.

Keywords:

high performance cluster

parallel computing

Linux

HPC

1 Úvod

Použití počítačů při složitých numerických výpočtech je dnes samozřejmostí. Počítače dokáží vykonat miliony operací za vteřinu. Pro některé velmi složité výpočty to však není dostačující. Na jednom počítači by tyto složité výpočty trvaly i několik let, proto se stále více používá paralelizace výpočtů.

Paralelizace výpočtu spočívá v rozdělení výpočtu na mnoho malých úloh. Tyto úlohy se rozdělují mezi více výpočetních jednotek, čímž se celý výpočet urychlí. K řešení nejsložitějších úloh pomocí paralelizace existují velká seskupení počítačů. Takováto seskupení se nazývají výpočetními clustery[1].

Zavádění výpočetních clusterů v České republice není ještě příliš běžné, oproti tomu v zahraničí jsou výpočetní clustery využívány velkými průmyslovými, výzkumnými a jinými společnostmi. Proto tuto práci věnuji problematice výpočetních clusterů.

V první kapitole se zaměřím na teorii clusterů, jejich druhy a vlastnosti. V následujících kapitolách podrobněji rozeberu z hlediska výkonu nejdůležitější části výpočetního clusteru. Postupně se budu věnovat hardwaru a softwaru. Z hardwarové části se blíže zaměřím na procesory, paměti, síť a disková pole.

Následně vytvořím praktický příklad instalace výpočetního clusteru na dvou konfiguracích. Na závěr představím praktické použití výpočetního clusteru.

[1] Též nazývané vysoce výkonné clustery

2 Cíl práce a metodika

Cílem této práce je sestavení fungujícího výpočetního clusteru. Jednotlivé kroky instalace budou zaznamenány a podrobněji vysvětleny.

K sestavení výpočetního clusteru použiji dvě testovací konfigurace. První testovací konfigurace slouží jako příklad domácího řešení výpočetního clusteru. Hlavní počítač (Master) je notebook, funkci uzlu (Nod) zastává starší počítač. Druhá testovací konfigurace simuluje reálný provoz clusteru. Na ní je potřeba ověřit plnou funkčnost jádra 2.6.X. Tato konfigurace je složena z několika výkonných serverů.

Pro co největší dostupnost uživatelům bude instalace probíhat na operačním systému Slackware Linux, který je volně k dispozici na webových stránkách. Tento operační systém je velmi přehledný a má podrobnou dokumentaci. Po nainstalování operačního systému provedu jeho konfiguraci. Systém je třeba nastavit pro použití jako hlavní počítač (Master). Bude nutné vytvořit novou kompilaci jádra. Toto jádro bude použito pro uzly. Pro distribuci úkolů použijeme projekt openMPI. Tento program nainstaluji a provedu testovací úlohu. Na závěr představím výpočetní cluster Amálka sídlící v Ústavu fyziky atmosféry akademie věd.

3 Teorie clusterů - Literární rešerše

V této kapitole se věnuji základnímu rozdělení clusterů. Clustery mají základní rozdělení podle použití. V dalších kapitolách se budu více věnovat výpočetním clusterů, zejména jejich částmi (hardware, software).

Co je to cluster?

Cluster je seskupení počítačů, které mezi sebou spolupracují a navenek se tváří jako jeden počítač.[9]

3.1 Počítačové clustery

3.1.1 Vysoce dostupné clustery HA (High-availability)

Vysoce dostupné clustery jsou zkonstruovány k nepřetržitému provozu nějaké služby. Jsou to dva nebo více počítačů, které se vzájemně zálohují. Při výpadku jednoho počítače, kde běží služba, se systém automaticky přepne na další. Mezi nejčastější aplikace patří: Web servery, mail servery, firewally, souborové servery, DNS a DHCP servery. Počítače jsou vzájemně propojeny sériovým kabelem (Heartbeat), který kontroluje provoz počítače a při jeho výpadku přepne na jiný. Heartbeat může být jen proces, který kontroluje, sleduje provoz počítače a posílá údaje po síti.

3.1.2 Clustery pro vyvážení zátěže (Load-balancing)

Tyto clustery se používají ke zvýšení kapacity serveru. Je vytvořen jeden virtuální server, pod který spadá celá řada serverů. Tyto servery si mezi sebou rozdělují zatížení. Používají se jako Firewally, cache, **IDS**[2].

[2] IDS (intrusion detection systems) - systém který detekuje nežádané manipulace se systémem přes internet. Chrání tak před útoky virů, trojských koňů, crackerů.

3.1.3 Vysoce výkonné clustery HPC (High-performance clusters)

Tyto clustery jsou navržené pro nejnáročnější výpočty. Jedná se o seskupení mnoha počítačů, které společně počítají zadané úlohy. Úloha je rozdělena do mnoha menších částí a ty jsou následně přiděleny každému jednotlivému počítači. Jedná se o paralelní výpočty, kdy každý počítač má za úkol vypočítat nějakou část.

3.2 Vysoce výkonné clustery (HPC)[3]

Na začátku této kapitoly vysvětlím některé pojmy, které se v oblasti výpočetních clusterů běžně používají.

Master (Job manager)

Master je nejdůležitější součást celého clusteru. Tento počítač má za úkol rozdělovat úlohy na jednotlivé části, které přiděluje uzlům. Na tomto počítači může být nainstalován Job Scheduler. Master se taká nazývá *Hlavní počítač.*

Uzel (NOD)

Uzel bývá u výpočetních clusterů bezdisková stanice, která je připojena do sítě, přijímá a počítá úkoly od hlavního počítače.

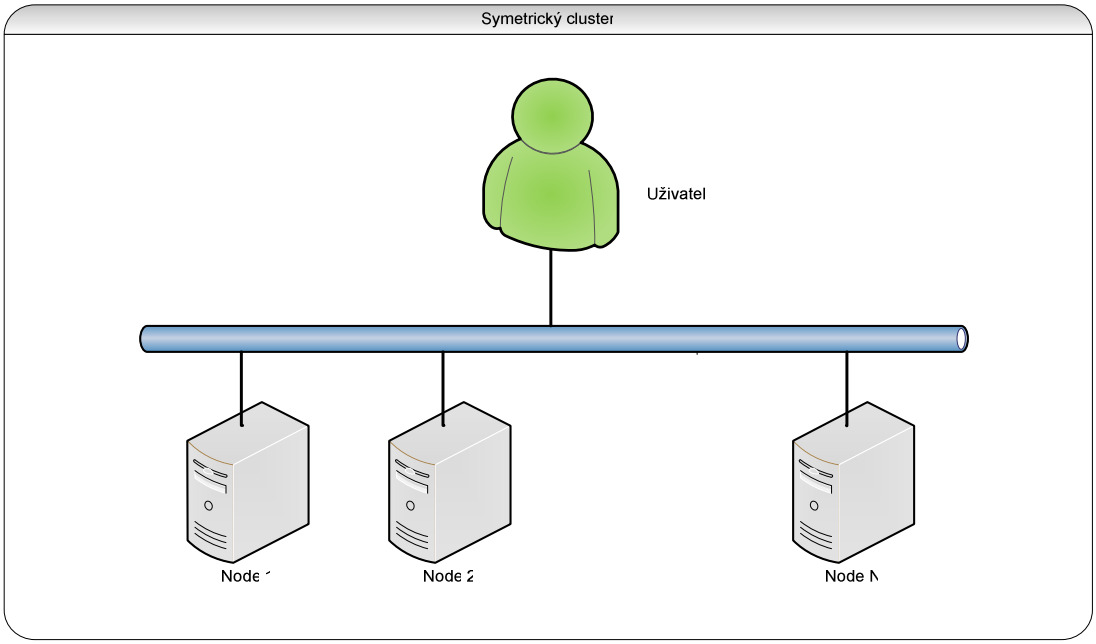
Paralelní počítání (Parallel computing)

[3] Podle Joseph D. Sloan. High Performance Linux Clusters. O`Reilly, 2004. 360s. ISBN: 0-596-00570-9.

Jedná se o počítání jednoho úkolu, úlohy na více procesorech, tak aby výsledek byl vypočítán rychleji než na jednom procesoru. Myšlenka tedy je rozdělit úlohu na co nejvíce malých podúloh, které se přidělí každému procesoru.

3.2.1 Symetrický cluster

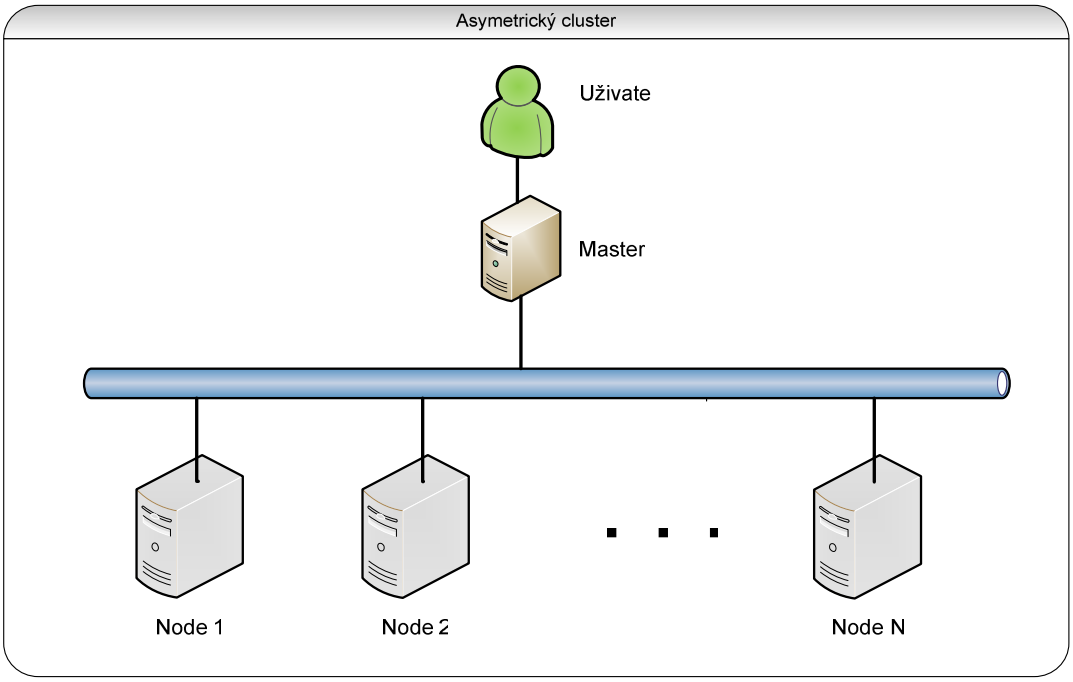
U Symetrického clustery fungují uzly jako jednotlivé počítače. Všechny počítače musí být nezávisle použitelné, je zde obtížnější řízení jednotlivých stanic a těžší udržení bezpečnosti.



Obrázek č.1 Symetrický cluster

3.2.2 Asymetrický cluster

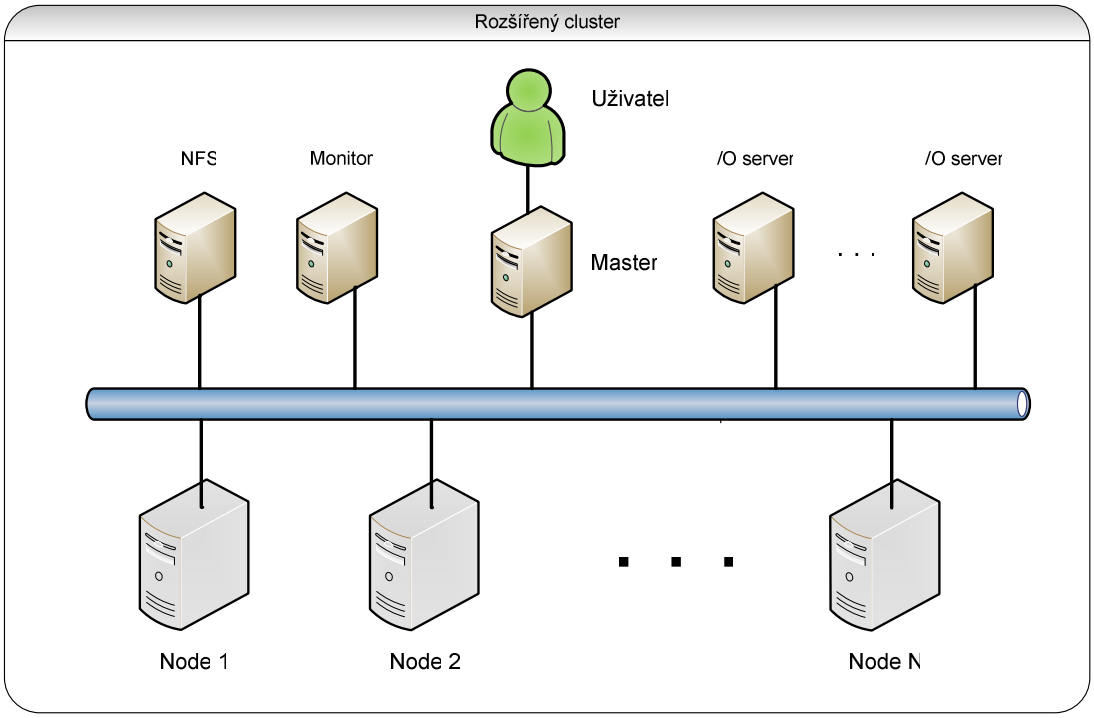
Asymetrický cluster má častější použití než-li symetrický cluster, hlavně z důvodů jednodušší správy všech uzlů. Není potřeba nastavovat všechny počítače zvlášť. Nevýhodou tohoto typu výpočetního clusteru je vysoká zátěž na hlavním počítači.



Obrázek č. 2 Asymetrický cluster

3.2.3 Rozšířený cluster

Tento cluster má rozložené zatížení na více strojích. Master obstarává jen rozdělování úloh, takže není téměř zatížen. Všechna potřebná data k výpočtů se získávají z diskových polí, nebo souborových serverů. Tato varianta clusteru může mít mnoho podob, záleží na konkrétním návrhu clusteru. Nevýhodou tohoto řešení jsou vysoké náklady na realizaci.



Obrázek č.3 Rozšířený cluster

3.3 Hardware

V této kapitole se podrobněji podíváme na důležité části výpočetního clusteru. Tyto části jsou důležité pro rychlé výpočty. Potupně se podíváme na procesory, paměti, disková pole a síť.

3.3.1 Procesory (CPU – central processing unit)

Procesory jsou výkonnou jednotkou počítače, která je nejdůležitější pro rychlost výpočtů. Jeden procesor není dostatečně rychlý, proto se používají stanice s více procesory.

Stanice s jedním procesorem

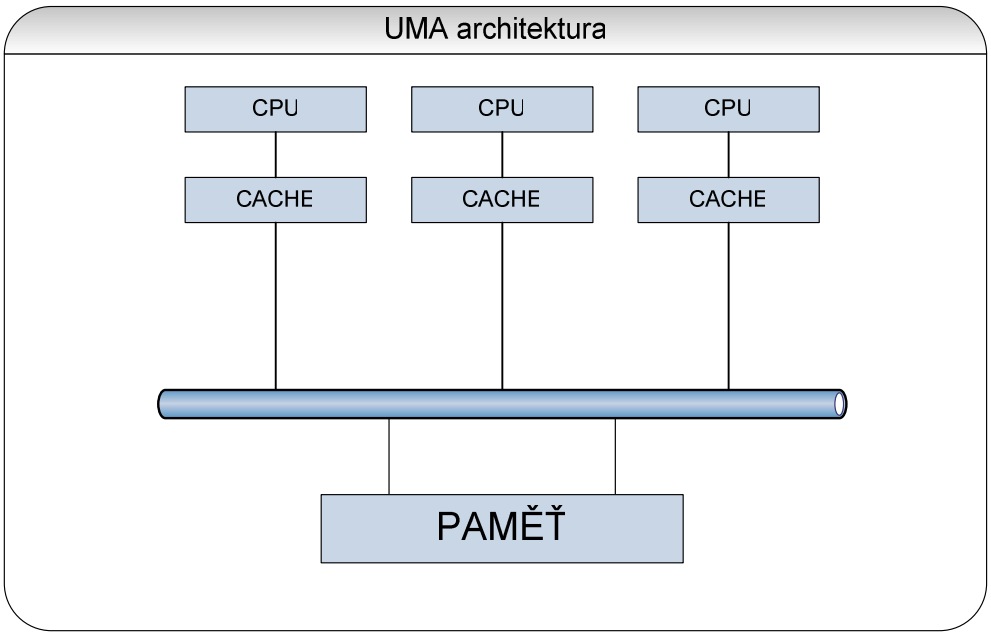
Stanice s jedním procesorem je tradiční architektura počítačů. Jedná se o Von Neumannovu architekturu, kdy se počítač skládá z pěti jednotek: řídící jednotka, aritmeticko-logická jednotka, paměť, vstupní zařízení, výstupní zařízení. Velmi záleží na rychlosti procesoru (kolik instrukcí může vykonat za jeden hodinový cyklus).

Víceprocesorové stanice

Víceprocesorové stanice se rozdělují na dvě základní kategorie. Na architekturu s jednotným přístupem k paměti (*uniform memory access (UMA))* a na architekturu s nejednotným přístupem k paměti *(nonuniform memory access (NUMA)).*

UMA architektura [9]

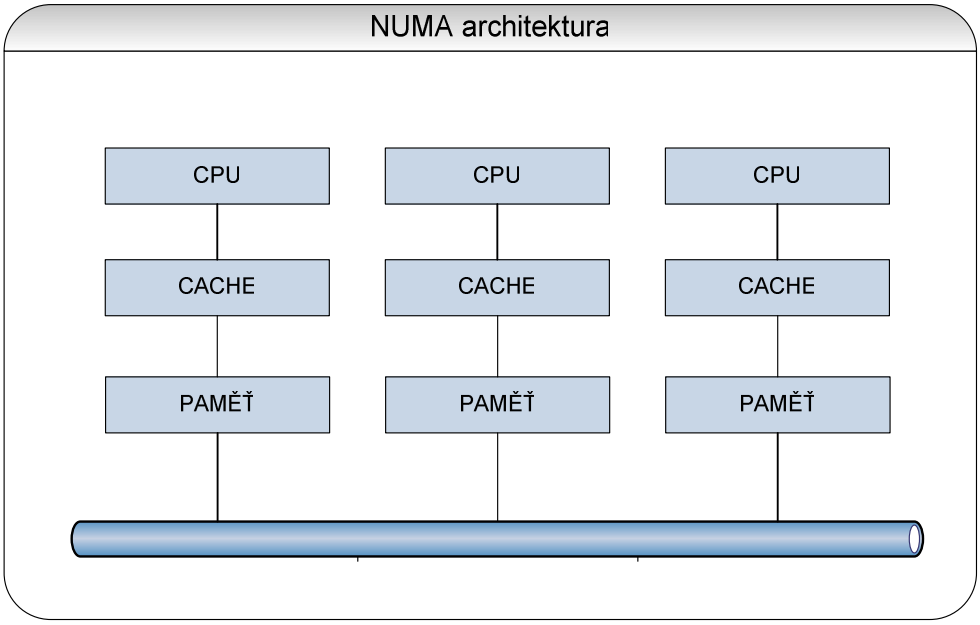
U UMA architektury jsou největší problémy se synchronizací všech CPU a správou paměti. Je nutné zamezit, aby dva procesory měli přístup do stejného paměťového místa.



Obrázek č. 6 UMA architektura

NUMA architektura [9]

U NUMA architektury mají všechny procesory vlastní paměť. Není zde problém s přístupem do paměti, každý procesor má přístup do celé paměti.



Obrázek č. 7 NUMA architektura

V dnešní době, kdy jsou více-jádrové procesory, se kombinují obě architektury. Jádra mají přístup do stejné paměti, která je rozdělována paměťovým řadičem (UMA architektura). Celý počítač je pak zapojen podle NUMA architektury.

Samozřejmě nemůžeme navyšovat počet procesorů do nekonečna. Od určitého počtu procesorů už neušetříme čas. Více o tom napoví Amdhalův zákon [9], který říká, že navyšovaní procesorů a zkrácení výpočtu není lineární. To znamená, že s navýšením počtu procesorů desetinásobně se výpočetní čas tolikrát nezkrátí.

Výrobci procesorů

V současné době máme několik významných zastupitelů, kteří dodávají procesory pro vysoce výkonné clustery. V následující tabulce je uvedeno jak jsou jednotlivé třídy zastoupeny mezi pěti sty největšími clustery na světě.

Tabulka č.1 Podíl procesorů mezi největšími výpočetními clustery [10]

třída procesorů	počet clusterů	procentuální zastoupení	počet procesorů
Power	**91**	**18.20 %**	**416492**
Cray	4	0.80 %	2538
Alpha	3	0.60 %	13768
PA-RISC	20	4.00 %	30708
Intel IA-32	**120**	**24.00 %**	**131962**
NEC	3	0.60 %	5888
Sparc	3	0.60 %	5440
Intel IA-64	35	7.00 %	60862
Intel EM64T	**108**	**21.60 %**	**123242**
AMD x86_64	**113**	**22.60 %**	**230061**

V tabulce je vidět, že největší zastoupení má společnost Intel, kde tvoří největší podíl Intel IA-32. Toto postavení je získáno dlouhodobou výrobou procesorů speciálně pro servery. Dalším tradičním výrobcem serverových procesorů je společnost IBM se svým procesorem Power. IBM je největším dodavatelem výpočetních clusterů na světě. Společnost IBM vlastní největší počítač **BlueGene**[4]. Od poloviny roku 2003 začala hrát významnou roli i společnost AMD. Tento výrobce vyrobil významnou třídu procesorů pro servery Opteron, které se velmi hodí i do výpočetních clusterů. Jejich obliba přetrvává, i když nyní je společnost AMD v mírném útlumu (2006-2007), který je vyvolán protahujícím se přechodem na novou výrobní technologii.

3.3.2 Paměti

V oblasti výpočetních clusterů je důležité použít takové paměti, které budou dostatečně zásobovat jednotlivé procesory daty. Se zvyšující se rychlostí procesorů rostou i nároky na operační paměť. Do výpočetních clusterů osazujeme paměti podle specifikací jednotlivých komponent (CPU, základní deska). Největší zastoupení mají **DDR SDRAM** (double date rate) moduly (datová propustnost do 5GB/s), které jsou stále více vytlačovány **DDR2 SDRAM** moduly, s teoretickou datovou propustností

[4] BlueGene je schopen vytváří výkon 280.6 teraFLOPS (počet operací s plovoucí čárkou za sekundu). Je tvořen 131072 procesory.

8-9,5 GB/s. V budoucnu je očekáván nástup **DDR3 SDRAM** a **DDR4 SDRAM** modulů s datovou propustností větší než 10GB/s.

3.3.3 Disková pole

Diskové pole slouží k uchování a zálohování dat, která potřebujeme k výpočtům. V současné době máme mnoho možností jak toto pole vytvořit:

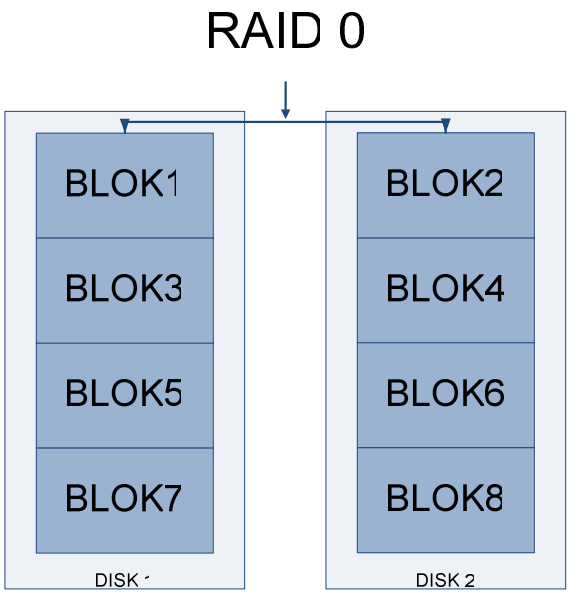
Master počítač s diskovým polem

Master počítač s dikovým polem je jedna z nelevnějších variant. Master (hlavní počítač) bude v sobě obsahovat dostatečný počet pevných disků, které budou spojeny do určitého pole **RAID (**Redundant Array of Independent Disks). Master bude potřebovat diskový řadič, který bude pole RAID podporovat. V Linuxu je možnost vytvořit „sofwarový raid“, bývá stejně rychlý jako pole vytvořené řadičem disků. Při použití disků nejsme omezeni na jeden typ disků, můžeme použít SATA, PATA, SCSI apod. Při výpočtech nebude toto pole příliš zatěžováno. Největší zátěž je při ukládání výsledků a výpočtů. Při použití rychlých disků se zkrátí doba mezi jednotlivými výpočty.

Nejpoužívanější typy polí u výpočetních clusterů:

Diskové pole typu RAID 0 (Stripping)

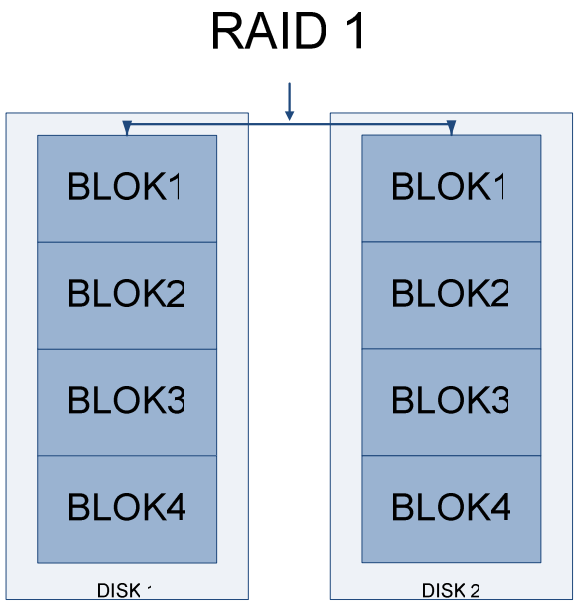
RAID 0 je seskupení disků, kdy se soubory rozdělují na jednotlivé části (bloky). Ty jsou postupně ukládány střídavě na jeden a druhý disk. Při poruše jednoho disku přijdeme o všechna data.



Obrázek č.8 RAID 0

Diskové pole typu RAID 1 (Mirroring)

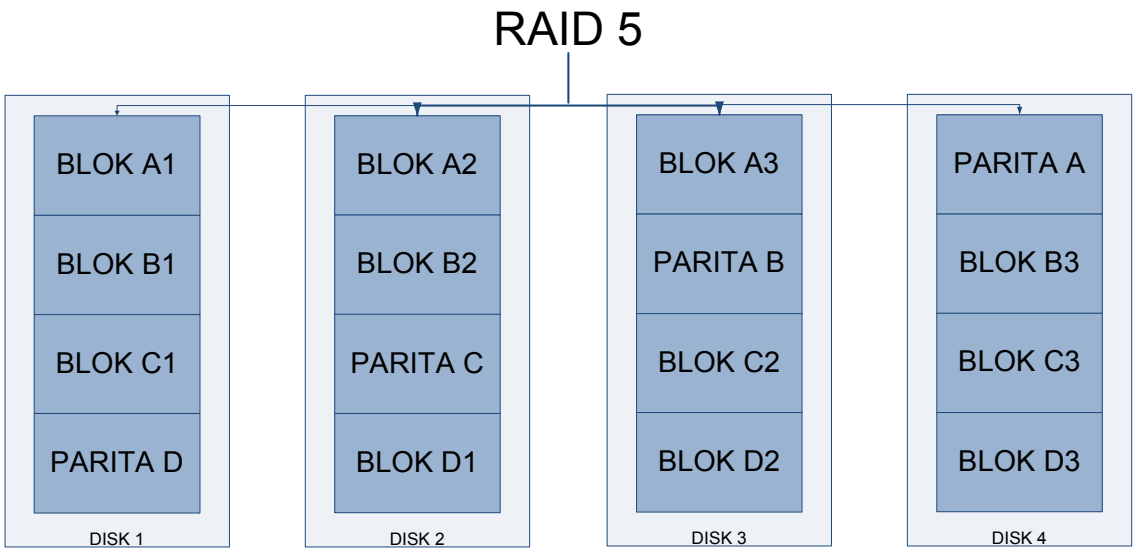
RAID 1 je seskupení disků tzv. zrcadlení. Data jsou ukládána paralelně na oba disky. Při poruše jednoho disku máme totožná data na druhém.



Obrázek č.9 RAID 1

Dikové pole typu RAID 5

U pole typu RAID 5 je použito N stejných disků (více jak 3), každému disku se počítá parita, která se ukládá střídavě na všechny disky.



Obrázek č. 10 RAID 5

Při použití externího diskového pole můžeme požít podobná zařízení od různých výrobců. Většinou se jedná o tzv. **SAN/NAS** (Storage Area Network/ Network-attached storage) disková pole. Například **Fibre Channel**, který je v současné době asi nejvíce používán, **iSCSI**, **HyperSCSI**, **ATA over ETHERNET, Infiniband**. Tato pole, mohou být připojena k hlavnímu počítači, nahrazují interní disková pole nebo mohou být připojena přímo do sítě.

3.3.4 Síť (Ethernet)

Důležitým prvkem k funkčnosti výpočetního clusteru je síť. Při konstrukci musíme vědět jestli bude síť dostatečně rychlá a nebude tak brzdit výpočty. V dnešní době máme k dispozici 100Mbit, 1Gbit a popřípadě 10Gbit sítě ethernet. Při konstrukci velkých výpočetních clusterů je dobré přemýšlet o síti optické nebo **QsNet, Emulex, cLAN**, **Infiniband**[5]**.**

[5] Infiniband, QsNET, cLAN - sítě používané především u výpočetních clusterů, kde je vysoká datová propustnost v řádově desítkách Gbit

3.4 Software

3.4.1 Operační systémy

Linux

Linux je nejrozšířenější operační systém pro výpočetní clustery, Je to především pro jeho univerzálnost, kompatibilitu a kompaktní velikost. Existuje mnoho výrobců, kteří dodávají kompletní řešení se systémem pro clustery (např. Redhat, Suse).

MS Windows

Ve světě velkých výpočetních clusterů se operační systém Windows téměř nepoužívá. Byla vydána edice určená přímo pro clustery. Společnost Microsoft tento systém stále vyvíjí. Operační systém Windows se uplatňuje především pro malé výpočetní clustery.

SUN Solaris

Solaris je operační systém od společnosti SUN microsystems. Systém prošel několika fázemi vývoje a dnes je k dispozici ve verzi 10, která se používá i pro výpočetní clustery.

Ostatní UNIX

Některé společnosti vyvíjejí vlastní systém, na kterém provozují výpočetní clustery. Většinou vycházejí z Unix. Největší takové společnosti jsou IBM (AIX), a HP (HP-Unix).

3.4.2 Programové vybavení

V dnešní době máme stovky programovacích jazyků, ale jen několik z nich můžeme použít v oblasti výpočetních clusterů. Jedná se především o FORTRAN nebo C/C++.

FORTRAN je základ výpočetních clusterů, aplikace se bez něho obejdou, ale existujících aplikací na FORTRANU je velmi mnoho, a proto se stále tento programovací jazyk používá. FORTRANU je více verzí, pohybují se mezi verzemi FORTRA 77 až FORTRAN 90. FORTRAN 77 spadá pod licenci GNU, kdežto FOTRAN 90 placený.

C/C++ je samozřejmá alternativa k Fortran. Programovací jazyk C je pravděpodobně nejlepší volbou. Má kompatibilitu s knihovnami a je k dispozici mnohem více programátorů, kteří pracují v C než ve FORTRANU.

Pokud bychom chtěli zvolit jiný programovací máme ve většině případů smůlu. Knihovny pro paralelní výpočty nejsou přenositelné na jiné jazyky. Na vývoji těchto knihoven se pracuje pro jazyky Java a Python.

Máme na výběr několik knihoven pro paralelní počítání. Jsou k dispozici knihovny Parallel Virtual Machine (PVM) a Message Passing Interface (MPI). Knihovna MPI je novější standart, který je preferován většinou uživatelů.

LAM/MPI (LAM Message Passing Interface) [11]

Tato sada API nástrojů nám umožňuje psát programy, které jsou přizpůsobeny paralelním výpočtům. MPI je vhodné pro superpočítače (např. velké clustery IBM SP, SGI Origin), ale samozřejmě funguje i na mnohem menších paralelních počítačích. MPI je navrženo tak, aby bylo nezávislé na použitém softwaru (Operačním Systému). K dispozici máme zdrojové kódy, které můžeme zkompilovat pro jakýkoliv myslitelný

systém. LAM/MPI je vyvíjeno na univerzitě v Indianě, ale na vývoji různých částí se může podílet téměř každý, protože tento projekt spadá pod přepracovanou licenci BSD[7].

openMPI (Massage Passing Interface) [1]

OpenMPI je sada knihoven a nástrojů pro paralelní počítání, které spadají pod open source[6] licenci založené na licenci BSD[7]. Podporuje mnoho operačních systémů. Je vyvíjeny sdružením akademických, výzkumných a průmyslových partnerů.

MPICH [9]

MPICH (Massage passing interface chameleon) byl vyvinut Williamem Groppem a Swingem Luskem stejně jako LAM/MPI je to sada knihoven a nástrojů pro paralelní výpočty. Program běží na většině Unix platforem a je dostupný i pro Windows.

AFS

AFS je distribuovaný souborový systém, který nabízí klient-server architekturu sdílení souborů.

MATLAB

Matlab je programové prostředí se skriptovacím programovacím jazykem, které slouží k vědeckotechnickým numerickým výpočtům, modelování , návrhu algoritmů, analýze a prezentaci dat, měření a zpracování signálů. Na toto prostředí existuje mnoho přádaných modulů z nichž nejznámější je Simulink, který slouží k modelování dynamických systémů. Program lze použít jak na Uniových systémech tak na Windows.

---

[6] Open source je název pro software s otevřeným zdrojovým kódem. Otevřený znamená, že máme přístup ke zdrojovému kódu a za jistých podmínek můžeme tento kód využívat a upravovat.
[7] Licence BSD je jednou z nejotevřenějších licencí, umožňuje. Umožňuje volné šíření licencovaného obsahu, vyžaduje uvedení autora.

Následující programy jsou důležité pro chod výpočetního clusteru:

SSH (Secure Shell) [2]

SSH je protokol, který umožňuje přenos dat po síti v šifrované podobě mezi dvěma počítači. SSH server naslouchá většinou na portu 22. K šifrování používá veřejný klíč. Nejčastěji se požívá ke vzdálenému přístupu k příkazové řádce.

DHCP (Dynamic Host Configuration Protocol) [2]

DHCP je protokol, který na základě nastavení přiděluje síťové IP adresy. Vyskytuje se téměř ve všech operačních systémech.

Job scheduler

Job scheduler je probram rozdělující jednotlivé úkoly podle zadaných kriterií. S použitím tohoto programu můžeme docílit lepší využití výpočetního výkonu. Můžeme přidělit jednomu uzlu více úkolů. Tento uzel nemusí na každý úkol používat 100% výkonu. Nejznámějšími zastupiteli jsou PBSpro, Condor, Maui job Scheduler, IBM load balancer.

DNS (domain name system) [2]

DNS je systém doménových jmen. Primární úlohou je vzájemné nahrazování číselných IP adres uzlů sítě a jejich jmen. U výpočetních clusterů se používá pro snadný přístup k jednotlivým uzlům. DNS server bývá instalován na hlavním počítači.

4. Praktické řešení

V této části bych chtěl předvést praktickou instalaci výpočetního clusteru. Nejprve nainstaluji operační systém Linux, následně provedu jeho konfiguraci. Po dokončení konfigurace nainstaluji program pro paralelizaci výpočtů openMPI.

4.1 Minimální konfigurace

Pro instalaci clusteru je potřeba nejprve vybrat vhodnou konfiguraci počítače. Minimální požadavky pro Slackware linux z ISO souborů jsou:

- 486 procesor
- 16MB RAM (32MB doparučeno)
- 100-500 megabytes místa na disku pro minimální a okolo 3.5GB pro plnou instalaci
- CD-ROM

Platí v případě, že nechceme požívat grafické rozhraní.

Pro tento projekt, kdy slouží Linux jako Master potřebujeme trochu jinou konfiguraci. Rychlost procesoru není velkým omezujícím prostředkem, procesor o rychlosti 600MHz a rychlejší bude stačit. U paměti bych doporučoval 256MB RAM a více. Disk postačí jeden o kapacitě 10G, záleží na množství dat, která budeme potřebovat k výpočtům. Rozhodně se vyplatí investovat do diskového pole o dvou a více discích. Budeme potřebovat CD mechaniku (nebo DVD), a síťovou (ethernetovou) kartu. Zde můžu zvolit 100Mbit kartu nebo 1Gbit kartu. Pokud použiji 1G kartu musím použít i 1G síťové prvky (switche). Největší přenos dat po síti nastává při bootovaní uzlů, načítání a ukládání dat k výpočtům.

Pro jednotlivé uzly je to trochu jiné. Můžeme říci, že čím rychlejší počítač, tím lepší. Uzel poběží i na pomalejším počítači, ale poté můžeme spekulovat jestli se vyplatí výpočetní cluster realizovat či není lepší rovnou pořídit jeden výkonný počítač. Velikost operační paměti má též vliv na celkovou rychlost uzlu. Pro každý procesor existuje optimální množství paměti, které je velmi těžce zjistitelné. Proto je vhodné použít

1 a více Gigabajtů. Zapotřebí je síťová karta (100Mbit, 1Gbit), která podporuje bootování po síti.

Testovací konfigurace I

Master: IBM T21; 256MB RAM; 40G disk

NOD: 1x Intel celeron 800, 128MB RAM

Síť: 100MBit

Testovací konfigurace II

Master: 4x intel pentium III 750MHz, SCSI disky 2x 8GB, 1G RAM, 1Gbit síť

2x NOD: 2x INTEL XEON 2,8GHz, 2GB RAM, 1Gbit síť

4.2 Rozdělení disku

Toto je testovací projekt, proto použiji jeden pevný disk. V reálném provozu by bylo nutné použít diskové pole pro větší rychlost a pro větší bezpečnost dat.

4.3 Zabezpečení sítě

Pokud bychom chtěli mít přístup k výpočetnímu clusteru zvenku prostřednictvím internetu, musel bych zabezpečit, aby se na správu clusteru nedostal někdo nepovolaný. V takovém případě je nutné nakonfigurovat firewall (iptables) a vzdálený přístup k příkazové řádce (SSH). V tomto projektu se jedná o cluster, který je tvořen odděleně od celé sítě, a proto konfiguraci těchto programů směrem ven vynecháme.

4.4 Rozdělení kořenového adresáře (/)[2]

V této kapitole jsem uvedl rozdělení kořenového adresáře operačního systému Slackware Linux, protože později budeme s těmito adresáři pracovat a vytvářet souborový sytém pro uzly.

Bin – jsou zde uživatelské programy. Je zde potřebné minimum k tomu, aby uživatel mohl používat systém. Jsou to například povely pro příkazovou řádku (ls, cp..)

/Boot – soubory které jsou potřeba pro bootvacího manažera LILO, jsou zde uložena jádra která můžeme bootavat. Pro přidání jádra do manažera musíme nakonfigurovat */etc/lilo.conf* a poté provést příkaz *# lilo.*

/Dev – adresář obsahující zařízení reprezentovaná souborem jako jsou například pevné disky, sériové porty..

/Etc – tento adresář obsahuje konfigurační soubory

/Home – adresář osahující data jednotlivých uživatelů

/Lib – systémové knihovny, které jsou potřebné pro základní operace (moduly, knihovny C)

/Mnt – adresář obsahující dočasné připojení k záznamovým zařízením (disky, odpojitelné disky,)

/Opt – (Optional software packages), jsou zde nainstalované přídavné balíčky

/Proc – je to jakási forma virtuálního filesystému, který zajišťuje přístup k informacím jádra. Například informace o procesoru.

/Root – adresář superuživatele

/Sbin – programy které běží pod superuživatelem a běžící během bootování. Normální uživatelé nemají práva spouštět

/Tmp – dočasné úložiště, všichni uživatelé mají práva číst i zapisovat do tohoto adresáře

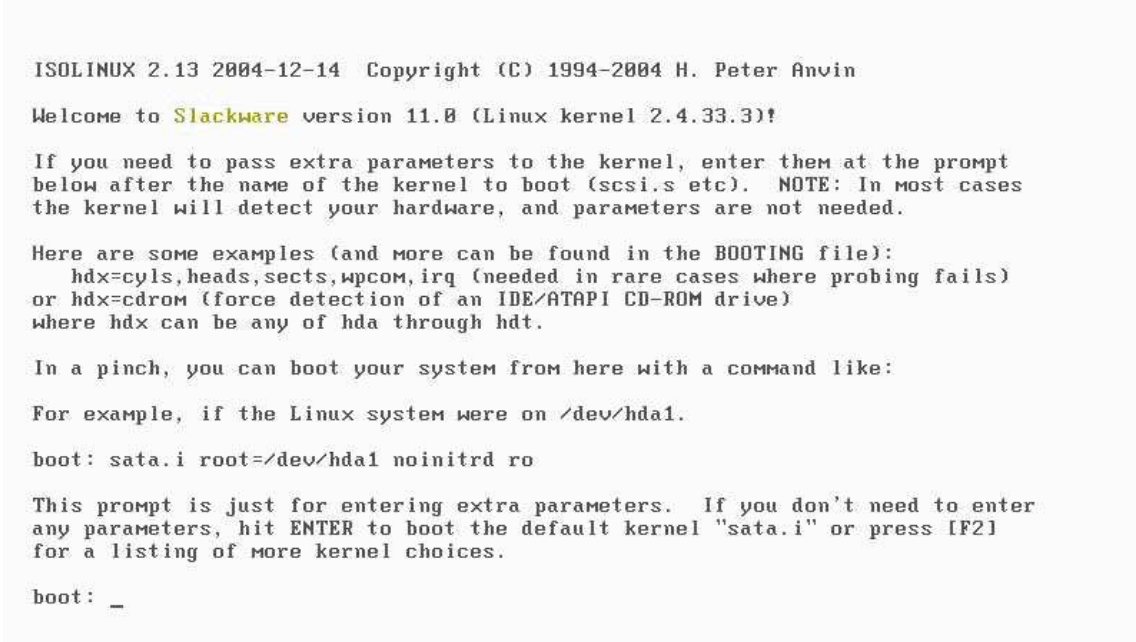
/Usr – nachází se zde mnoho programů, dokumentů, zdrojových kódů jádra. Do tohoto adresáře se nejčastěji ukládají programy

/Var – zde jsou systémové losovací soubory, nakrčovaná data a programy uzamčená soubry.

4.5 Instalace Slackware Linux [2]

Nejdříve je potřeba si rozvrhnout, z jakého média budeme chtít systém nainstalovat. Zvolil jsem instalaci z *iso* obrazů disků. Které jsou dostupné na internetové adrese http://slackware.com/. Máme na výběr několik verzí. V našem případě budu instalovat verzi 11. Verzi 11 jsem zvolil z důvodů aktuálních modulů (ovladačů), které budu potřebovat na použitý hardware. Iso obraz zapíši na CD popřípadě na DVD disk. Nyní můžeme pokračovat k samotné instalaci Slackware Linuxu.

Po vložení CD/DVD do mechaniky počítače musíme nastavit v BIOSu položku pro bootování[8] z CD. Pokud jsme udělali tento krok správně, ukáže se nám obrazovka Slackware Linux.



[8] Bootovaní znamená zavádění systému

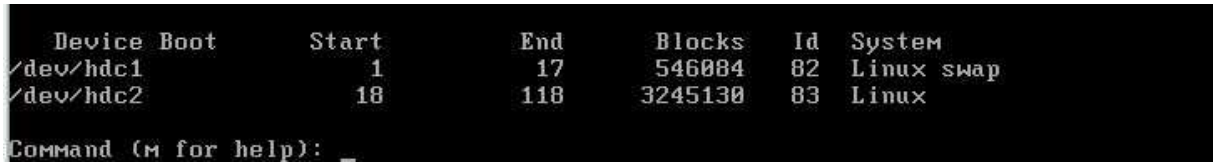
Obrázek č.10 Úvodní obrazovka Slackware Linux

Standardní instalace Slackware Linuxu probíhá s jádrem 2.4.33.3, Vzhledem k podpoře více SATA a SCSI řadičů je nutné nainstalovat jádro 2.6.X. V nabídce si můžeme vybrat, které jádro má být zavedeno. Použil jsem příkaz *#test26.s*[9], který znamená že, bude zavedeno jádro 2.6.18. Po zavedení jádra se systém zeptá jaké chceme zvolit rozložení klávesnice. Zvolíme *default* pro *US keybord layout.* Systém nás požádá o zadaní přihlašovacích údajů *(root).*

Před samotnou instalací Slackware Linuxu je nutné rozdělit disk na několik oddílů. Toto provedeme příkazem *root# fdisk /dev/název_disku* (místo názvu disku vyplním disk, na který chceme instalovat např */hda*). Nyní jsme v nabídce programu *fdisk*, pro výpis příkazů nám poslouží písmeno *„m“*. Rozdělení disku proběhne následovně. První část bude oddíl *Swap* odkládací prostor (měl by se rovnat dvojnásobku operační paměti). Druhá část bude kořenový oddíl operačního systému Linux */root*. Můžeme vytvořit libovolný počet oddílů (partition), ale pro naše účely to není potřeba. Pro vytvoření Linux Swap tedy zadáme „n“ pro nový oddíl, „p“ pro primární, partition číslo (*partition number*) je „1“ (je to první partition), *fist cylinder* necháme *„1“*. Poslední (*last*) *cylinder* nastavíme přibližně na dvojnásobek operační paměti např. *„+1000M“* (oddíl o velikosti 1G).

Příkazem *„p“* si prohlídneme tabulku, kde by měl být už zapsán oddíl */dev/hda1*. Pokud tomu tak není, zkusíme vytvořit oddíl znovu. Oddíl máme vytvořený, ale ještě musíme označit tento oddíl jako Swap. To provedeme příkazem *„t“*, kde zadáme číslo *„82“* pro Linux Swap, kompletní seznam možných označení se vypíše příkazem *„l“*. Po opětovném zadání příkazu *„p“* uvidíme, že oddíl je již označený Linux swap. Přejdeme na vytvoření kořenového oddílu. Postup je obdobný, nejdříve *„n“* pro novou partition, *„p“* pro primární, číslo bude *„2“* pro druhý oddíl. Pokud nebudeme vytvářet více oddílů, necháme *First cylinder* a *Last cylinder* beze změn. Po zadání příkazu *„p“* uvidíme přibližně toto:

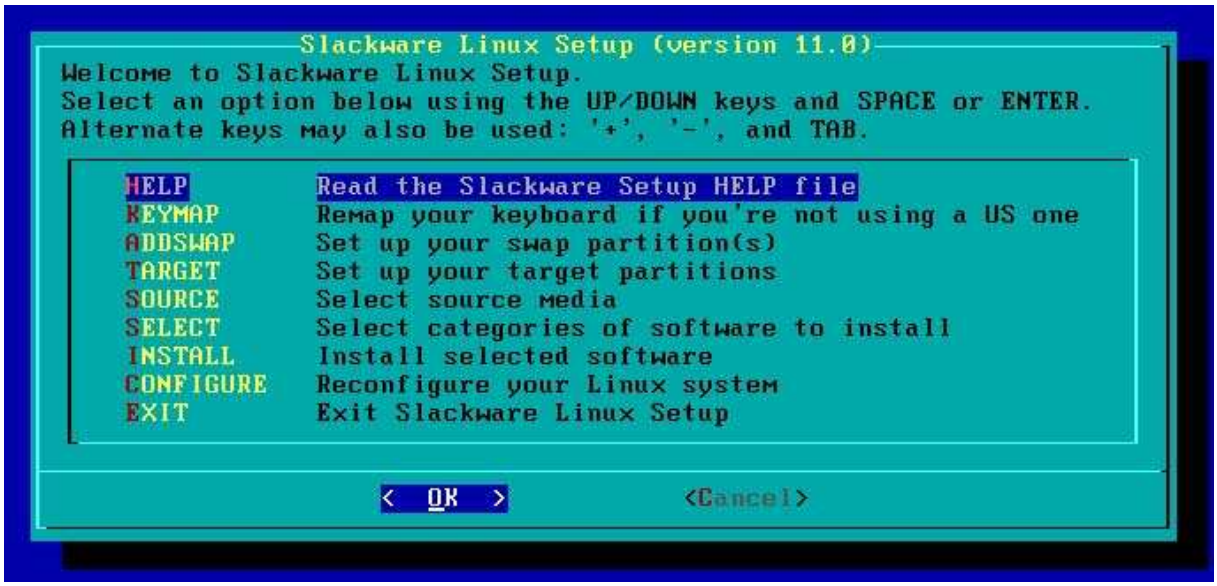
[9] # označuje příkazovou řádku



Obrázek č. 11 Seznam vytvořených oddílů

Rozdělený disk už máme. Teď už stačí jen všechno na disk zapsat, provedeme to příkazem „*w*“.

Po zapsání informací o rozdělení disku můžeme začít s instalací operačního systému. Napíšeme *root# setup*[10].



Obrázek č. 11 základní menu instalace Slackware Linux

V nabídce postupujeme od první položky. Položka *KEYMAP* určuje zvolené rozložení klávesnice. Setup nám nabízí další činnost, tou je připojení Swap oddílu. Můžeme zde překontrolovat oddíl jestli neobsahuje vadné bloky. Další krok je připojení kořenového oddílu. Dále máme na výběr, ze kterého média chceme systém nainstalovat. Například DVD médium, u kterého zvolíme v další volbě automatický režim vyhledávání zdroje.

V následující nabídce zaškrtáváme balíčky, které chceme nainstalovat (A, AP, D, E, F, K, KDE, KDEI, L, N, T, TCL, X, XAP, Y). Pro minimální instalaci stačí

[10] Příkazovou řádka je dále zkrácena na „root#“, kde *root* je označení pro superuživatele.

balíčky z A, AP. Nejjednodušší je nastavení všech balíčků, ale zabírá to i nejvíce místa na disku. Po vybrání sad balíčků nám setup nabídne varianty instalace. První volba je instalace s názvem *full*, je to instalace všech balíčků ze sad bez potvrzení. *Expert* je plná kontrola nad balíčky, kde si vybereme balíčky, které chceme nainstalovat. *Menu* nám poskytne přehled, ve kterém můžeme instalovat různé balíčky ze sad, nejsou zde uvedeny povinné balíčky, ty se instalují automaticky. *Newbie* instalace se ptá na každý nepovinný balíček otázkou „Yes“ , „No“, „Skip“. To znamená jestli chceme nainstalovat balíček, nebo nenainstalovat, nebo přeskočit s tím, že se k němu instalace později vrátí. Instalace typu *Custom, Tagpath*, jsou určeny pro použití vlastního stromu instalace, který si můžeme vytvořit před instalací.

Po nainstalování balíčků nám systém nabídne výběr jádra, které chceme nainstalovat. Vyberu jádro z CDROM test26.s. Krok vytvoření bootovací diskety přeskočíme. Ve výběru modemu zvolíme *„no modem“*. Na obrazovce „Enable Hotplug“ zvolíme *Yes*, jádro bude mít možnost nacházet nový i stávající hardware automaticky. Dalším krokem je zavedení boot manageru, v tomto případě se jedná o LILO. Zvolíme nabídku *Simple*. Ve výběru rozlišení zvolím optimální pro monitor (1024x768x256). Speciální parametry pro LILO nechceme ponecháme prázdné. LILO zavedeme do MBR (master boot record). V nabídce typu připojení myší vyberu *PS2*. V dalším okně GPM dám *yes*. Nyní máme možnost nastavit síť, můžeme později vyvolat příkazem *root# netconfig* V našem případě nastavíme vše ručně. *Hostname: „Master“* (jméno počítače), *domain*: „ . “ (nemáme vytvořenou doménu), dále vybereme *static IP: „192.168.10.1“* (Master bude sloužit jako DHCP server proto je potřeba nastavit adresu napevno). *Mask: „255.255.255.0“*, Gateway „x.x.x.x“ (brána, nemusíme zadávat, master nebude mít přístup do jiné sítě). *Nameserver* nemusíme vyplňovat. V nabídce servisů spustíme *rc.inetd* (daemon pro síť), *rc.syslog (*pro zápis logů*), rc.sshd (*pro vzdálený přístup SSH.. Přidat fonty nechceme. Nastavíme aktuální čas (*Europe.Prague*). Pokud jsme si nainstalovaly Windows managery (KDE, GNOME)[11], vybereme, který se má spouštět jako výchozí. Nastavíme heslo pro

[11] KDE, GNOME – desktopová prostředí pro Linux a další operační Uniové systémy.

uživatele root, které potvrdíme opětovným zapsáním. Instalaci máme hotovou, můžeme opustit setup a restartovat příkazem *root# reboot.*

4.6 Konfigurace Slackware

Při bootování se nám zobrazí manažer (LILO) s výběrem naší instalace. V zavedení jádra se nám vygenerují klíče pro budoucí spojení SSH.

Přihlásíme se pod uživatelem root. První krok je překontrolování fungující sítě, provedeme příkaz *root# ifconfig.* Měla by se zobrazit rozhraní *eth0 a lo,* v případě, že síťové rozhraní *eth0* není k dispozici pokusíme se zavést modul naší síťové karty (např *root# modprobe e100*). Vzhledem k tomu, že jsme zaváděli jádro 2.6.x nemáme moduly k dispozici. Musíme je tedy zkompilovat spolu s jádrem. Čerstvý zdrojový kód jádra si stáhneme na stránce www.kernel.org. Rozbalíme do složky /usr/src (*root# utar -xfz*) a v složce jádra provedeme konfiguraci *root# make menu config*, Zde si zvolíme, které ovladače zkompilujeme do jádra, nejvíce je potřeba síťová karta a podpora file systému. Po konfiguraci provedeme příkaz *root@# make bzImage*, kterým se spustí samotná kompilace. Po vytvoření bzImage, vytvoříme moduly *#make modules* (v případě, že jsme nějaké nastavili v konfiguraci jádra). Po vytvoření modulů je třeba je nainstalovat *root# modules_install*.

Důležitým prvkem pro zprovoznění clusteru je konfigurace DHCP serveru. Systém bude přidělovat klientů (uzlům) IP adresy, podle MAC[12] adresy síťové karty. Nastavení DHCP server v Slackware Linuxu se provádí v souboru */etc/dhcpd.conf.* V tomto konfiguračním souboru nastavím potřebně parametry [2]:

[12] MAC adresa je hardwerová adresa karty, které je uložena na síťové kartě a je určena výrobcem

```
default-lease-time 86400;
max-lease-time 172800;
allow booting;
allow bootp;
ddns-update-style ad-hoc;
next-server 192.168.10.1;
filename „/pxelinux.0“;

subnet 192.168.10.0 netmas 255.255.255.0
{
range 192.168.10.2 192.168.10.255;

        # definice uzlu
        host nod1 { hardware ethernet 00:00:44:00:45:FF;
         fixed-address 192.168.30.3;}

}
```

Dále můžeme pokračovat v libovolném přidávání MAC adres uzlů.

Dalším krokem je aktivace TFTP[13]. To je potřeba k nahrání jádra ze sítě. To provedeme v souboru */etc/inetd.conf* zrušením komentáře (#) u *tfpt*. Nastavíme složku odkud se bude nahrávat jádro */tftpboot.*

Jádro je příliš velké a nevejde se do paměti síťové karty, proto musíme použít zavaděč pro nahrání jádra (PXELINUX). Pxelinux je zavaděč, který se nahraje do paměti síťové karty a podle konfigurace najde, začne nahrávat jádro. Pxelinux je volně ke stažení na stránkách http://syslinux.zytor.com/pxe.php. Kde je i dokumentace k tomuto projektu. Po rozbalení Pxelinuxu na disk, najdeme soubor pxelinux.0, který uložíme do složky /tftpboot. Vytvoříme zde adresář */tftpboot/pxelinux.cfg*, do kterého vytvoříme konfigurační soubor s názvem MAC adresy našeho uzlu (nod1 “00:00:44:00:45:FF“) nebo soubor *default*. Pokud zvolíme první variantu, můžeme každému uzlu na základě jeho MAC adresy přiřadit jiné jádro, hodí se především v případě, že nemám jednotnou platformu (např. AMD a INTEL).

Do konfiguračního souboru zadáme řádky [3]:

```
LABEL linux
        KERNEL /bzImage
        APPEND  ip=dhcpd root=/dev/nfs nfsroot=192.168.20.1:/FTP/boot rw
```

Jádro bzImage bude uloženo ve složce */tftpboot*. Toto jádro vytvoříme kompilací. Nejlepší je zkompilovat potřebné věci do jádra a nevytvářet moduly, moduly se uzlům hůře zavádějí. Příkazem *APPEND* předáváme parametry jádra, *ip* znamená jak získá nebo jakou bude mít uzel IP adresu. V našem případě přidělujeme IP adresu pomocí DHPC. Příkazem *nfsroot* určujeme odkud bude uzel nahrávat file system. Zadává se zde IP adresa NFS serveru, zde je to adresa hlavního počítače (Masteru). Na Masteru je třeba tento NFS server nastavit, to provedeme v */etc/exports* zadáním řádky

*/tftpboot/boot *(rw,no_root_squash,no_subtree_check,sync)*

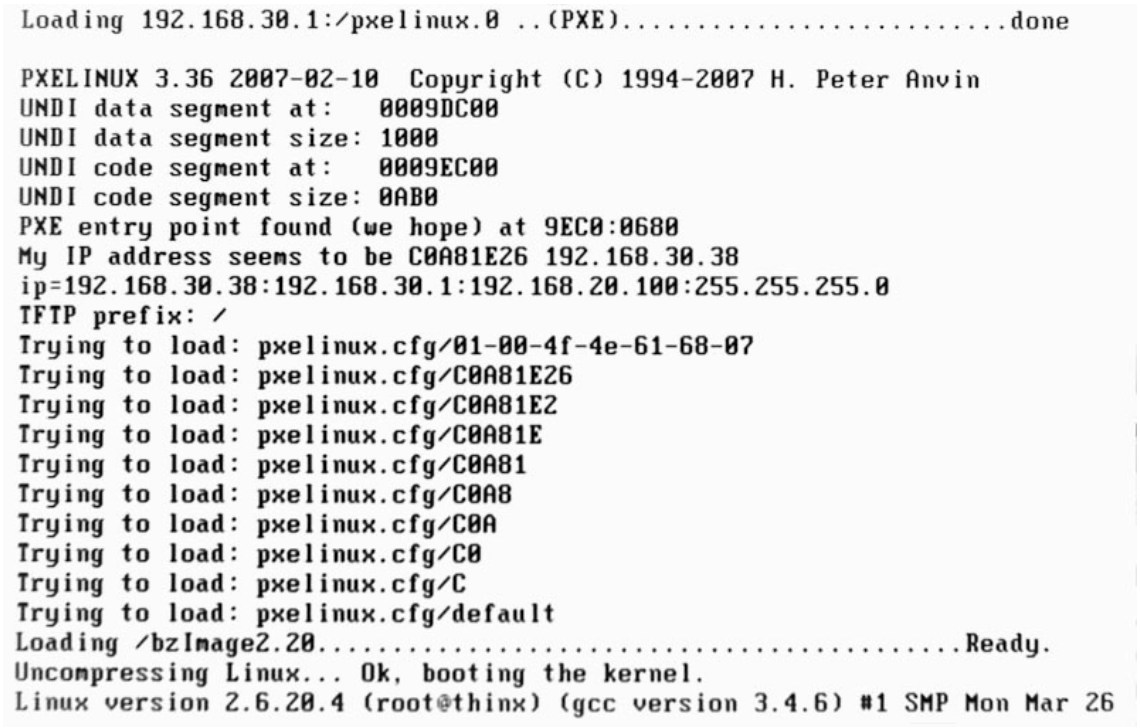
Toto nastavení nám umožní přístup všech počítačů k souborů uloženým v */ftpboot/boot*.

Do vytvořené složky */tftpboot/boot/* přeneseme potřebné adresáře se soubory (root,sbin,bin,var,proc atd.). V této složce je kořenový adresář uzlu, proto ještě musíme nastavit jiné jméno. V souboru /etc/hosts nastavíme jméno na NOD. V případě použití více uzlů je vhodné použít DNS server, který bude přidělovat jednotlivým uzlům jména.

Pro spuštění NFS serveru musíme provést příkaz *root#: /etc/rc.d/rc.nfsd start.* V případě korektního spuštění se objeví přibližně tento dialog :

```
Starting NFS server daemons –r
        /usr/sbin/exportsfs –r
        /usr/sbin/rpc.rquatad
        /usr/sbin/rpc.nfsd 8
        /usr/sbin/rpc,,mountd
```

Pokud jsme nastavili vše správně měla by se objevit přibližně tato obrazovka:



Obrázek č. 12 bootovaní jádra

Na obrazovce je vidět, že DHCP server správně přidělil IP adresu a jádro se přenáší TFTP . Pokud máme správně zkompilované jádro měl by se objevit přihlašovací dialog.

4.7 Instalace MPI

Pro distribuované výpočty jsem si vybral program openMPI. Program je volně ke stažení na www.open-mpi.org. Instalace je zabalena, proto je nutné ji nejdříve rozbalit. Po rozbalení do příslušné složky (např. /tmp). Můžeme provést kompilaci. Nejprve v adresáři openMPI provedeme příkaz *root@nod# ./configure --prefix=/home/openMPI/* (za prefix se nachází cesta kam chceme program nainstalovat). Po provedení konfigurace můžeme přejít k samotné kompilaci (instalaci). Příkazem *root# make all install* se provede kompilace s nainstalováním do cesty zadané konfigurací.

Stejný postup provedeme i na počítači NOD. Po nainstalování openMPI můžeme přejít k testovací fázi. Uděláme kontrolu zdali se nám do adresáře */lib* nahrály knihovny k openMPI, pokud se zde nenachází nahrajeme je z nainstalovaného adresáře.

V nainstalovaném adresáři se nachází složka */examples*. V této složce se nacházejí jednoduché příklady. Na NODu zkompilujeme příklad ring_c.c příkazem *root@nod# mpicc ring_c.c –o ring*. Vytvoří se nám soubor ring který můžeme vyzkoušet. Zpět na masteru provedeme tento příkaz *root@master# mpirun –np 1 ring*. Kde –np znamená kolik chceme spustit úkolů. Pokud nám výpočetní cluster funguje správně vypíší se sestupně hodnoty 9-0.

4.8 Problémy

Při instalaci výpočetního clusteru narazíme na mnoho problémů, většinou triviálních. Pro jejich řešení doporučuji pročítat manuály, používat Linuxový odchytávač síťových paketů *tcp dump* a číst systémové a programové logovací soubory. Dalším užitečným pomocníkem pro instalaci jsou MC (midnight commander – souborový manažer), který v některých krocích nahrazuje příkazovou řádku a editor VI[14].

Jedním z problémů je kompilace nového jádra. Může se stát, že se nám nepovede začlenit všechny ovladače. V případě, že nemáme možnost přidat ovladače přímo do jádra musíme použít *initrd*[15], do kterého potřebné ovladače přidáme. V této instalaci není *initrd* použit, vše potřebné bylo přidáno do jádra.

[14] VI je editor, kterým je možné editovat prostý text
[15] Initrd (initial RAM disk) – RAM disk který je načítán jádrem a obsahuje ovladače zařízení ve formě modulů. Umožňuje použít ovladače, které nejsou zkompilovaná v jádře.

4.9 Praktický příklad[16]

Praktický příkladem výpočetního clusteru jsem si vybral Super počítač Amálka, který se nachází v Ústavu fyziky a atmosféry akademie věd v Praze (ÚFA). Jedná se o nerychlejší výpočetní cluster v ČR. Realizací tohoto výpočetního clusteru je pověřena firma SPRINX Systéme.

Tento superpočítač se skládá z 84 dvou jádrových procesorů Intel® Xeon™ 5140 (Woodcrest). Které jsou po dvou umístěny v 1U počítačových bednách. A ze 192 jedno jádrových procesorů Intel Xeon™ s frekvencí 2,8 GHz. K tomu je potřeba 180GB RAM a 20Tbyte diskového prostoru, který je tvořen několika diskovými poli typu RAID 5. Celý systém tedy dává výkon změřený LINPACKem 1,13 Tflops. To znamená, že Amálka má teoretický výkon 2,5 bilionů operací v plovoucí čárce za sekundu.

Amálka běží na operačním systému linux Slackware s nejnovějším jádrem. Kdy je nainstalován sytém distribuce úkolů OPEN MPI.
Tento superpočítač počítá především numerické výpočty. Tým Dr. Pavla Trávníčka usilovně pracoval poslední dva roky na těchto projektech:

- vysvětlit některé procesy, které probíhají v magnetosféře Země a interpretovat pozorování družice Cluster II (ESA)
- vytvořit model magnetického pole Merkuru, který slouží pro plánování družicových misí BepiColombo a MESSENGER k této planetě

Dalšími plánovanými výpočty jsou

- studium Měsíčních magnetických anomálií, které mají podobně jako zemská magnetosféra schopnost odstínit sluneční vítr, který je životu nebezpečný. Měsíc by se tak mohl stát vhodným pro budování základen s lidskou posádkou.

[16] Tisková zpráva. *hpc.sprinx.cz* [online]. 2006, roč. 2006 [cit. 2007-06-05]. Dostupný z WWW: <www.hpc.sprinx.cz>.

- příprava misí k výzkumu Slunce jakými jsou Solar Orbiter a Solar Probe plánovaných na příští desetiletí.

5 Závěr

V teoretické části práce jsem se věnoval základním vlastnostem výpočetních clusterů. Zaměřil jsem se na jejich nejdůležitější části, které jsou důležité pro celkový výkon. Tyto části jsou nejenom hardwarové, ale i softwarové. V kapitole hardware jsem se věnoval procesorům, pamětem, typům sítí a diskovým polím. Na tyto části je nahlíženo z hlediska realizace výpočetních clusterů, a proto nezacházím příliš hluboko do teorie fungování hardwaru. V kapitole o softwaru jsem nejdříve uvedl operační systémy, které lze použít pro stavbu výpočetních clusterů. Nejsou zde uvedeny všechny existující, ale jen výběr nejpoužívanějších a nejznámějších.

Z praktické části vyplývá, že instalace výpočetního clusteru se zdařila a obě testovací konfigurace fungují podle zadaného cíle. Na druhé testovací konfiguraci se mi podařilo ověřit plnou funkčnost jádra 2.6.X.

Nejprve jsem na hlavní počítač (Master) nainstaloval operační systém Slackware Linux, který byl následně nastaven pro jako DHCP server, NFS server. Dále probíhalo nastavení PXELINUX, který umožňuje nahrávání jádra, file systému po síti. Instalace na jiných operačních systémech typu Linux by probíhala velmi podobně.

Dalším krokem ve vývoji tohoto výpočetního clusteru bude instalace a konfigurace job scheduleru. Pokud bude vše fungovat, bude celá instalace implementována na superpočítač Amálka.

6 Zdroje:

[1] OpenMPI [online]. Bloomington USA : The Open MPI Project , 2004-2007 , 20-června-2007 [cit. 2007-06-05]. Infromace o openMPI projektu. Dostupný z: < *Http://www.open-mpi.org*>

[2] Slackbook [online]. The Revised Slackware Book Project, 2004-2007 , 29. prosince 2006 [cit. 2007-06-05]. Dokumentační projekt k linux slackware. Dostupný z < *http://www.slackbook.org/html/*>

[3] PXELINUX [online].: PXELINUX Documentation, 2004-2007 , 9.června 2007, [cit. 2007-06-05]. Dokumentace k PXE Linux. Dostupný z: < *http://syslinux.zytor.com/pxe.php* >

[4] VALOUŠEK, Ondřej . *Jak nabootovat linux po siti* [online]. 1999-2007, 24. 10. 2005 [cit. 2007-06-05]. Jak nabootovat linux po síti. Dostupný z: <http://www.abclinuxu.cz/clanky/navody/jak-nabootovat-linux-po-siti>.

[5] Storage area network [online]. Wikipedia, 4. června 2007 [cit. 2007-06-05]. Základní charakteristika SAN. Dostupný z: <*http://en.wikipedia.org/wiki/Storage_area_network*>

[6] Zdeněk Mařík. Konfigurace rozsáhlých datových systémů v prostředí OS Unix. 1. vydání. Praha: BEN, 2001. 156s. ISBN 80-7300-012-1.

[7] Linux high performance computing and linux clusters [online]. LinuxHPC.org, 2001-2007 [cit. 2007-06-05]. Dostupný z : <*http://www.linuxhpc.org/*>.

[8] Parallel_computer [online]. Wikipedia, 1. června 2007 [cit. 2007-06-05]. Paralelní počítání. Dostupný z: <*http://en.wikipedia.org/wiki/Parallel_computer*>

[9] Joseph D. Sloan. High Performance Linux Clusters. O`Reilly, 2004. 360s. ISBN: 0-596-00570-9.

[10] Statistics on high-performance computers : TOP 500 supercomputers [online]. Top500.org, 2000-2006 , 11.2006 [cit. 2007-06-05]. Statistiky 500 největších superpočítačů na světě. Dostupný z : <http://top500.org/stats>.

[11] LAM/MPI [online]. Trustees of Indiana University, 14. února 2007 [cit. 2007-06-05]. Informace o LAM/MPI. Dostupný z <http://www.lam-mpi.org/>

[12] Pořiďte si RAID 1 (Zrcadlení) [online], Svět hardware. 12.1.2005 [cit. 2007-06-05]. Informace o polích typu RAID. Dostupný z: <*http://www.svethardware.cz/art_doc-F06BA8749FE1FD0AC1256F610053B1D5.html*>

7 Přílohy

Příloha 1 – fotografie superpočítače Amálka