

開
放
的
열린
مفتوح
libre
मुक्त
ಮುಕ್ತ
livre
libero
ముక్త
开放的
açık
open
nyílt
:::
πικρ
オープン
livre
ανοικτό
offen
otevřený
öppen
открытый
வெளிப்படை

open

 USE  MELHORE  ESPALHE

Dia da Liberdade do Software 2008 - Fortaleza

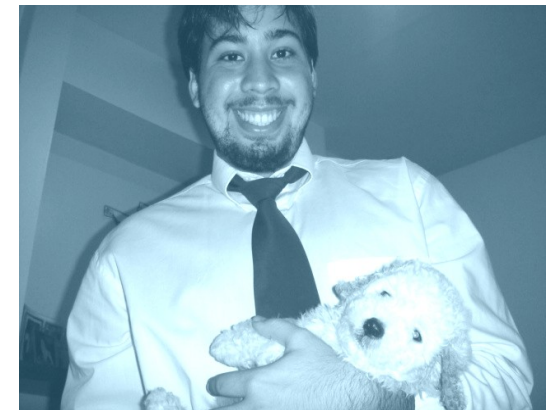
OpenSolaris a céu aberto

Silveira Neto
Campus Ambassador, Sun Microsystems



Quem eu sou?

- Estudante de Computação na UFC
- Membro
 - CEJUG (Ceará Java User Group)
 - ParGO (Paralelismo, Grafos e Otimização)
- Software Livre
 - Palestras
 - Código
- Blogueiro
 - silveiraneto.net
 - eupodiatamatando.com
- Embaixador de Campus da Sun Microsystems
 - NetBeans, OpenSolaris, JavaFX, etc.
- Músico amador.





Agenda

- OpenSolaris
 - ZFS
 - Dtrace
 - Virtualização



OpenSolaris

- Solaris
- Sistema operacional livre
 - <http://opensolaris.org>
 - <http://opensolaris.com>
- Funcionalidades de ponta
 - ZFS, Dtrace, Containers, etc.
- Userland GNU



Distribuições

- OpenSolaris
 - <http://www.opensolaris.com>
- Belenix
 - <http://belenix.sarovar.org/>
- Martux
 - <http://www.martux.org/>
- Nexenta
 - <http://www.nexenta.org/os>
- Solaris
 - <http://www.sun.com/software/solaris/>
- Schillix
 - <http://schillix.berlios.de/>
- Milax
 - <http://milax.org>



ZFS, pra que?

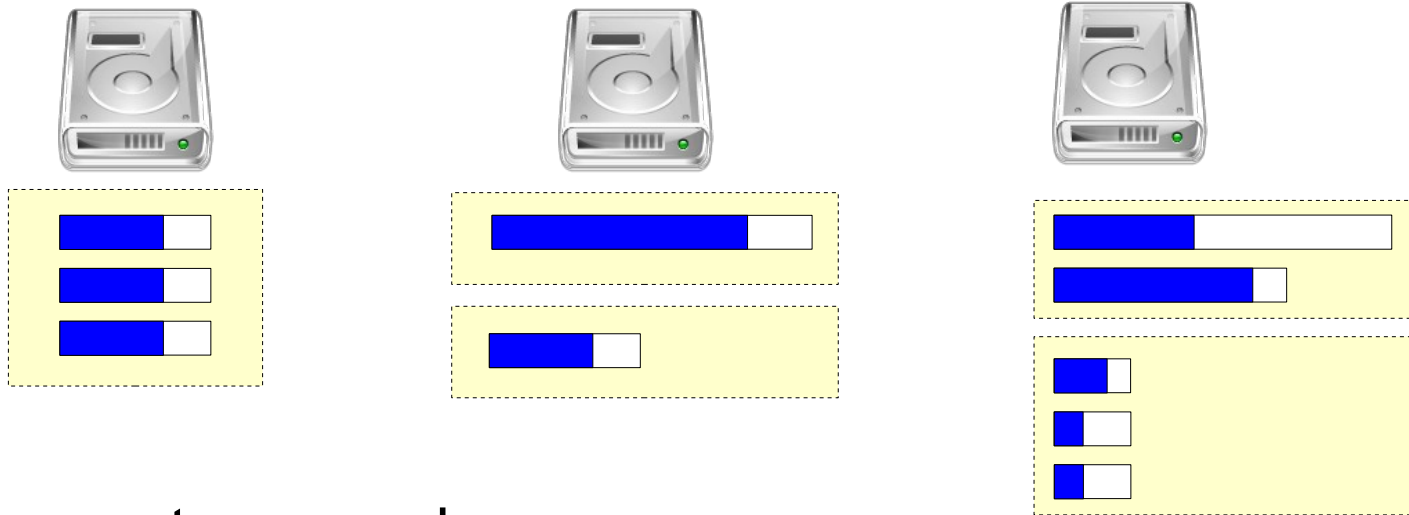
- Capacidade
 - 128-bit → 2^{128} bytes
 - “Para lotar um ZFS seria necessário mais energia do que o suficiente para ferver os oceanos” [1]
 - Lei de Moore
 - ZFS pode guardar 16 exabytes [2]
- Falhas
 - “há apenas dois tipos de unidades de disco: aquelas que já falharam e as que estão prestes a falhar”, Jonathan Schwartz [3]

ZFS, algumas características

- Administração fácil
 - pools
- Integridade dos dados
 - Detecção e correção de falhas silenciosas
- Operação transaccional
 - tudo ou nada
- Backup
 - snapshots
 - clones

Gerenciamento de Volumes Tradicional

Dispositivos e partições:



Armazenamento ocupado:



Armazenamento disponível:



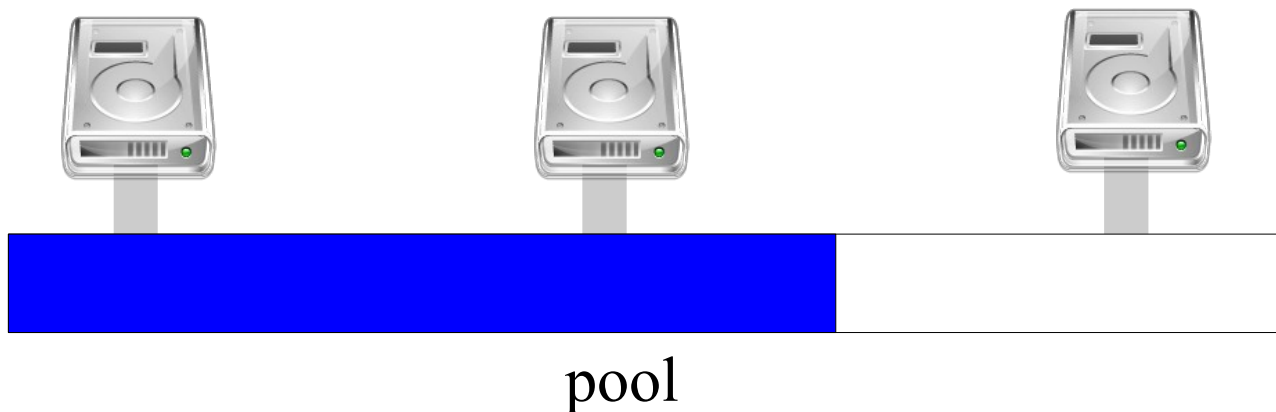
| | |
|--|---------------------|
| | utilizado |
| | livre |
| | sistema de arquivos |



Volumes Tradicionais

- volumes ou partições para cada sistema de arquivos
- aumenta ou diminui na mão
- armazenamento fragmentado
- cada sistema de arquivos tem uma largura limitada
- administração trabalhosa

ZFS pools



- “poças” de armazenamento
- compartilhamento de armazenamento
- esqueça as partições
- cresce e diminui automaticamente
- administração fácil

ZFS em exemplos

- Criando um mirror chamado “tank”

```
# zpool create tank mirror c0t0d0 c1t0d0
```

- Criando um sistema de arquivos para home, montado em /export/home

```
# zfs create tank/home  
# zfs set mountpoint=/export/home tank/home
```

- Criando sistema de arquivos para cada usuário
 - Montados em /export/home graças a herança

```
# zfs create tank/home/felipe  
# zfs create tank/home/davi  
# zfs create tank/home/ramiro
```



ZFS em exemplos (2)

- Comprei dois discos novos, adiciono-os no pool tank como um novo mirror

```
# zpool add tank mirror c2t0d0 c3t0d0
```

- Compartilhar dados dos usuários na rede

```
# zfs set sharenfs=rw tank/home
```

ZFS em exemplos (3)

- Compressão de dados no pool

```
# zfs set compression=on tank
```

- Felipe só poderá usar 10Gb de espaço

```
# zfs set quota=10g tank/home/felipe
```

- Davi precisa ter certeza que vai ter 20Gb

```
# zfs set reservation=20g tank/home/davi
```



ZFS Snapshots

- Snapshots
 - “máquina do tempo”
 - criação instantânea
 - sem espaço adicional
 - blocos copiados só quando mudam
 - usuário pode fazer sozinho
- ♦ .zfs/snapshot de cada sistema raiz
- ♦ não precisa chamar o administrador do sistema

ZFS em exemplos (4)

- Snapshot dos dados do Ramiro

```
# zfs snapshot tank/home/ramiro@quintafeira
```

- Ramiro precisa do backup de terça?

```
# zfs rollback tank/home/ramiro@tercafeira
```

- Como era mesmo aquele arquivo que eu apaguei sexta?

```
# firefox ~ramiro/.zfs/snapshot/quintafeira/foto.jpg
```



DTrace

- framework para sondagem
- tempo real
- para o kernel
- para aplicações
- linguagem de script D

Dtrace em exemplos

- Contagem de syscalls por programa

```
dtrace -n 'syscall:::entry { @num[execname] = count(); }'
```

- Contagem de syscalls por processo

```
dtrace -n 'syscall:::entry { @num[pid,execname] = count(); }'
```

- Bytes escritos por processo

```
dtrace -n 'sysinfo:::writech { @bytes[execname] = sum(arg0); }'
```

- Bytes lidos por processo

```
dtrace -n 'sysinfo:::readch { @bytes[execname] = sum(arg0); }'
```



Virtualização

- Containers
 - Zones
 - Consolidação
- Branded Zones
 - Personalização do pacotes
 - Outros sistemas (ex: Linux)



Uma comunidade, mas que nome?

- OSUG
 - OpenSolaris User Group
- FOSUG?
- CEOSUG?
- ~~OSUG~~
- Solar
- Sol
- Livre
- Aberto
- Sol Livre
- Sol Aberto



Referências

1. Post no blog do Jeff Bonwick, “128-bit storage: are you high?”
http://blogs.sun.com/bonwick/entry/128_bit_storage_are_you
2. ZFS Frequently Asked Questions, <http://opensolaris.org/os/community/zfs/faq/>
3. Post no blog do Jonathan Schwartz,
http://blogs.sun.com/jonathan_pt/entry/o_glamour_do_transporte_coletivo
4. Apresentação “ZFS, The Last Word in File Systems”, Jeff Bonwick, Bill Moore.
5. Livro “Introdução a Sistemas Operacionais: Uma abordagem prática usando o projeto OpenSolaris”, Sun Microsystems, Inc., 2007.
6. Apresentação “An Introduction to OpenSolaris”, Peter Karlsson.
7. DTrace Topics Intro, http://www.solarisinternals.com/wiki/index.php/DTrace_Topics_Intro
8. Artigo ZFS, Wikipédia, <http://en.wikipedia.org/wiki/ZFS>
9. Tema da apresentação, ícones por Chandan <http://blogs.sun.com/chandan>
10. Alguns outros ícones por Everaldo Coelho <http://www.everaldo.com>

開
放
的
열린
مفتوح
libre
मुक्त
ಮುಕ್ತ
livre
libero
ముక్త
开放的
açık
open
nyílt
:::
πικρ
オープン
livre
ανοικτό
offen
otevřený
öppen
открытый
வெளிப்படை

open

 USE  MELHORE  ESPALHE

Obrigado!

Silveira Neto

Campus Ambassador, Sun Microsystems

silveiraneto@gmail.com / silveira@sun.com

<http://silveiraneto.net>

