

Ressourcenmanagement im professionellen Einsatz

Rolf M Dietze
rolf.dietze@dietze-consulting.de

May-2009

Ressourcenmanagement

Konsolidierung und Ressourcenmanagement

Klassische Verfahren: i5

Zone und Multidomain

Prozessor und Ressourcen

CPU, Processorbinding

Prozessorsets

Memory

I/O, Net, IPQoS

I/O, Net crossbow

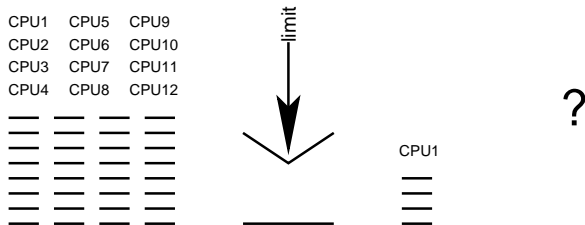
I/O, Disk

Project

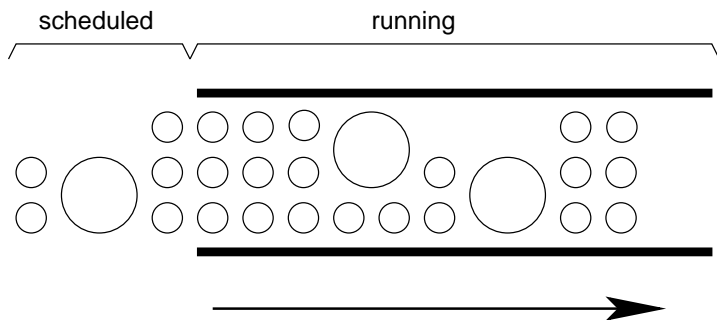
Pool

Conclusion

Ressourcenmanagement

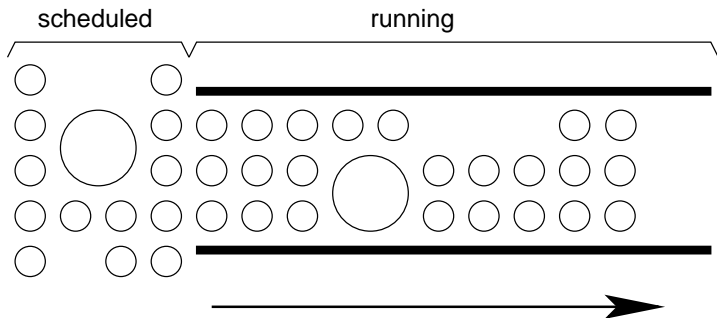


Ressourcenmanagement



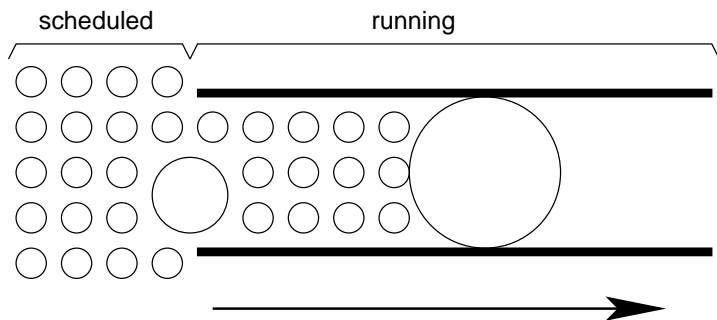
jobs/time light load

Ressourcenmanagement



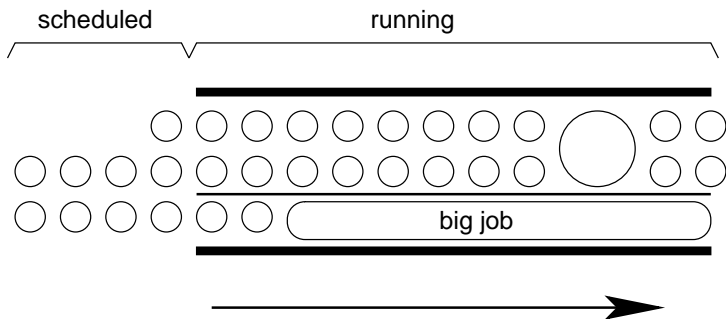
jobs/time moderate load

Ressourcenmanagement



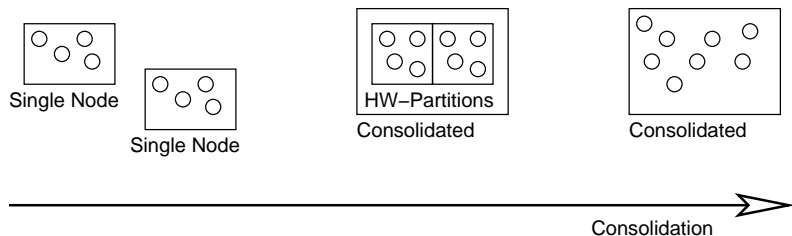
jobs/time overload, big jobs

Ressourcenmanagement

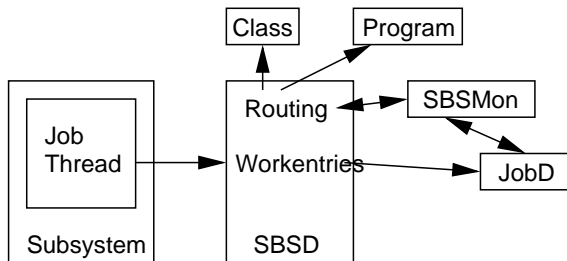


jobs/time overload, big jobs, limited

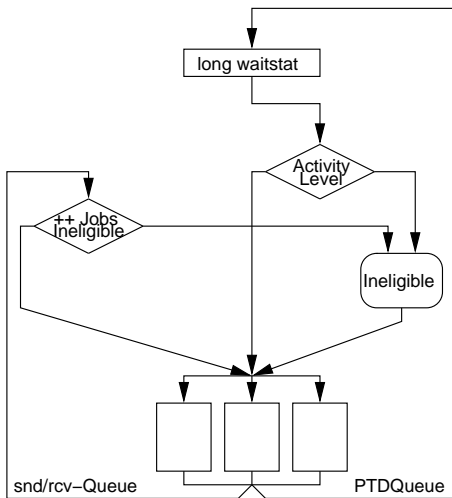
Konsolidierung und Ressourcenmanagement



Klassische Verfahren: i5

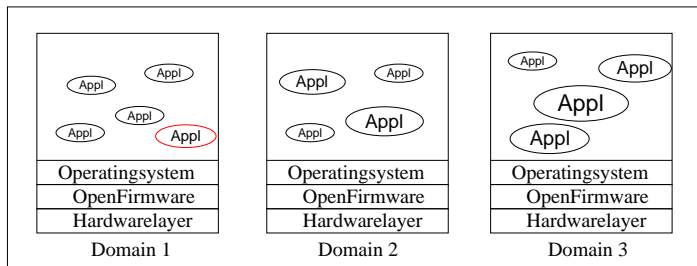


Klassische Verfahren: i5

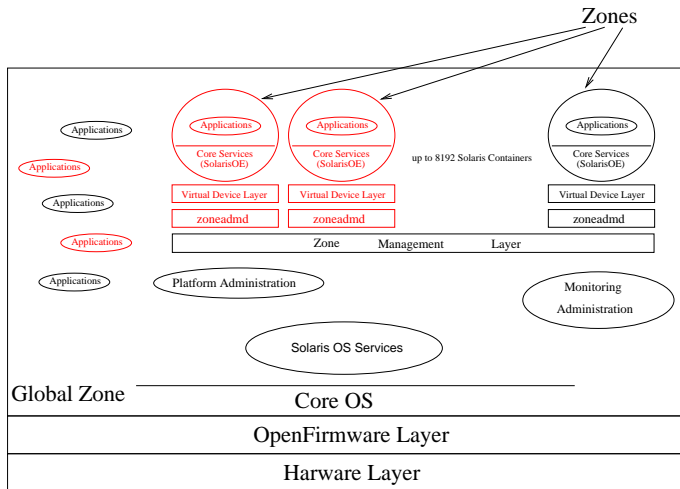


Multidomain

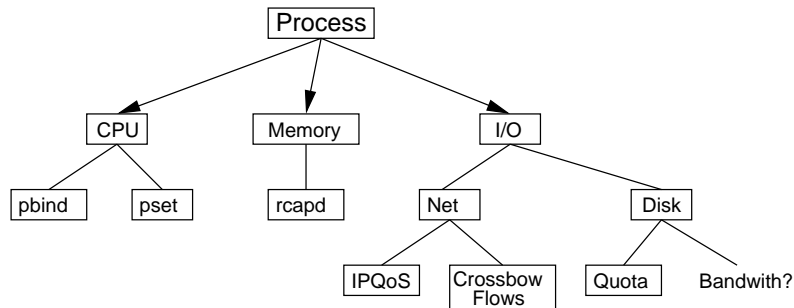
Machine Platform:



Zone



Prozessor und Ressourcen



CPU, Processorbinding

```
root@denebola pts/47 / 91# psrinfo
0      on-line   since 03/06/2009 13:25:59
1      on-line   since 03/06/2009 13:25:57
```

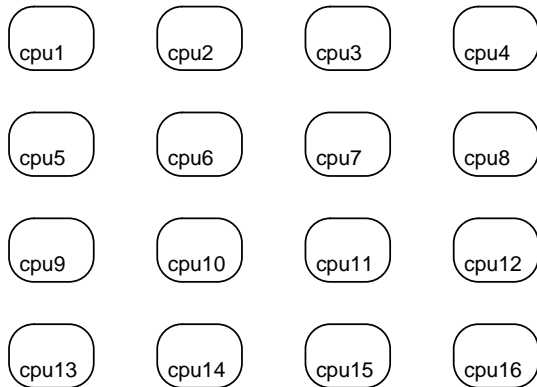
```
root@denebola pts/47 / 92# pbind -b 1 29824
process id 29824: was not bound, now 1
```

```
root@denebola pts/47 / 93# pbind -Q
process id 29824: 1
```

```
root@denebola pts/47 / 94# pbind -U 1
```

```
root@denebola pts/47 / 95# pbind -Q
```

Prozessorsets



Prozessorsets

cpu1

cpu2

5
cpu3

5
cpu4

set 1 → PID 201

1
cpu5

1
cpu6

5
cpu7

5
cpu8

set 2 → PID 305

set 3 → PID 1024

1
cpu9

3
cpu10

2
cpu11

2
cpu12

set 4 → no PID

set 5 → PID 995

3
cpu13

3
cpu14

4
cpu15

6
cpu16

set 6 → no PID

Processorset walkthrough

```
root@menkar pts/17 / 3# psrinfo
0      on-line   since 01/03/2007 11:39:42
1      on-line   since 01/03/2007 11:39:42
2      on-line   since 01/04/2007 18:25:59
3      on-line   since 01/03/2007 11:39:42
root@menkar pts/17 / 4# psrset
root@menkar pts/17 / 5# psrset -c 1 2
user processor set 1: processor 1 2
root@menkar pts/17 / 6# psrset -b 1 29054
process id 29054: was not bound, now 1
root@menkar pts/17 / 7# ps -e -o pset,pid |grep 29054
PSET  PID
    1 29054
root@menkar pts/17 / 8# psrset -U 1
root@menkar pts/17 / 9# ps -e -o pset,pid |grep 29054
PSET  PID
    - 29054
```

Memory

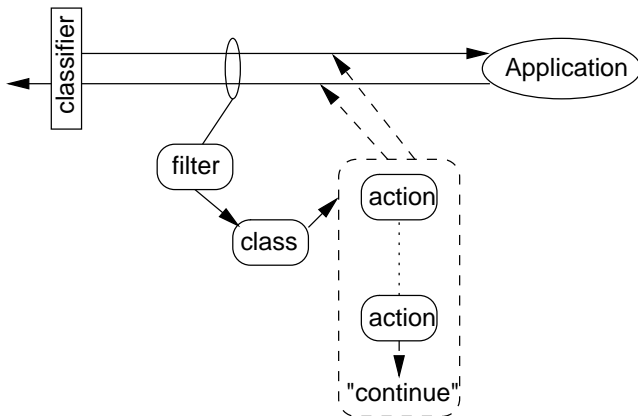
```
root@denebola pts/49 / 1# rcapadm -c 40 -E -i scan=20,sample=10,report=
                                state: enabled
    memory cap enforcement threshold: 40%
        process scan rate (sec): 20
    reconfiguration rate (sec): 60
        report rate (sec): 10
        RSS sampling rate (sec): 10

root@denebola pts/49 / 2# rcapadm -m 900M -z global

root@denebola pts/49 / 3# swap -l
swapfile          dev      swaplo  blocks    free
/dev/dsk/c1t0d0s1 32,25      16 16777456 14637424

root@denebola pts/49 / 4# rcapadm -D
```

I/O, Net, IPQoS



I/O, Net, crossbow

I/O, Net, crossbow

- L3/L4 Flow (Flowcontrol)
- Bandbreitenlimitierung
- Durchsatz, Prioritaeten
- CPU Zuordnung zu Flows und vNics

Flows basieren aus:

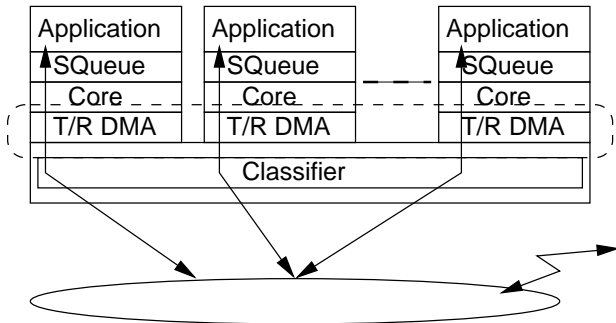
- Protocol/Port
- TCP/UDP,iSCSI,....
- IP-Adressen

Einstellungen:

- Prioritaet/Bandbreitengrenzen/CPU

```
root@mir-p2 3# flowadm create-flow -l eth2 protocol=tcp,local_port=2049
-p maxbw=40M nfs-l
root@mir-p2 4# flowadm set-flowprop -l eth2 -p maxbw=100M nfs-l
root@mir-p2 9# dladm create-vnic -l eth3 -m random maxbw=100M -p cpus 2
```

betrachtete Features



I/O, Disk

UFS \mapsto Quota

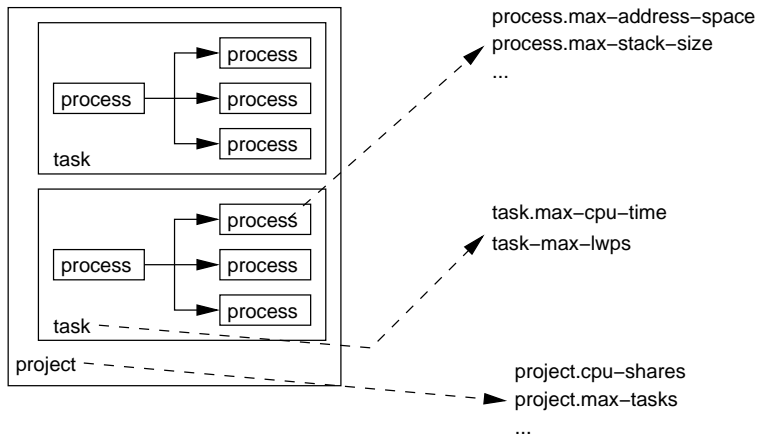
ZFS \mapsto Properties: quota, refquota reservation

Bandbreite/Durchsatz?

Beeinflussbar durch:

- Stripes, Concats, Raid5
- Keine solarisseitige Resourcelimitierung bzgl. HDStorage
- Bandbreitenbeeinflussung durch HDStorage etc.

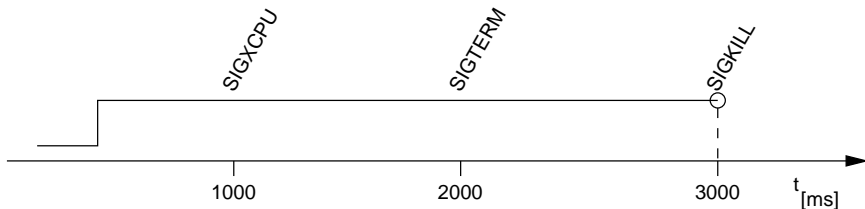
Project



Project

/etc/project:

```
prg:103:example program:root::process.max-cpu-time=(privileged,1000,signal=SIGXCPU),\  
      (privileged,2000,signal=SIGTERM), (privileged,3000,signal=SIGKILL)
```

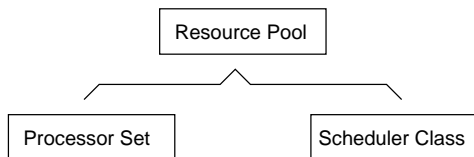


Einstellung von Privilegelevels, nach SunDoc:ID: 817-1975

project.user

PID	USERNAME	SIZE	RSS	STATE	PRI	NICE	TIME	CPU	PROCESS/NLWP
26461	lisa	522M	498M	run	1	0	2:10:36	29%	opera/58
29013	mjk	14M	4944K	sleep	1	15	0:08:29	7.9%	xdvi-motif.b
25739	mjk	509M	419M	sleep	55	0	6:28:15	6.3%	firefox-bin/
27818	mjk	237M	121M	sleep	59	0	1:17:16	1.4%	acroread/1
1192	mjk	4216K	3520K	cpu0	15	5	0:00:03	1.3%	prstat/1
12497	lisa	198M	191M	sleep	57	0	5:57:40	0.8%	Xsun/1
17944	mjk	408M	254M	sleep	49	0	2:37:19	0.8%	Xsun/1
7377	lisa	204M	58M	sleep	59	0	2:22:04	0.6%	acroread/1
2787	root	2776K	976K	sleep	59	0	6:17:27	0.1%	rpc.rstatd/1
PROJID	NPROC	SWAP	RSS	MEMORY		TIME	CPU	PROJECT	
3	184	2079M	1861M	23%		22:07:22	21%	default	
0	67	292M	300M	3.7%		10:35:25	0.2%	system	
1	1	1736K	5640K	0.1%		0:00:02	0.0%	user.root	
9	73	830M	811M	97.2%		8:00:02	32%	user.lisa	
Total: 252 processes, 607 lwps, load averages: 2.07, 1.84, 1.79									

Pool

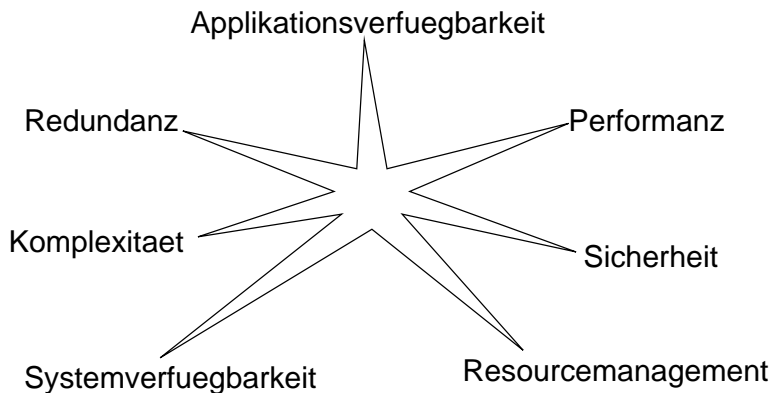


pool: pset_default

pool1: pset3

pool2: pset2, pool.scheduler"FSS"

Conclusion



Bibliographie

- Rolf Dietze. *SunCluster. Serververfügbarkeit unter Solaris*. Springer-Verlag, Berlin, Heidelberg, New York, August 2009. ISBN 3-540-33805-5.
- Rolf Dietze, Tatjana Heuser, and Jörg Schilling. *OpenSolaris für Anwender, Administratoren und Rechenzentren*. Springer-Verlag, Berlin, Heidelberg, New York, March 2006. ISBN 3-540-29236-5.
- Frank G. Soltis. *Fortress Rochester: The Inside Story of the IBM iSeries*. 29th Street Press, 2001. ISBN 1-58304-083-8.
- Sun 817-1592. *Solaris Containers-Resource Management and Solaris Zones*, June 2005. Sun Microsystems Inc., Solaris 10 Release.
- Sun 817-1975-11. *Solaris 10 Resource Manager Developer's Guide*, December 2005.
- <http://opensolaris.org/os/project/crossbow>