1

# Herramientas de Desarrollo



## Desarrollo

2

### *Objetivos*

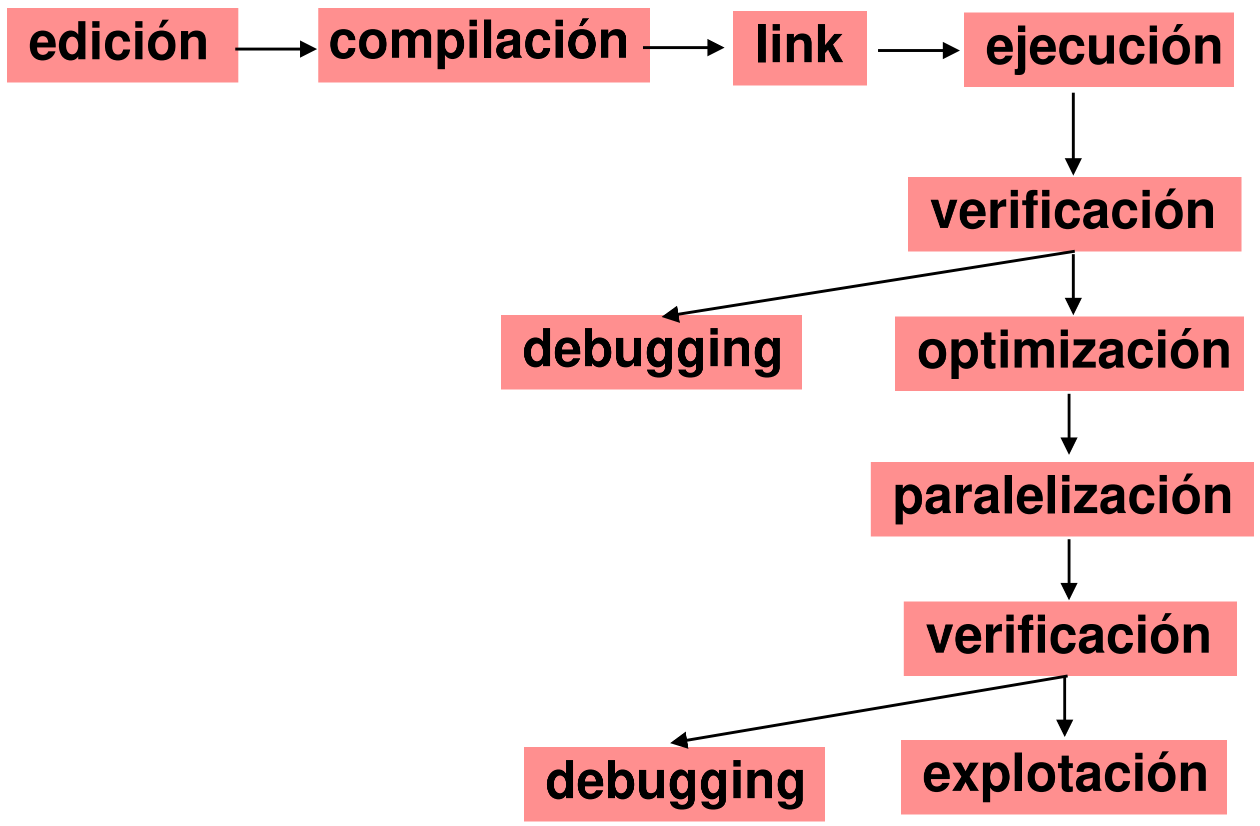
**Metodología y herramientas para el desarrollo de código:**

* **Herramientas de gestión de fuentes**
* **Herramientas de ejecución y timing**
* **Herramientas de Profiling**
* **Herramientas de Monitorización**

**El empleo de estas herramientas ahorra tiempo y fustración**

# Fases de Desarrollo








# Edición



- vi (standard Unix)
- Jot (sólo con emulación GL)
- nedit (emulación X)
- Emacs (emulación X)
  - versión freeware y comercial
- otros freeware (EDT, ne, ...)

Otra opción consiste en editar en local desde workstation o PC
y mandarlo a compilar al server (p.e. a través de NQS).

# Herramientas de Gestión de código fuente

5

## *Ficheros fuente*

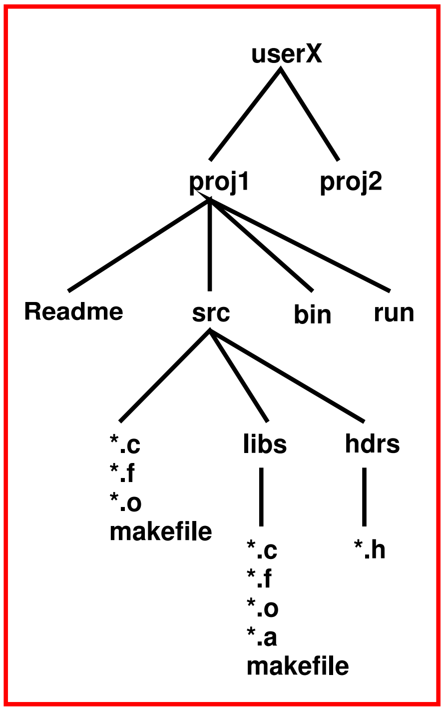
* Organizarse en árbol

* Módulos pequeños:

- * usar: fsplit / split
- * compilación más rápida
- * menos esfuerzo de compilación
- * mejor estudio del código
- * flexibilidad de optimización

* Documentación

- * sccs/rcs
- * Make
- * Readme






# Make

6

* documentación implícita

* Minimiza las compilaciones

* 100% "oops free"

* make / pmake / smake

```
OBJS= file1.o file2.o file3.o

FFLAGS= –O3 –mips4 –64
LDFLAGS= –lm –mips4 –64

all:        pgm1

pgm1:       $(OBJS)
<tab>       f77 –o pgm1 $(OBJS) $(LDFLAGS)

file2.o:    file2.f
<tab>       f77 $(FFLAGS) –static –c file2.f

clean:
<tab>       –rm –f $(OBJS) pgm1 core
```

# Revision Control System (RCS)

7

* Automatic Documentation
* Roll Back
* 100% "oops free"
* ci / co / rlog

```
mkdir RCS
ci file1
RCS/file1,v  <--  file1
enter description, terminated with single '.' or end of file:
NOTE: This is NOT the log message!
>> This is my first version
>> ^D
done
rlog file1
co file1
RCS/file1,v  -->  file1
revision 1.1
done
ls -l file1
-r--r--r--    1 benoit   nuucp         5 Dec 29 15:37 file1
```



# Revision Control System (RCS)

8

## ***RCS Commands***

* creation: (mkdir RCS) ci <file>
* versions: rlog <file>
* add new: ci <file>
* retrieve: co –r<version> <file>
* edit: co –l<version> <file>
* diff: rcsdiff –r<version> –r<version> <file>

# SCCS

9

## ***SCCS***

* Automatic Documentation
* Roll Back
* 100% "oops free"
* admin / get / delta / prs
* Make Integrated

admin –ifile1.c s.file1.c
make –g t



# SCCS

1 0

## ***SCCS Commands***

* creation: admin –i<file> s.<file>
* versions: prs s.<file>
* add new: delta s.<file>
* retrieve: get –r<version> s.<file>
* edit: get –e –r<version> s.<file>
* diff: sccsdiff –r<version> –r<version> s.<file>

## Compilación y link

11

f77 programa.f –o programa

cc programa.c –o programa

f77 –O3 principal.f rut1.f rut2.f –o ejecutable –lm

cc –O0 –g main.c mod1.c –o prog.debug

f77 –Ofast=ip27 –OPT:IEEE_aritmetic=3 –lfastm –lm



## Ejecución

12

> programa — el direc "." ha de estar en el path, sino hacer: set path = (. $path) o bien ./programa

verificar las opciones del comado limit (csh, tcsh) o ulimit –a (ksh):

```
cputime          unlimited
filesize         unlimited
datasize         524288 kbytes
stacksize        65536 kbytes
coredumpsize     0 kbytes
memoryuse        121732 kbytes
descriptors      200
vmemoryuse       524288 kbytes

time(seconds)          unlimited
file(blocks)           unlimited
data(kbytes)           524288
stack(kbytes)          65536
memory(kbytes)         121732
coredump(blocks)       0
nofiles(descriptors)   200
vmemory(kbytes)        524288
```

> runon 3 programa — ejecuta programa en la cpu num. 3 (sino ha sido restringida).

# Timing

13

> time programa <– da los tiempos de user sys y total de la ejecución

> timex programa <– da los tiempos de user sys y total de la ejecución y mas info adicional

```
time du -ks /usr
1126051 /usr
0.3u 5.5s 0:28 20% 0+0k 2901+155io 0pf+0w


timex -s date
Tue Jan  4 10:39:59 MET 1994

real        0.07
user        0.01
sys         0.03


IRIX dope 5.1.1.1 10011611 IP20    01/04/94

10:39:59  %usr  %sys %intr  %wio %idle %sbrk  %wfs %wswp %wphy %wgsw %wfif
10:40:00    10    70     0    10    10     0   100     0     0     0     0

10:39:59  device   %busy  avque  r+w/s  blks/s   w/s wblks/s  avwait  avserv
10:40:00   dks0d1     30    1.0     15     360     5     160     0.0    20.0
           dks0d2      0    0.0      0       0     0       0     0.0     0.0

 (etc...)
```



# Rutina de tiempos

14

## prog. ejemplo

```
int main(int argc,char *argv[])
{
    double ia, cia, scia;
    double  ta, tb;
    int i;
    ta = tiempo();

    scia = 0.0;
    for (i=1;i<1000000;i++)
    {
        ia = i * i;
        cia = cos (ia) * sin(ia) * tan (ia);
        scia = scia + cia;
    }

    tb = tiempo();
    printf("\n\nTIEMPO:%g\n",tb-ta);

}
```

## rut. tiempo

```
#include <stdio.h>
#include <sys/times.h>
#include <unistd.h>
#include <math.h>

double
tiempo() {
  struct tms ticks;
  long clktck;
  double tiempo;

  clktck=sysconf( _SC_CLK_TCK );
  times(&ticks);
  tiempo=(double)ticks.tms_utime/clktck;

  return (tiempo);
}
```

# Ejecución en paralelo

15

– cuando se ejecuta una aplicación que ha sido paralelizada, si no se especifica nada se ejecuta sobre todas las cpu's del sistema, si es que este límite no ha sido restringido previamente por el administrador

– para controlar el número de threads de ejecución se emplea la variable de entorno:

MP_SET_NUMTHREADS

– también se puede especificar el tipo de scheduling de las threads y el tamaño de los chuncks de trabajo con:

MP_SCHEDTYPE
MP_CHUNK



# Profiling

16

***Profiling***

* Traza de la ejecución

* Para medir la eficiencia de la optimización, la posibilidad de patologías (cache trashing, flase sharing,...) o simplemente para obtener la distribución de tiempos por rutinas o líneas.

3 maneras:

– PC–sampling: muestreo estadístico

– Basic Bloc: tiempo ideal por bloque bloque

– SpeedShop (6.2 en adelante, nuevo)

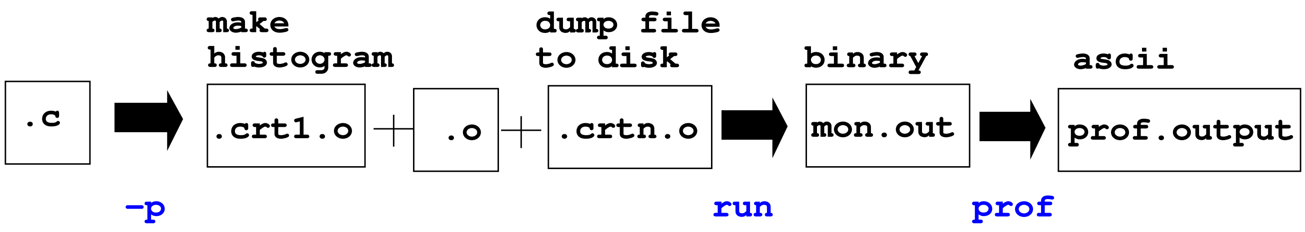
# PC – sampling

17

**# cc –p –o pgm  *.o          (o f77 –p –o pgm)**

**# pgm**

**# prof [–heavy]  pgm > prof.output**






# PC–sampling output example

18

```
-------------------------------------------------------------------------------
Profile listing generated Sat Feb 10 15:43:16 1996
    with:       prof -lines ./pgm
-------------------------------------------------------------------------------

samples   time    CPU    FPU   Clock    N-cpu  S-interval Countsize
   6289    63s  R4000  R4010 175.0MHz   1      10.0ms      4(bytes)

Each sample covers 4 bytes for every 10.0ms ( 0.02% of 62.8900s)


-------------------------------------------------------------------------------
  -p[rocedures] using pc-sampling.
  Sorted in descending order by the number of samples in each procedure.
  Unexecuted procedures are excluded.
-------------------------------------------------------------------------------

samples   time(%)      cum time(%)      procedure (file)

   4799    48s( 76.3)   48s( 76.3)       tdma2d_ (./pgm:tdma2d.f)
    999    10s( 15.9)   58s( 92.2)       scoeff_ (./pgm:scoeff.f)
    280   2.8s(  4.5)   61s( 96.6)        norma_ (./pgm:norma.f)
    184   1.8s(  2.9)   63s( 99.6)        resid_ (./pgm:resid.f)
     22  0.22s(  0.3)   63s( 99.9)       inisol_ (./pgm:inisol.f)
      3  0.03s(  0.0)   63s(100.0)       grdgen_ (./pgm:grdgen.f)
      1  0.01s(  0.0)   63s(100.0)        _write (/usr/lib/libc.so.1:write.s)
      1  0.01s(  0.0)   63s(100.0)        fflush (/usr/lib/libc.so.1:flush.c)

   6289    63s(100.0)   63s(100.0)          TOTAL
```

19

# Basic Block counting

**cc –o pgm *.o**

**pixie pgm**

**setenv LIBRARY_PATH .**

**pgm.pixie**

**prof –pixie [–gprof] [–heavy] pgm > pixie.output**

| binary executable | | instrumented binary exec. | | binary res. | | ascii |
|---|---|---|---|---|---|---|
| pgm | $\xrightarrow{\text{pixie}}$ | pgm.pixie | $\xrightarrow{\text{run}}$ | pgm.Counts | $\xrightarrow{\text{prof}}$ | pixie.output |



20

# Basic Block ("pixie")

```
 Prof run at: Sat Feb 10 15:47:25 1996
Command line: prof -lines -pixie ./pgm


        1924411021: Total number of cycles
         10.99663s: Total execution time
        1043450615: Total number of instructions executed
             1.844: Ratio of cycles / instruction
               175: Clock rate in MHz
             R4000: Target processor modelled


---------------------------------------------------------
Procedures sorted in descending order of cycles executed.
Unexecuted procedures are not listed. Procedures
beginning with *DF* are dummy functions and represent
init, fini and stub sections.
---------------------------------------------------------

        cycles(%)  cum %     secs    instrns    calls procedure(dso:file)

 856119356(44.49)  44.49     4.89  549667545    50 tdma2d_(pgm:tdma2d.f)
 603861460(31.38)  75.87     3.45  301034270    10 scoeff_(pgm:scoeff.f)
 309978810(16.11)  91.97     1.77   72215620    10 norma_(pgm:norma.f)
 148484476( 7.72)  99.69     0.85  116216406    10 resid_(pgm:resid.f)
   3494750( 0.18)  99.87     0.02    2602792     1 inisol_(pgm:inisol.f)
   2303239( 0.12)  99.99     0.01    1596774     1 grdgen_(pgm:grdgen.f)
     29793( 0.00)  99.99     0.00      20365   390 __flsbuf(/usr/lib/libc.so.1:_flsbuf.c)
     26279 ( 0.00) 99.99     0.00      20082   330 x_putc(/usr/lib/libftn.so:wsfe.c)
```

# Hardware Counter Registers



- **System call interface**

  man `r10k_counters(5)`

- **Timex-like interface** (`perfex(1M)`)

  ```
  % perfex [options] command [arguments]
  ```

- **libperfex**

  ```
  ff77|cc -o program <objects> -lperfex

  man ssapi(3), libperfex
  ```



# Hardware Counter Registers



Cntr 0

```
Issued instructions.........................................
Issued loads................................................
Issued stores...............................................
Issued store conditionals...................................
Failed store conditionals...................................
Decoded branches............................................
Quadwords written back from scache..........................
Correctable scache data array ECC errors....................
Primary instruction cache misses............................
Secondary instruction cache misses..........................
Instruction misprediction from scache way prediction table...
External interventions......................................
External invalidations......................................
Virtual coherency conditions................................
Graduated instructions......................................
Cycles......................................................
```

Cntr 1

```
Graduated instructions......................................
Graduated loads.............................................
Graduated stores............................................
Graduated store conditionals................................
Graduated floating point instructions.......................
Quadwords written back from primary data cache..............
TLB misses..................................................
Mispredicted branches.......................................
Primary data cache misses...................................
Secondary data cache misses.................................
Data misprediction from scache way prediction table.........
External intervention hits in scache........................
External invalidation hits in scache........................
Store/prefetch exclusive to clean block in scache...........
Store/prefetch exclusive to shared block in scache..........
```

# SpeedShop tools

2 3

SpeedShop funciona sobre 6.2 y posteriores y soporta ejecutables compilados en 6.2 (o32, n32 y 64), o con los MIPS Pro7.0 (n32 y 64). Soporta C, C++, f77 y f90, ADA y asm. Los programas deben utilizar shared libraries (dso’s), los non_shared no están soportados.

**perfex**: ejecuta un programa y reporta dos contadores (uno de cada grupo a elegir).

Alternativamente tambien puede multiplexar los 32 contadores y extrapola los totales. Es útil para identificar qué problema existe (p.e. sec. cache misses).

**ssrun**: ejecuta un programa haciendo un muestreo del estado del PC y del stack.

Escribe los resultados sobre un fichero para su posterior análisis (prof). Se elige el periodo de mustreo y el tipo de información a muestrear. Es útil para localizar dónde está el problema.



# Perfex results

2 4

```
perfex -a -x -y pion > $$_pion.perfex.log
```

Based on 195 MHz IP27

| | Event Counter Name | Counter Value | Typical Time (sec) | Minimum Time (sec) | Maximum Time (sec) |
|---|---|---|---|---|---|
| 0 | Cycles | 141173020032 | 723.964205 | 723.964205 | 723.964205 |
| 16 | Cycles | 141173020032 | 723.964205 | 723.964205 | 723.964205 |
| 2 | Issued loads | 37268484880 | 191.120435 | 191.120435 | 191.120435 |
| 18 | Graduated loads | 31149178384 | 159.739376 | 159.739376 | 159.739376 |
| 21 | Graduated floating point instructions | 25979331440 | 133.227341 | 66.613670 | 6927.821717 |
| 9 | Primary instruction cache misses | 1069342400 | 98.818205 | 30.873834 | 98.818205 |
| 3 | Issued stores | 16479530304 | 84.510412 | 84.510412 | 84.510412 |
| 19 | Graduated stores | 15698867104 | 80.507011 | 80.507011 | 80.507011 |
| 6 | Decoded branches | 13902875328 | 71.296797 | 71.296797 | 71.296797 |
| 25 | Primary data cache misses | 302157408 | 13.961222 | 4.369661 | 13.961222 |
| 24 | Mispredicted branches | 1409072704 | 10.260940 | 4.624649 | 37.719792 |
| 22 | Quadwords written back from primary data cache | 210691808 | 4.159813 | 3.392678 | 4.808095 |
| 7 | Quadwords written back from scache | 3505712 | 0.115059 | 0.076047 | 0.115059 |
| 26 | Secondary data cache misses | 87168 | 0.033750 | 0.022065 | 0.037549 |
| 10 | Secondary instruction cache misses | 16176 | 0.006263 | 0.004095 | 0.006968 |
| 31 | Store/prefetch exclusive to shared block in scache | 1168 | 0.000006 | 0.000006 | 0.000006 |
| 1 | Issued instructions | 185437324528 | 0.000000 | 0.000000 | 950.960639 |
| 4 | Issued store conditionals | 0 | 0.000000 | 0.000000 | 0.000000 |
| 5 | Failed store conditionals | 0 | 0.000000 | 0.000000 | 0.000000 |
| 8 | Correctable scache data array ECC errors | 0 | 0.000000 | 0.000000 | 0.000000 |
| 11 | Instruction misprediction from scache way prediction table | 6122480 | 0.000000 | 0.000000 | 0.031397 |
| 12 | External interventions | 483312 | 0.000000 | 0.000000 | 0.000000 |
| 13 | External invalidations | 1251040 | 0.000000 | 0.000000 | 0.000000 |
| 14 | Virtual coherency conditions | 0 | 0.000000 | 0.000000 | 0.000000 |
| 15 | Graduated instructions | 166773430944 | 0.000000 | 0.000000 | 855.248364 |
| 17 | Graduated instructions | 166811652400 | 0.000000 | 0.000000 | 855.444371 |
| 20 | Graduated store conditionals | 0 | 0.000000 | 0.000000 | 0.000000 |
| 23 | TLB misses | 0 | 0.000000 | 0.000000 | 0.000000 |
| 27 | Data misprediction from scache way prediction table | 8150768 | 0.000000 | 0.000000 | 0.041799 |
| 28 | External intervention hits in scache | 483280 | 0.000000 | 0.000000 | 0.000000 |
| 29 | External invalidation hits in scache | 668832 | 0.000000 | 0.000000 | 0.000000 |
| 30 | Store/prefetch exclusive to clean block in scache | 0 | 0.000000 | 0.000000 | 0.000000 |

# Perfex statistics

25

```
Statistics
=========================================================================================
Graduated instructions/cycle................................................    1.181341
Graduated floating point instructions/cycle.................................    0.184025
Graduated loads & stores/cycle..............................................    0.331848
Graduated loads & stores/floating point instruction.........................    2.068876
Mispredicted branches/Decoded branches......................................    0.101351
Graduated loads/Issued loads................................................    0.835805
Graduated stores/Issued stores..............................................    0.952628
Data mispredict/Data scache hits............................................    0.026983
Instruction mispredict/Instruction scache hits..............................    0.005726
L1 Cache Line Reuse.........................................................  154.045166
L2 Cache Line Reuse......................................................... 3465.379956
L1 Data Cache Hit Rate......................................................    0.993550
L2 Data Cache Hit Rate......................................................    0.999712
Time accessing memory/Total time............................................    0.351179
L1--L2 bandwidth used (MB/s, average per process)...........................   18.012087
Memory bandwidth used (MB/s, average per process)...........................    0.092890
MFLOPS (average per process)................................................   35.884829
```



# SpeedShop ssrun experiments

26

**ssusage**: ~time

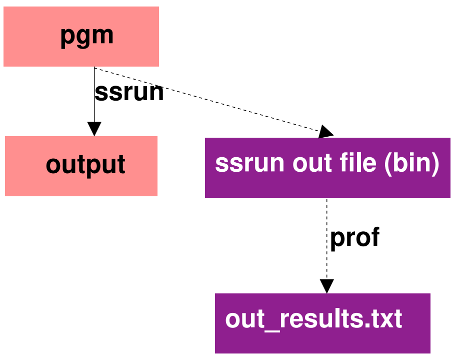
ex: # ssusage prog [arg1 arg2 ... argn ]

1.89 real, 0.04 user, 0.29 sys, 15 majf, 2 minf, 0 sw, 1 rb, 0 wb, 21 vcx, 38 icx

**ssrun**: run speedshop experiments

ex: # ssrun <flags> –<exp_type> pgm [arg1 arg2 ... argn ]

# prof <flags> <out_ssrun_file> > out_results.txt




usertime
[f]pcsamp[x]
ideal
fpe

En sistemas con HW performance counters, (R10000 systems) se soportan además los siguientes tipos de experimentos:

[f]gi_hwc
[f]cy_hwc
[f]ic_hwc
[f]isc_hwc
[f]dc_hwc
[f]dsc_hwc
[f]tlb_hwc
[f]gfp_hwc
prof_hwc
heap

para más info: man speeshop

27

## *Metodología de optimización*

* Optimizar las rutinas de una en una (no todo a la vez).

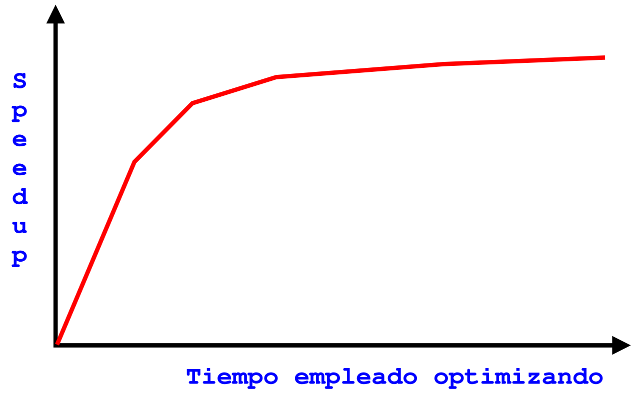
* Empezar por la rutina que gasta más tiempo

* Después la segunda...

* Después la tercera...

* etc...

* parase cuando la mejora marginal no sea significativa








28

## *Herramientas de monitorización*

* gr_osview

* osview

* top

* gr_top

* gmemusage

* ps

* par

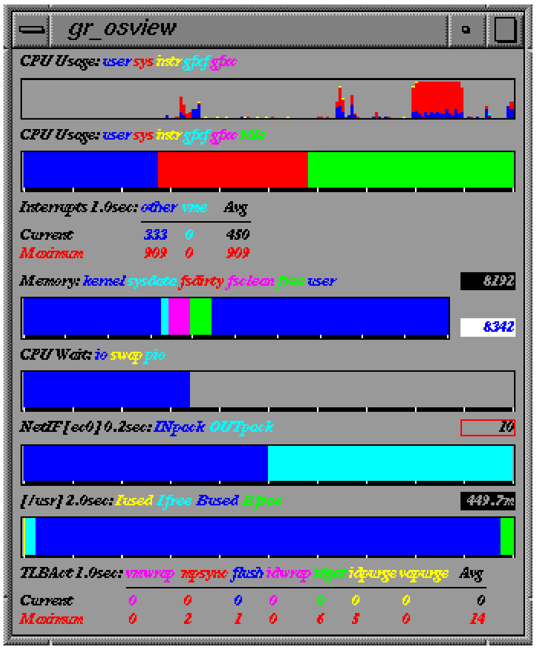
# Graphical Resource Utilization Overview

2 9

## *gr–osview*

cpu strip
cpu
intr numeric max tracksum(10)
rmem tracksum
wait
netif
disk(/usr)
tlb numeric max tracksum(1)

customization in: .gr_osview






# ASCII based Resource Utilization

3 0

## *osview*

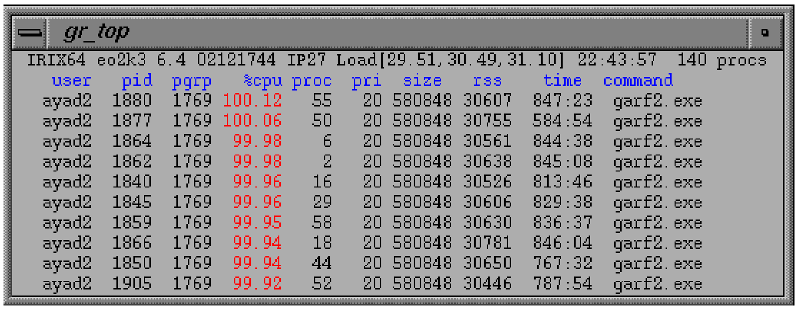
```
Osview 2.1 : One Second Average    dope     01/04/94 10:28:51 #0     int=5s
Load Average            onfile        30    rcvbyte         0
  1 Min      0.658      freed          0  UDP
  5 Min      0.466      unmodswap      0    ipackets        0
  15 Min     0.218      unmodfile      0    opackets        0
CPU Usage             Block Devices         dropped         0
  %user       6.00      lread      144.0K   errors          0
  %sys        4.00      bread      715.0K  IP
  %intr       2.00      %rcache    -396.5   ipackets        0
  %gfxc       0.00      lwrite      17.0K   opackets        0
  %gfxf       0.00      bwrite         0    forward         0
  %sxbrk      0.00      %wcache    100.0    dropped         0
  %idle      88.00      phread         0    errors          0

(etc...)
```

(Usar "h,j,k,l" para desplazarse y "D,I" para Delete e Insert una section.)

# top / gr_top

31

```
IRIX64 itv 6.4 02121744 IP27 Load[0.01,0.00,0.00] 22:28:07   81 procs
    user   pid  pgrp  %cpu proc  pri  size   rss    time  command
    root  1456  1456  0.47    *   20   200    42    8:56  pmcd
  atrill 24706 24706  0.44    3   20   116    73    0:00  top
   guest 18356 18354  0.12    *   20   116    64    2:31  rdistd
    root  1404  1404  0.03    *   20   141    68    3:22  mediad
    root   463     0  0.01    *   20   115    47    1:26  mmscd
    root   485   485  0.01    *   32   116   116    1:01  xntpd
    root  1408   359  0.01    *   20   122    47    0:21  fam
  benoit 23195 23194 99.57   22   20  5159  5056   26:21  bse3Dsecd
  benoit 23654 23651 99.62   12  g20  5331  5100    7:09  bse3Dsecd_ben
  benoit 23653 23651 99.55   13  g20  5331  5100    7:10  bse3Dsecd_ben
   ayad2  1866  1769 99.61   18   20 580850 30853 2963:47  garf2.exe
   ayad2  1912  1769 99.57    0   20 580848 30541 2965:10  garf2.exe
    root 24695 24695  0.01    *   20    96    56    0:00  rlogind
    root   333     0  0.00    *  +67     0     0    7:52  nfsd
    root   332     0  0.00    *  +67     0     0    7:46  nfsd
    root   322     0  0.00    *  +67     0     0    7:51  nfsd
    root   331     0  0.00    *  +67     0     0    7:54  nfsd
```


gr_top

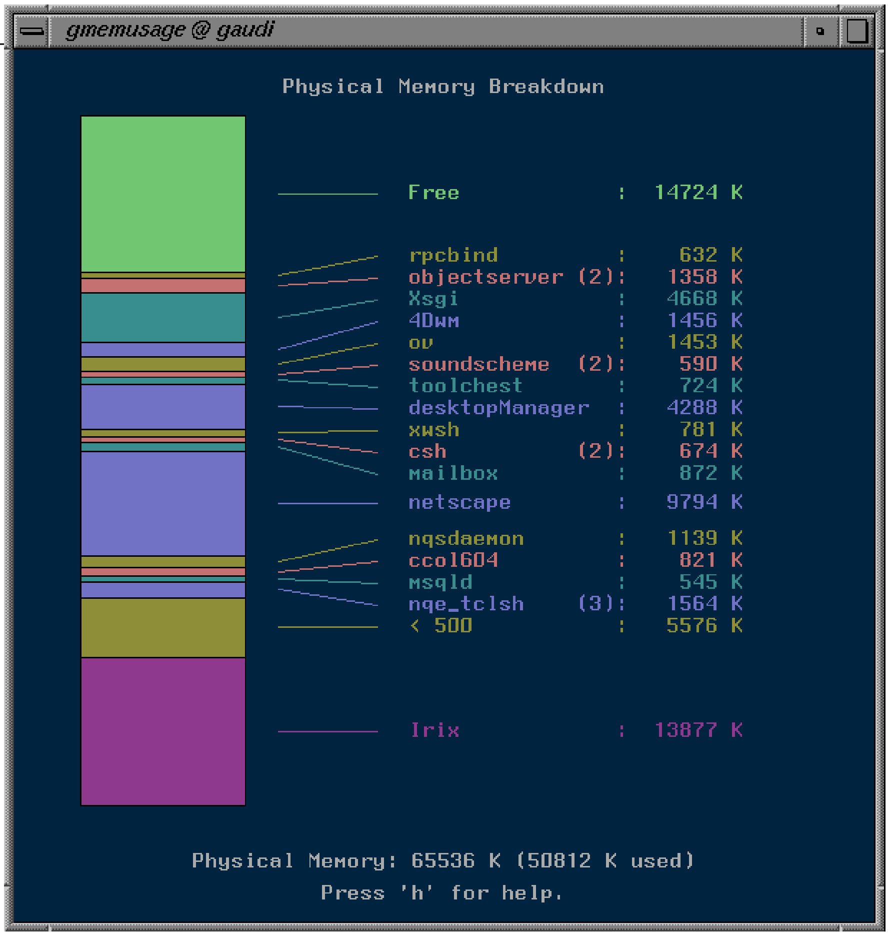
IRIX64 eo2k3 6.4 02121744 IP27 Load[29.51,30.49,31.10] 22:43:57 140 procs

| user | pid | pgrp | %cpu | proc | pri | size | rss | time | command |
|---|---|---|---|---|---|---|---|---|---|
| ayad2 | 1880 | 1769 | 100.12 | 55 | 20 | 580848 | 30607 | 847:23 | garf2.exe |
| ayad2 | 1877 | 1769 | 100.06 | 50 | 20 | 580848 | 30755 | 584:54 | garf2.exe |
| ayad2 | 1864 | 1769 | 99.98 | 6 | 20 | 580848 | 30561 | 844:38 | garf2.exe |
| ayad2 | 1862 | 1769 | 99.98 | 2 | 20 | 580848 | 30638 | 845:08 | garf2.exe |
| ayad2 | 1840 | 1769 | 99.96 | 16 | 20 | 580848 | 30526 | 813:46 | garf2.exe |
| ayad2 | 1845 | 1769 | 99.96 | 29 | 20 | 580848 | 30606 | 829:38 | garf2.exe |
| ayad2 | 1859 | 1769 | 99.95 | 58 | 20 | 580848 | 30630 | 836:37 | garf2.exe |
| ayad2 | 1866 | 1769 | 99.94 | 18 | 20 | 580848 | 30781 | 846:04 | garf2.exe |
| ayad2 | 1850 | 1769 | 99.94 | 44 | 20 | 580848 | 30650 | 767:32 | garf2.exe |
| ayad2 | 1905 | 1769 | 99.92 | 52 | 20 | 580848 | 30446 | 787:54 | garf2.exe |



# gmemusage

32



# ps

33

```
< 101 atrill@eo2k3:/itv/global3/atrill > ps -efl
  F S      UID   PID  PPID  C PRI NI  P    SZ:RSS      WCHAN    STIME TTY      TIME CMD
 b0 S     root     1     1  0  20  0  *     40:37    81100b0 07:53:37 ?        0:19 /etc/init
 b0 R     root     2     1  0  20  0 35      0:0           - 07:53:37 ?      298:48 coalesced
 b0 S     root   108     1  0  20  0  *    101:59   932fe8d8 07:53:42 ?        0:00 /sbin/xlv_plexd -m 4 -w 1
 b0 S     root   112     1  0 176 RT  *   156:156    14195e0 07:53:43 ?        0:00 /sbin/xlv_labd
 b0 S     root   172     1  0  20  0  *     96:56     dec560 07:53:51 ?        0:00 /usr/etc/syslogd
 b0 S     root   175     1  0  19  0  *     94:53    5d8c0b0 07:53:51 ?        0:00 /usr/sbin/cached
 b0 S     root   327     1  0  20  0  *    101:56     deca38 07:53:59 ?        0:07 /usr/etc/routed -h -Prdisc_interval
 b0 S     root   338     1  0  20  0  *    137:87     deca90 07:54:00 ?        0:00 /usr/etc/rpcbind
 b0 S     root   339     1  0  20  0  *     94:52     deca98 07:54:05 ?        0:00 /usr/etc/ypbind -ypsetme
 b0 S     root   357     1  0  67 RT  *      0:0     13c9a48 07:54:06 ?        0:01 /usr/etc/nfsd 4
 b0 S     root   360     1  0  68 RT  *      0:0     13caad0 07:54:06 ?        0:00 /usr/etc/biod 4
 b0 S     root  1736     1  0  20  0  *   181:122   9a3558b0 07:54:45 ?        1:08 /usr/craysoft/nqe/3.2.0.2/bin/nqe_t
 b0 S     root  1741   401  0  20  0  *     96:58     def668 07:54:46 pts/0    0:00 rlogind
 b0 S    ayad2  1742  1741  0  20  0  *   148:109   946a00b0 07:54:46 pts/0    0:00 -tcsh
 b0 S     root  1746     1  0  20  0  *   184:125   c383c0b0 07:54:52 ?        1:41 /usr/craysoft/nqe/3.2.0.2/bin/nqe_t
 b0 S    ayad2  1769  1742  0  20  0  *     28:19   99ea0368 07:55:31 pts/0    0:00 /bin/sh @sunday
 b0 S    ayad2  1770  1769  0  20  0  *    140:88   9c7698b0 07:55:31 pts/0    0:00 /bin/tcsh -f /lv2/ayad/adnoc/ecl/ma
 b0 S    ayad2  1798     1  0  20  0  * 558635:9182    def830 07:55:32 pts/0    1:33 /lv2/ayad/adnoc/SGI/pvm/pvm3.3.11_7
 b0 R    ayad2  1836  1770  0  20  0  9 580854:30607        - 07:55:34 pts/0  870:18 /lv2/ayad/adnoc/SGI/e300/source2/ga
 b0 S    ayad2  1837  1798  0  20  0  * 580848:30576   3d5928 07:55:36 pts/0  31:33 /lv2/ayad/adnoc/SGI/e300/source2/ga
 b0 S    ayad2  1838  1798  0  20  0  * 580848:30578   3d5968 07:55:36 pts/0  40:24 /lv2/ayad/adnoc/SGI/e300/source2/ga
 b0 S    ayad2  1839  1798  0  20  0  * 580848:30527  4ed52c8 07:55:36 pts/0  39:42 /lv2/ayad/adnoc/SGI/e300/source2/ga
 b0 R   benoit 23653 23651  0  20 GN 12  5331:5100          - 10:26:36 pts/10  5:40 ./bse3Dsecd_ben
 b0 R   benoit 23654 23653  0  20 GN 13  5331:5100          - 10:26:37 pts/10  5:39 ./bse3Dsecd_ben
...
```



# par

34

```
par -irsSS ls
Appendix_tools.tex     atrill.home          esquema_O2.jpg         msdocs                snap.rgb
...
     0mS       par(  832): was sent signal SIGUSR1
     1mS       par(  832): END-pause() errno = 4 (Interrupted function call)
     1mS       par(  832): received signal SIGUSR1
     1mS       par(  832): sigreturn(0x7fff2b58) OK
     2mS       par(  832): execve(./ls, 0x7fff2f4c, 0x7fff2f54) errno = 2 (No such file or directory)
     3mS       par(  832): execve(/usr/sbin/ls, 0x7fff2f4c, 0x7fff2f54) errno = 2 (No such file or directory)
     3mS       par(  832): execve(/usr/bsd/ls, 0x7fff2f4c, 0x7fff2f54) errno = 2 (No such file or directory)
     4mS        ls(  832): execve(/sbin/ls, 0x7fff2f4c, 0x7fff2f54) OK
     6mS        ls(  832): open(/lib/rld, O_RDONLY, 04) = 3
     6mS        ls(  832): read(3, <7f 45 4c 46 01 02 01 00 00 00 00 00 00 00 00 00>..., 512) = 512
     6mS        ls(  832): elfmap(3, 0x7fff2d38, 2) = 0xfb60000
     6mS        ls(  832): close(3) OK
     7mS        ls(  832): getpagesize() = 4096
...
    13mS        ls(  832): fstat64(3, 0x7fff2c00) OK
    13mS        ls(  832): brk(0x10011000) OK
    13mS        ls(  832): ngetdents64(3, 0x1000c288, 16384, 0x7fff2c94) = 4024
    14mS        ls(  832): close(3) OK
    14mS        ls(  832): ioctl(1, TCGETA, 0x7fff1208) = 0
    15mS        ls(  832): write(1, "Appendix_tools.tex     atrill.", 101) = 101
  ...
    17mS        ls(  832): write(1, "Silicon_News.sc.old    desktop", 77) = 77
    17mS        ls(  832): prctl(PR_GETNSHARE) = 0
    17mS        ls(  832): exit(0)
exit                : 1 times
read                : 1 times
write               : 9 times
open                : 4 times
close               : 4 times
time                : 1 times
brk                 : 3 times
syssgi              : 5 times
ioctl               : 3 times
sysmp               : 2 times
execve              : 4 times
fcntl               : 1 times
sigreturn           : 1 times
prctl               : 1 times
mmap                : 2 times
lxstat              : 1 times
fxstat              : 1 times
ngetdents64         : 1 times
```

# Herramientas de monitorización – System

35

## *Herramientas de monitorización*

* sar
* PCP (performance Co–Pilot)
* accounting
* nstats
* xbstat


Copyright: SiliconGraphics


# System Activity Reporter (SAR)

36

## *Sar*

| | |
|---|---|
| * –u | (cpu) |
| * –b | (buffer) |
| * –d | (disks) |
| * –w | (swap & switch) |
| * –q | (run queues) |
| * –p | (paging) |
| * –r | (virtual memory) |
| * –I | (interrupts) |
| * (etc...) | |
| * –A | (everything) |

```
sar -r 1 3

IRIX 5.1.1.1 IP20 01/04/94

10:34:53 freemem freeswp
10:34:54     323   36752
10:34:56     320   36752
10:34:57     321   36752


10:34:57 freemem freeswp
Average      321   36752
```


Copyright: SiliconGraphics

# Resumen de soluciones SW

37

### *Compilers*

C, PowerC
C++
F77, PowerF77
F90, PowerF90

ADA83
ADA95
PASCAL

### *GUI tools*

ViewKit
RappidApp

### *CASE*

WorkShop
- Debugger gráfico
- Static Analizer
- Performance Analizer
- Builder

ProMPF
ProADA
SpeedShop
ClearCASE
Tracker

### *Libraries*

REACTpro
OpenGL/IRIS–GL
OpenInventor
IRIS Performer
Imagevision
DigitalMedia
Impressario
Device Driver
Complib
SCSL


Copyright: SiliconGraphics