

Introducción a pmlib v2.0

Manuel F. Dolz *et al*



UNIVERSITAT
JAUME I

Green HPC_{IA} 
High Performance Computing
and Architectures

3 de julio de 2012

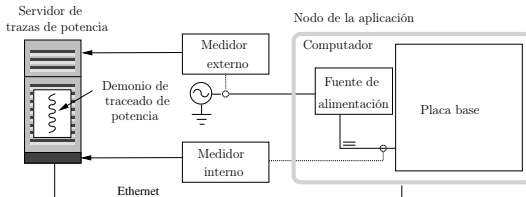
Índice

- 1 ¿Qué es pmlib?
 - Esquema general
 - Dispositivos de medición de potencia
 - Funcionamiento del demonio en el servidor
- 2 Rutinas y herramientas pmlib
 - Rutinas básicas biblioteca pmlib
 - Comando de consulta pm_info
- 3 Ejemplos de uso

¿Qué es pmlib?

Biblioteca pmlib

- Paquete de medición de la Universidad Jaume I
- Interfaz para interactuar y utilizar nuestros medidores de potencia
- También compatible con medidores de potencia comerciales



- **Demonio servidor:** recolecta los datos de potencia de los medidores y los envía a los clientes.
- **Biblioteca cliente:** habilita la comunicación con el servidor a través de una serie de primitivas básicas para recolección de datos de potencia.

Dispositivos de medición de potencia

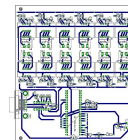
- **Dispositivos internos:** miden la potencia disipada por los componentes de la placa base

- Dispositivo interno basado en microcontrolador

- Transductores LEM HXS 20-NP con microcontrolador PIC
- Frecuencia de muestreo: de 25 Hz a 100 Hz
- Puerto serie RS232

- Dispositivo National Instruments, tarjeta de adquisición de datos:

- NI9205 / cDAQ-9178
- Frecuencia de muestreo: 7 KHz!
- Puerto USB



- **Dispositivos Externos:** Miden la potencia global de la máquina

- WattsUp? Pro .NET

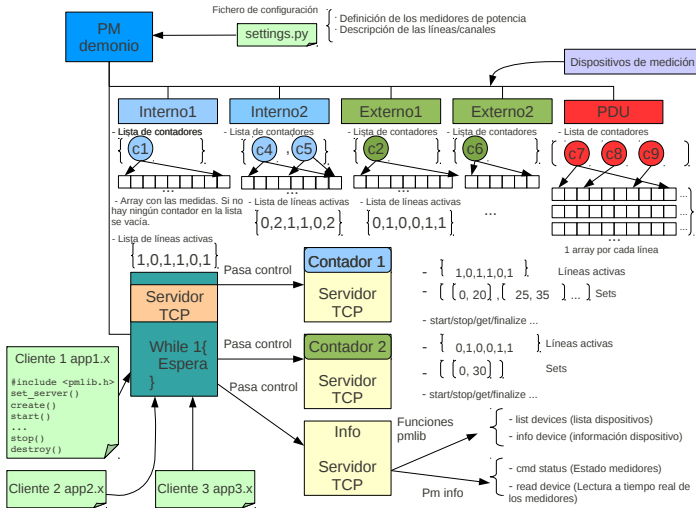
- Frecuencia de muestreo: 1 Hz
- Sólo 1 enchufe!
- Puertos USB/Ethernet

- Unidad de distribución de potencia APC 8653

- Frecuencia de muestreo: 1 Hz
- 24 enchufes
- SNMP/ssh vía Ethernet



Funcionamiento del demonio en el servidor



Gestión del demonio en el servidor

Fichero de configuración settings.py:

- Dirección IP y puerto donde escucha el servidor
- Ruta del fichero LOG
- Definición de “computadoras”
- Definición de “dispositivos”
- Definición de “líneas”

Ejemplo de inicio del demonio:

Puesta en marcha y parada del demonio en el servidor

```
1 $ pm_server --start
2 Starting devices...
3 Starting device DCMeter1...      [ OK ]
4 Starting device DCMeter2...      [ OK ]
5 Starting device PDU...           [ OK ]
6 Starting device WattsUp2...      [ OK ]
7 Starting server...               [ OK ]
8 Server listening at ('0.0.0.0', 6526)
9
10 $ pm_server --stop
11 Stopping devices...
12 Stopping device DCMeter1...      [ OK ]
13 Stopping device DCMeter2...      [ OK ]
14 Stopping device PDU...           [ OK ]
15 Stopping device WattsUp2...      [ OK ]
16 Stopping server...               [ OK ]
```

Rutinas básicas biblioteca pmlib (I)

Estructuras:

- `server_t` Tipo de datos "servidor" (IP y puerto)
- `device_t` Tipo de datos "dispositivo" (nombre, frecuencia y núm. de líneas)
- `line_t` Tipo de datos "línea" (máscara de bits que representa líneas)
- `counter_t` Tipo de datos "contador" (tiempos y medidas de potencia)

Rutinas:

- `int pm_set_server(char *svrip, int port, server_t *svr)`
✓ *Inicializa el servidor con la IP y puerto para la comunicación con el servidor*
- `int pm_get_devices(server_t *svr, char** ldev, int *ndev)`
✓ *Devuelve una lista con los dispositivos actualmente conectados*
- `int pm_get_device_info(server_t *svr, char* devn, device_t * dev)`
✓ *Devuelve la frecuencia de muestreo y el número de líneas del dispositivo indicado*

Rutinas básicas biblioteca pmlib (II)

- `int pm.set_lines(char *lines, line_t *line)`
 - ✓ *Selecciona las líneas indicadas*
Cadena separada por "-" o ",". Ejemplo: "1-4,6,9-10".
- `void LINE_SET(line_t *line , int s_line)`
 - ✓ *Activa la línea i-ésima en el array de líneas*
- `void LINE_CLR(line_t *line , int s_line)`
 - ✓ *Desactiva la línea i-ésima en el array de líneas*
- `int LINE_ISSET(line_t *line , int s_line)`
 - ✓ *Consulta si la línea i-ésima en el array de líneas está activo*
- `void LINE_SET_ALL(line_t *line)`
 - ✓ *Activa todas las líneas en el array de líneas*
- `void LINE_CLR_ALL(line_t *line)`
 - ✓ *Desactiva todas las líneas en el array de líneas*
- `int pm.create_counter(char *ldev, line_t line, int aggr,
int freq, server_t svr, counter_t *pm_ctr)`
 - ✓ *Envía una petición al servidor para la creación de un nuevo contador*
- `int pm.start_counter(counter_t *pm_ctr)`
 - ✓ *Empieza la medición del contador*
- `int pm.continue_counter(counter_t *pm_ctr)`
 - ✓ *Continúa la medición conservando datos previos*

Rutinas básicas biblioteca pmLib (III)

- `int pm_stop_counter(counter_t *pm_ctr)`
✓ *Detiene la medición del contador*
- `int pm_get_counter_data(counter_t *pm_ctr)`
✓ *Vuelca los datos en memoria*
- `int pm_print_data_stdout(counter_t *pm_ctr, line_t lines, int set)`
✓ *Imprime los datos por la salida estándar*
- `int pm_print_data_text(char *file, counter_t *pm_ctr, line_t lines, int set)`
✓ *Imprime los datos en un fichero de texto (separados por tabulador)*
- `int pm_print_data_csv(char *file, counter_t *pm_ctr, line_t lines, int set)`
✓ *Imprime los datos en un fichero de texto en formato csv (separados por ;)*
- `int pm_print_data_paraver(char *file, counter_t *pm_ctr, line_t lines, int set, char *unit)`
✓ *Imprime los datos en un fichero en formato compatible con Paraver*
- `int pm_finalize_counter(counter_t *pm_ctr)`
✓ *Finaliza el contador en el servidor*

Comando de consulta pm_info (I)

`pm_info -h|--help` *Muestra ayuda sobre el comando*

```
1 $ pm_info --help
2 Usage: pm_info -s|--server SERVER:PORT
3       pm_info -l|--lines
4       pm_info -c|--counters
5       pm_info -r|--read DEVNAME [-f|--freq FREQ]
6
7 Options:
8   -h, --help            show this help message and exit
9   -s SERVER:PORT, --server=SERVER:PORT
10  -l, --lines
11  -c, --counters
12  -r DEVNAME, --read=DEVNAME
13  -f FREQ, --freq=FREQ
```

`pm_info -s|--server SERVER` *Proporciona información de los dispositivos*

```
1 $ pm_info -s joule.act:6526
2 Dev: DCMeter1 - Freq: 28 Hz - Lines: 12
3   - Computer: wats3
4
5 Dev: DCMeter2 - Freq: 28 Hz - Lines: 12
6   - Computer: wats1
7   ...
8
9 Dev: WattsUp2 - Freq: 1 Hz - Lines: 1
10  - Computer: wats1
```

Comando de consulta pm_info (II)

`pm_info -s|--server SERVER -l|--lines` *Lista los dispositivos con la descripción de sus líneas*

```

1 $ pm_info -s joule.act:6526 -l
2 Dev: PDU - Freq: 1 Hz - Lines: 24
3   * Lines:
4     - Line: 0 - Name: Outlet 1 - Voltage: 220.0 V - Desc: Outlet 1 - Computer: tintorrum.act.uji.es
5     - Line: 1 - Name: Outlet 2 - Voltage: 220.0 V - Desc: Outlet 2 - Computer: compute-1-0
6     - Line: 2 - Name: Outlet 3 - Voltage: 220.0 V - Desc: Outlet 3 - Computer: compute-1-1
7
8     ...
9
10    - Line: 22 - Name: Outlet 23 - Voltage: 220.0 V - Desc: Outlet 23 - Computer: compute-0-3
11    - Line: 23 - Name: Outlet 24 - Voltage: 220.0 V - Desc: Outlet 24 - Computer: NULL

```

`pm_info -s|--server SERVER -c|--counters` *Lista los dispositivos con los contadores actuales*

```

1 $ pm_info -s joule.act:6526 -c
2 Dev: DCMeter1 - Freq: 28 Hz - Lines: 12
3   - Computer: wats3
4   * Counters:
5     - Counter: 22064592 - Address: ('150.128.82.152', 51118) - Selected lines: 0:1:6:7:9:10:11
6     ...
7
8 Dev: WattsUp2 - Freq: 1 Hz - Lines: 1
9   - Computer: wats1

```

Las opciones `-l` y `-c` pueden ser usadas conjuntamente!

Comando de consulta pm_info (III)

`pm_info -s|--server SERVER -r|--read DEVNAME [-f|--freq FREQ]` *Lee los datos del medidor a tiempo real y con la frecuencia indicada*

```

1 $ pm_info -s joule.act:6526 -r DCMeter1
2 13:02:43- 0.00: 29.30: 0.00: 17.81: 34.92: 37.50: 41.72: 29.30: 14.06: 9.84: 37.73: 0.47= 252.66
3 13:02:44- 0.00: 27.89: 0.00: 16.64: 35.16: 38.20: 44.53: 31.17: 14.06: 9.14: 36.09: 0.00= 252.89
4 13:02:44- 0.00: 28.13: 0.00: 17.58: 34.92: 37.50: 40.08: 28.59: 13.83: 10.31: 39.14: 0.00= 250.08
5                                     ...
6
7 $ pm_info -s joule.act:6526 -r WattsUp2 -f 1
8 14:53:46 - 65.90 = 65.90
9 14:53:47 - 66.10 = 66.10
10 14:53:48 - 66.00 = 66.00
11 14:53:49 - 66.10 = 66.10
12 14:53:50 - 65.90 = 65.90
13 14:53:51 - 65.90 = 65.90
14 14:53:52 - 65.90 = 65.90
15 14:53:53 - 66.10 = 66.10
16 14:53:54 - 66.00 = 66.00
17 14:53:55 - 65.90 = 65.90
18                                     ...

```

Ejemplo 1

Uso de funciones de obtención de información de los dispositivos

ej1.c

```
1 #include <stdio.h>
2 #include "pmlib.h"
3
4 int main (int argc, char *argv[])
5 {
6     server_t server;
7     device_t disp;
8     char **list;
9     int i, num_devices;
10
11     printf("Empieza pm_set_server\n");
12     pm_set_server("150.128.83.55", 6526, &server);
13
14     printf("Empieza pm_get_devices\n");
15     pm_get_devices(server, &list, &num_devices);
16
17     printf("Numero de dispositivos: %d\n", num_devices);
18     for(i=0; i<num_devices; i++)
19         printf("Disp %d: %s\n", i, list[i]);
20
21     printf("Empieza pm_get_device_info\n");
22     pm_get_device_info(server, list[0], &disp);
23     printf("Nombre: %s\n", disp.name);
24     printf("Frecuencia maxima: %d\n", disp.max_frequency);
25     printf("Numero de lineas: %d\n", disp.n_lines);
26     return 0;
27 }
```

Ejemplo 1

Resultado

```
1 $ gcc ejemplo1.c pmlib.a -o ejemplo1
2 $ ./ejemplo1
3 Empieza pm_set_server
4 Empieza pm_get_devices
5 Numero de dispositivos: 5
6 Disp0: DCMeter2
7 Disp1: WattsUp1
8 Disp2: WattsUp2
9 Disp3: DCMeter1
10 Disp4: PDU
11 Empieza pm_get_device_info
12 Nombre: DCMeter2
13 Frecuencia maxima: 28
14 Numero de lineas: 12
```

Ejemplo 2

Uso de funciones de obtención de medidas

ej2.c

```
1 #include <stdio.h>
2 #include "pmlib.h"
3
4 int main (int argc, char *argv[])
5 {
6     server_t server;
7     counter_t counter;
8     line_t lines;
9     int i, frequency= 0, aggregate= 1;
10
11     pm_set_server("150.128.83.55", 6526, &server);
12     pm_set_lines("1-3", &lines);
13     pm_create_counter("DCMeter1", lines, !aggregate, frequency, server, &counter);
14     pm_start_counter(&counter);
15     for (i=0; i<3; i++){
16         sleep(1);
17         pm_stop_counter(&counter);
18         sleep(1);
19         pm_continue_counter(&counter);
20     }
21     pm_stop_counter(&counter);
22     pm_get_counter_data(&counter);
23     pm_print_data_text("out.txt", counter, lines, -1);
24     pm_finalize_counter(&counter);
25     return 0;
26 }
```

Ejemplo 2

Resultado

```

1 $ ./ej2.x
2 $ cat out.txt
3 Set_id   Time          Line 1          Line 2          Line 3          Sum
4 1         0.000000      29.766075      0.000000      17.344013      47.110088
5 1         0.035717      28.359804      0.000000      16.640877      45.000679
6 1         0.071434      29.062941      0.000000      17.812769      46.875710
7
8
9
10 2         0.000000      28.594183      0.000000      17.812769      46.406952
11 2         0.037041      27.656668      0.000000      19.453419      47.110085
12 2         0.074082      29.766075      0.000000      17.109634      46.875710
13
14
15
16 3         0.000000      29.062941      0.000000      17.344013      46.406952
17 3         0.037040      30.234833      0.000000      17.109634      47.344467
18 3         0.074079      28.125425      0.000000      18.281527      46.406952
19
20

```


Ejemplo 3

Uso de funciones de impresión de datos

ej3.c

```
1 #include <stdio.h>
2 #include "pmlib.h"
3
4 int main (int argc, char *argv[])
5 {
6     server_t server;
7     counter_t contador;
8     line_t lines;
9     int set= -1, frequency= 0, aggregate= 1;
10
11     pm_set_server("150.128.83.55", 6526, &server);
12     pm_set_lines("5-8", &lines);
13     pm_create_counter("DCMeter1", lines, !aggregate, frequency, server, &contador);
14
15     pm_continue_counter(&contador);
16     sleep(1);
17     pm_stop_counter(&contador);
18
19     pm_get_counter_data(&contador);
20
21     pm_print_data_text("out.txt", contador, lines, set);
22     pm_print_data_csv("out.csv", contador, lines, set);
23     pm_print_data_paraver("out.prv", contador, lines, set, "us");
24     pm_print_data_stdout(contador, lines, set);
25
26     pm_finalize_counter(&contador);
27     return 0;
28 }
```

Ejemplo 3

Resultado en formato .txt

```

1 $ ./ej3.x
2 $ cat out.txt
3 Set_id Time Line 5 Line 6 Line 7 Line 8 Sum
4 1 0.000000 36.563053 43.360031 31.172346 13.125198 124.22067
5 1 0.035720 37.500568 41.250626 29.297318 13.828334 121.87683
6 1 0.071439 38.203705 41.953758 31.875483 13.828334 125.86127
7 1 0.107159 37.969326 41.485004 28.359804 14.531470 122.34560
8 1 0.142878 36.563053 41.953758 31.641104 14.297091 124.45500
9 1 0.178598 38.203705 41.250626 29.062941 13.828334 122.34560
10 1 0.214317 37.734947 41.719383 32.109859 13.125198 124.68939
11 1 0.250037 37.734947 41.485004 29.297318 13.828334 122.34560
12 1 0.285756 37.500568 41.719383 30.469212 14.062713 123.75187
13 1 0.321476 39.375595 41.250626 29.062941 14.531470 124.22063
14 1 0.357195 37.031811 43.125652 29.297318 14.765848 124.22062
15 1 0.392915 39.141216 40.781868 29.062941 13.828334 122.81435
16 1 0.428634 37.734947 42.422516 29.062941 14.765848 123.98625
17 1 0.464354 39.609974 41.250626 30.234833 14.062713 125.15814
18 1 0.500074 37.266190 42.188137 28.828562 14.531470 122.81436
19 1 0.535793 40.078732 43.125652 29.062941 13.828334 126.09565
20 1 0.571513 38.438080 40.781868 29.062941 15.000227 123.28311
21 1 0.607232 41.250626 41.953758 30.234833 14.062713 127.50193
22 1 0.642952 38.203705 41.250626 28.359804 14.062713 121.87684
23 1 0.678671 40.313110 43.828789 28.359804 13.125198 125.62690
24 1 0.714391 39.141216 41.485004 29.766075 13.593956 123.98625
25 1 0.750110 40.078732 41.953758 29.766075 13.828334 125.62689
26 1 0.785830 37.031811 42.656895 29.062941 14.062713 122.81436
27 1 0.821549 40.313110 42.422516 29.766075 13.593956 126.09565
28 1 0.857269 38.203705 41.250626 29.531696 14.062713 123.04873
29 1 0.892988 36.797432 44.766304 29.062941 13.125198 123.75187
30 1 0.928708 38.203705 41.250626 29.297318 13.359577 122.11122
31 1 0.964427 39.141216 44.531925 29.531696 14.297091 127.50192

```

Ejemplo 3

Resultado en formato .csv

```

1 $ cat out.csv
2 Set_id;Time;Line 5;Line 6;Line 7;Line 8;Sum
3 1;0.000000;36.563053;43.360031;31.172346;13.125198;124.220627
4 1;0.035720;37.500568;41.250626;29.297318;13.828334;121.876839
5 1;0.071439;38.203705;41.953758;31.875483;13.828334;125.861275
6 1;0.107159;37.969326;41.485004;28.359804;14.531470;122.345604
7 1;0.142878;36.563053;41.953758;31.641104;14.297091;124.455009
8 1;0.178598;38.203705;41.250626;29.062941;13.828334;122.345604
9 1;0.214317;37.734947;41.719383;32.109859;13.125198;124.689392
10 1;0.250037;37.734947;41.485004;29.297318;13.828334;122.345604
11 1;0.285756;37.500568;41.719383;30.469212;14.062713;123.751877
12 1;0.321476;39.375595;41.250626;29.062941;14.531470;124.220634
13 1;0.357195;37.031811;43.125652;29.297318;14.765848;124.220627
14 1;0.392915;39.141216;40.781868;29.062941;13.828334;122.814354
15 1;0.428634;37.734947;42.422516;29.062941;14.765848;123.986252
16 1;0.464354;39.609974;41.250626;30.234833;14.062713;125.158142
17 1;0.500074;37.266190;42.188137;28.828562;14.531470;122.814362
18 1;0.535793;40.078732;43.125652;29.062941;13.828334;126.095657
19 1;0.571513;38.438080;40.781868;29.062941;15.000227;123.283119
20 1;0.607232;41.250626;41.953758;30.234833;14.062713;127.501930
21 1;0.642952;38.203705;41.250626;28.359804;14.062713;121.876846
22 1;0.678671;40.313110;43.828789;28.359804;13.125198;125.626900
23 1;0.714391;39.141216;41.485004;29.766075;13.593956;123.986252
24 1;0.750110;40.078732;41.953758;29.766075;13.828334;125.626892
25 1;0.785830;37.031811;42.656895;29.062941;14.062713;122.814362
26 1;0.821549;40.313110;42.422516;29.766075;13.593956;126.095657
27 1;0.857269;38.203705;41.250626;29.531696;14.062713;123.048737
28 1;0.892988;36.797432;44.766304;29.062941;13.125198;123.751877
29 1;0.928708;38.203705;41.250626;29.297318;13.359577;122.111221
30 1;0.964427;39.141216;44.531925;29.531696;14.297091;127.501923

```

Ejemplo 3

Resultado en formato .prv (Paraver)

```

1 $ cat out.prv
2 #Paraver (20/06/2012 at 09:52):1000147.104263_us:1(1):1:1:(1:1)
3 2:0:1:1:1:0.000000:500000015:36.563053
4 2:0:1:1:1:0.000000:500000016:43.360031
5 2:0:1:1:1:0.000000:500000017:31.172346
6 2:0:1:1:1:0.000000:500000018:13.125198
7 2:0:1:1:1:0.000000:500000009:124.220627
8 2:0:1:1:1:35719.539438:500000015:37.500568
9 2:0:1:1:1:35719.539438:500000016:41.250626
10 2:0:1:1:1:35719.539438:500000017:29.297318
11 2:0:1:1:1:35719.539438:500000018:13.828334
12 2:0:1:1:1:35719.539438:500000009:246.097473
13 2:0:1:1:1:71439.078876:500000015:38.203705
14 2:0:1:1:1:71439.078876:500000016:41.953758
15 2:0:1:1:1:71439.078876:500000017:31.875483
16 2:0:1:1:1:71439.078876:500000018:13.828334
17 2:0:1:1:1:71439.078876:500000009:371.958771
18 2:0:1:1:1:107158.618314:500000015:37.969326
19 2:0:1:1:1:107158.618314:500000016:41.485004
20 2:0:1:1:1:107158.618314:500000017:28.359804
21 2:0:1:1:1:107158.618314:500000018:14.531470
22 2:0:1:1:1:107158.618314:500000009:494.304382
23 2:0:1:1:1:142878.157752:500000015:36.563053
24 2:0:1:1:1:142878.157752:500000016:41.953758
25 2:0:1:1:1:142878.157752:500000017:31.641104
26 2:0:1:1:1:142878.157752:500000018:14.297091
27 2:0:1:1:1:142878.157752:500000009:618.759399
28 .
29 .
30 .

```

Ejemplo 4

Uso de la opción aggregate

ej4.c

```
1 #include <stdio.h>
2 #include "pmlib.h"
3
4 int main (int argc, char *argv[]){
5     server_t servidor;
6     counter_t counter, counter2;
7     line_t lines, lines2;
8     int set= -1, frequency= 0, aggregate= 1;
9     pm_set_server("150.128.83.55", 6526, &servidor);
10    pm_set_lines("1,3-5", &lines);
11
12    pm_create_counter("DCMeter2", lines, !aggregate, frequency, servidor, &counter);
13    pm_create_counter("DCMeter2", lines, aggregate, frequency, servidor, &counter2);
14
15    pm_start_counter(&counter);
16    pm_start_counter(&counter2);
17    sleep(1);
18    pm_stop_counter(&counter);
19    pm_stop_counter(&counter2);
20    pm_get_counter_data(&counter);
21    pm_get_counter_data(&counter2);
22
23    pm_set_lines("3,4", &lines2);
24    pm_print_data_text("out_non_aggregate.txt", counter, lines2, set);
25    pm_print_data_text("out_aggregate.txt", counter2, lines, set);
26    pm_finalize_counter(&counter);
27    pm_finalize_counter(&counter2);
28    return 0;
29 }
```

Ejemplo 4

Resultado: muestra las medidas de las líneas 3 y 4 y la suma de los valores de ambas líneas

```
1 $ cat out_non_aggregate.txt
2 Set_id Time Line 3 Line 4 Sum
3 1 0.000000 0.234379 13.125198 13.359577
4 1 0.037081 0.703136 8.203249 8.906384
5 1 0.074162 0.000000 17.812769 17.812769
6 1 0.111242 0.703136 8.672007 9.375142
7 1 0.148323 0.703136 17.344013 18.047150
8 1 0.185404 0.234379 8.672007 8.906385
9 1 0.222485 0.703136 8.906384 9.609520
10 1 0.259565 0.703136 8.906384 9.609520
11 1 0.296646 0.234379 11.250171 11.484550
12 1 0.333727 0.468757 8.672007 9.140763
13 1 0.370808 0.234379 11.250171 11.484550
14 1 0.407889 0.468757 8.437628 8.906384
15 1 0.444969 0.000000 8.203249 8.203249
16 1 0.482050 0.234379 17.578390 17.812769
17 1 0.519131 0.000000 9.140763 9.140763
18 1 0.556212 0.468757 13.125198 13.593955
19 1 0.593293 0.000000 8.672007 8.672007
20 1 0.630373 0.234379 9.140763 9.375142
21 1 0.667454 0.703136 9.140763 9.843899
22 1 0.704535 0.468757 8.203249 8.672006
23 1 0.741616 0.468757 8.437628 8.906384
24 1 0.778696 0.000000 8.672007 8.672007
25 1 0.815777 0.000000 9.140763 9.140763
26 1 0.852858 0.468757 8.437628 8.906384
27 1 0.889939 0.468757 9.140763 9.609520
28 1 0.927019 0.234379 8.672007 8.906385
29 1 0.964100 0.703136 8.203249 8.906384
30 1 1.001181 0.000000 8.203249 8.203249
```

Ejemplo 4

Resultado: muestra sólo el valor agregado de todas las líneas medidas (1, 3, 4 y 5)

```
1 $ cat out_aggregate.txt
2 Set_id   Timea      Value_aggregate
3 1         0.000000    25.078505
4 1         0.037060    14.531470
5 1         0.074120    24.375368
6 1         0.111180    15.468984
7 1         0.148241    25.312883
8 1         0.185301    14.765848
9 1         0.222361    17.812769
10 1        0.259421    16.640877
11 1        0.296481    21.797205
12 1        0.333541    15.000227
13 1        0.370602    20.390934
14 1        0.407662    13.593956
15 1        0.444722    14.062713
16 1        0.481782    22.500341
17 1        0.518842    20.156555
18 1        0.555902    18.281527
19 1        0.592962    18.750284
20 1        0.630023    12.422063
21 1        0.667083    15.703362
22 1        0.704143    13.593956
23 1        0.741203    16.172119
24 1        0.778263    13.828334
25 1        0.815323    15.703362
26 1        0.852383    13.359577
27 1        0.889444    14.765848
28 1        0.926504    14.297091
29 1        0.963564    14.297091
30 1        1.000624    18.515905
```

Gracias

¿Dudas? ¿Preguntas?