



TINE Release 4.x.x News

(Feb 17, 2016: That was the month that was !)

“What a long, strange trip it’s been”

[Release 4.5.8]

■ C-Library: Noteworthy Bug-fixes

- A couple of memory overwrites ...
 - Long-standing (i.e. several years!)
 - harmless (?)
- A couple of (exotic) deadlocks ...
 - Startup: using FindNameServerOnNetwork() but not having a network.
 - Cleanup: closing a TCP recv thread socket when data are still coming in.

■ Java: Noteworthy Bug-fixes

- ENS concurrency with address resolution and info queries ...
 - 'Just the right conditions'
 - primary and secondary ENS
 - no cshosts.csv
 - address query to non-existent server

■ Java: Noteworthy Workarounds

- LogRecord Class and java 8 and the fec.log.0 problem.
 - Scan for defunct .lck files at server startup.

[Release 4.5.8]

■ Embellishments: C-Lib

- Make use of known systematics when *GetSystemPropertyInformation()* is called with a stock or meta- property!
 - (also fixed a bug when memory 'allocation' is requested)
 - e.g. *GetSystemPropertyInformation(..."DEVICES")*
 - The 'property information' for stock property 'DEVICES' is known locally (ask for 1000 NAME64s !)
 - Why make this call in the first place? => happens automatically in e.g. MatLab if the caller says *XCOMM(address, "DEVICES")* or e.g. Python *PyTine.get(address, "DEVICS")* instead of using the 'query' calls (e.g. *PyTine.list(address)*).

[Release 4.5.8]

■ Embellishments: C-Lib

- A 'trace link' has now been introduced (a la java) e.g.:

```
set trace = /PETRA/BPM/#0[Orbit.X]
```

- Outputs all link activity for the trace link at the console (foreground or via [attachfec](#)).
- Very useful if a client has 'many' link table entries and you are interested in following one of them.

Release 4.5.8

Trace link example:

```
2:mcskxterm01 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

get connections
> Current Connection Table
>[0] //ENSEQM/#0[SIZES] 2 value(s)
    @ 0 msec (REGISTER ----) (cbId: 0) - UP
>[1] /TEST/CAS/ARCHIVER[REMOVEALARMS] 0 value(s)
    @ 200 msec ( CANCEL -UDP-) (cbId: 1) - -
>[2] /TEST/SineServer/SineGen0[Frequency] 10 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 1) - UP
>[3] /TEST/SineServer/SineGen0[Sine] 8192 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 2) - UP
>[4] /TEST/Mag.Corr/H2[Strom.Ist] 1 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 3) - DOWN
>[5] /TEST/Mag.Corr/H2[Strom.Soll] 1 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 4) - DOWN
>[6] /TEST/SineServer/SineGen1[Noise] 10 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 6) - UP
>[7] /DESY2/BunchStrom_IMA/IMA-DE05[BunchStrom] 1 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 9) - UP
>[8] /DESY2/BunchStrom_IMA/Board1[HorOffset] 1 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 10) - UP
>[9] /HASYLAB/SpsInfo/P00[BufferPressureBandAlarm] 12 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 11) - UP
>[10] /TEST/ARCHIVER/AMGEN[LOADSIN] 3 value(s)
    @ 1000 msec ( TIMER -UDP-) (cbId: 12) - DOWN

Connected to mcskxterm01  SSH2 - aes128-cbc - hmac-md5 - nc 80x24
```

```
2:mcskxterm01 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

>
>
set trace=/PETRA/BPM/#0[Orbit.X]
>current link trace: /PETRA/BPM/#0[Orbit.X]
>
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
set >[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
trac>[processData (lid: 53)] /PETRA/BPM/#0[Orbit.X] mode TIMER
>[fireCallback (lid: 53)] /PETRA/BPM/#0[Orbit.X] needs to notify!
e=

Connected to mcskxterm01  SSH2 - aes128-cbc - hmac-md5 - nc 80x24
```

[Release 4.5.8]

■ Embellishments: C-Lib

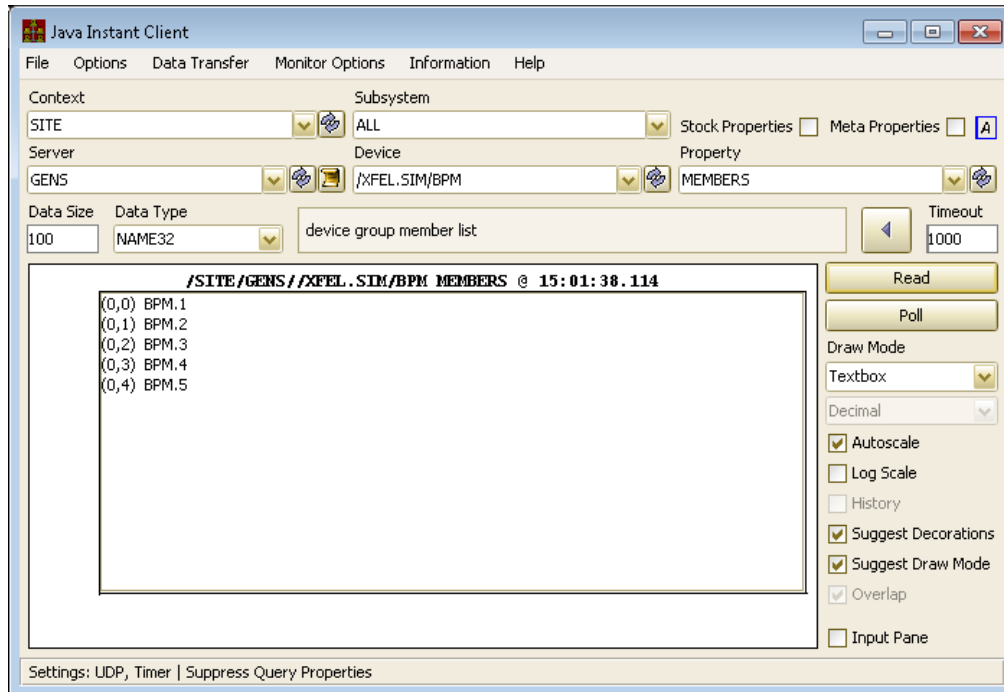
- Multi-threaded equipment function calls *refactored*:
 - **Was**: resident threads responsible for registered property or stock property access.
 - **Now**: detached threads handle isolated calls.
 - Less likely to get *operation_busy* !
 - Multiple simultaneous calls to `<P>.HIST` possible.
 - **But**: Tread safety within the equipment module handler is the developer's responsibility!
 - **Note**: A registered property must be declared to run in its own thread (not the default).
 - **Note**: Some stock and meta properties do this automatically.

[Release 4.5.8]

- **Embellishments: C-Lib and Java**
 - *wildcard* access involving more than one host.
 - e.g. “/PETRA/BPM/*[Orbit.X]”
 - if all *devices on same host* then handled as a **single call** (if a multi-channel array, it’s even more efficient).
 - if *some devices are redirected* to other servers then ...
 - split into N calls and piece them together ...

Release 4.5.8

- **Embellishments: C-Lib and Java**
 - *wildcard* access involving more than one host.
 - e.g. `'/XFEL.SIM/BPM/*[X.SA1]'`



Server 'BPM' in context 'XFEL.SIM' is a 'group' server distributed over 5 different 'hosts' ...

Release 4.5.8

- **Wildcard access** to server *groups*.

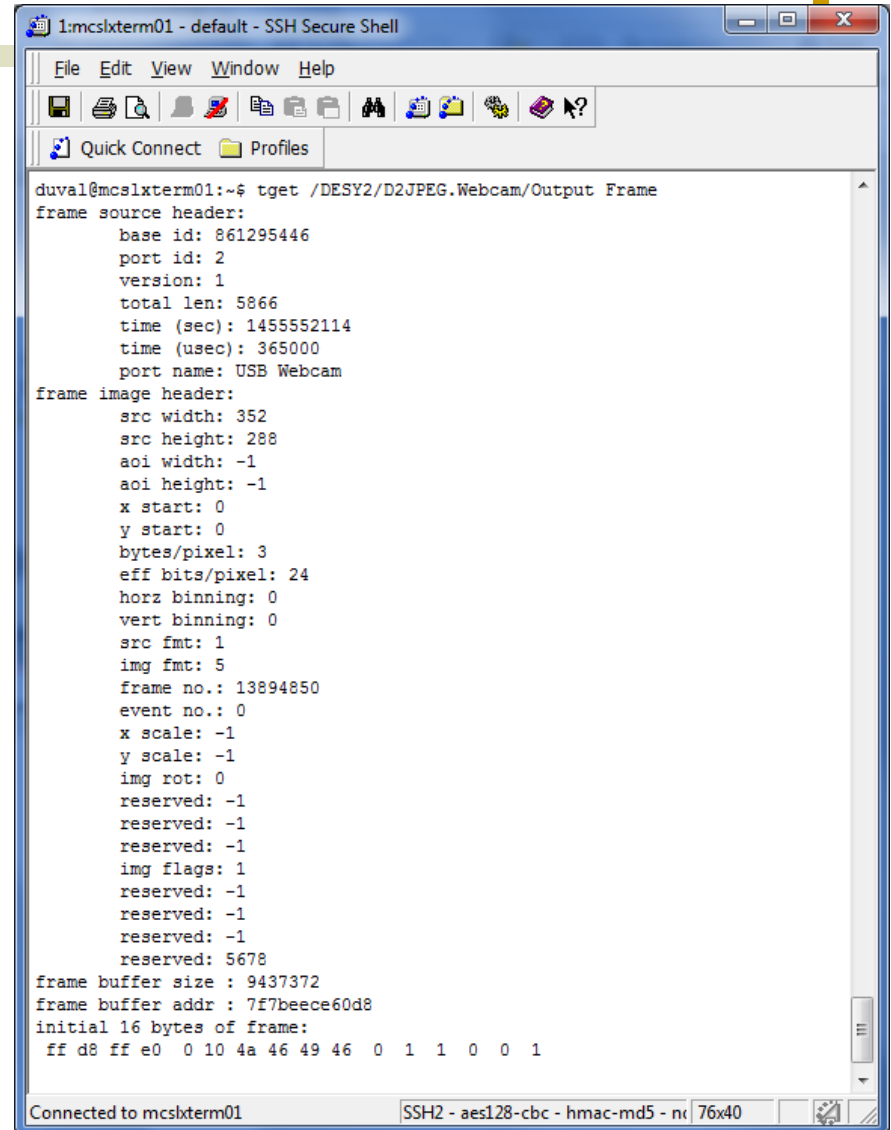
The screenshot displays two windows from a Java-based application. The top window, titled "Application Debugging console (Windows 7 6.1)", shows a command prompt with the text "clients contracts connections messages debug level: off 1 2 3 4 command: get wildcards|". Below the command prompt, it lists the "Current wildcard link table" for the path "/XFEL.SIM/BPM/*[X.SA1]" (output: 500 USTRING), showing active members BPM.1 through BPM.5.

The bottom window, titled "Java Instant Client", shows a monitoring interface. The "Context" is set to "/XFEL.SIM" and the "Subsystem" is "ALL". The "Server" is "BPM" and the "Device" is "*". The "Property" is "X.SA1". The "Data Size" is 500 and the "Data Type" is "USTRING". The "data reading with filter" is active. The "Timeout" is 1000. The "Read" button is highlighted. The "Draw Mode" is "SimpleHistogram". The "Autoscale" checkbox is checked. The "Log Scale" checkbox is unchecked. The "History" checkbox is unchecked. The "Suggest Decorations" checkbox is checked. The "Suggest Draw Mode" checkbox is checked. The "Overlap" checkbox is checked. The "Input Pane" checkbox is unchecked. The "Settings" are "UDP, Timer | Suppress Query Properties". The "Δt=1.000 s, 0 cycles" is displayed at the bottom right.

The main display area of the Java Instant Client shows a histogram plot titled "/XFEL.SIM/BPM/* X.SA1 @ 14:55:03.960". The y-axis ranges from -0.02 to 0.2, and the x-axis ranges from 0 to 350. The plot shows a noisy signal with a peak around 140.

Release 4.5.8

- Command line embellishments
 - tget & tmonitor now handle almost all datatypes and sizes :

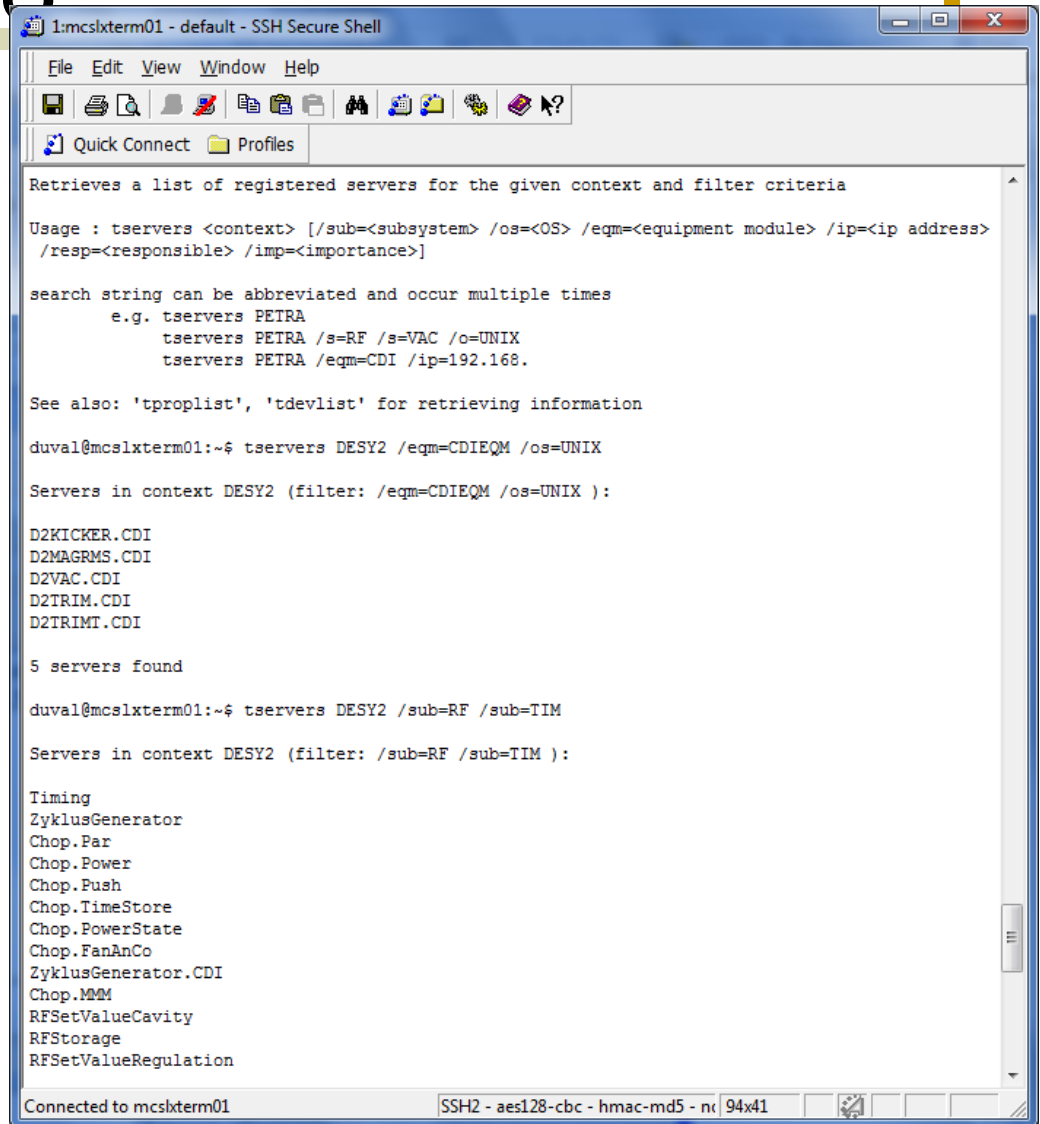


```
1:mcslxterm01 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
duval@mcslxterm01:~$ tget /DESY2/D2JPEG.Weboam/Output Frame
frame source header:
  base id: 861295446
  port id: 2
  version: 1
  total len: 5866
  time (sec): 1455552114
  time (usec): 365000
  port name: USB Webcam
frame image header:
  src width: 352
  src height: 288
  aoi width: -1
  aoi height: -1
  x start: 0
  y start: 0
  bytes/pixel: 3
  eff bits/pixel: 24
  horz binning: 0
  vert binning: 0
  src fmt: 1
  img fmt: 5
  frame no.: 13894850
  event no.: 0
  x scale: -1
  y scale: -1
  img rot: 0
  reserved: -1
  reserved: -1
  reserved: -1
  img flags: 1
  reserved: -1
  reserved: -1
  reserved: -1
  reserved: 5678
frame buffer size : 9437372
frame buffer addr : 7f7beece60d8
initial 16 bytes of frame:
  ff d8 ff e0 0 10 4a 46 49 46 0 1 1 0 0 1
Connected to mcslxterm01  SSH2 - aes128-cbc - hmac-md5 - nc 76x40
```

[Release 4.5.8]

■ Command line embellishments

- tservers with more search possibilities :



```
1:mcslxterm01 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

Retrieves a list of registered servers for the given context and filter criteria

Usage : tservers <context> [/sub=<subsystem> /os=<OS> /eqm=<equipment module> /ip=<ip address>
      /resp=<responsible> /imp=<importance>]

search string can be abbreviated and occur multiple times
  e.g. tservers PETRA
       tservers PETRA /s=RF /s=VAC /o=UNIX
       tservers PETRA /eqm=CDI /ip=192.168.

See also: 'tproplist', 'tdevlist' for retrieving information

duval@mcslxterm01:~$ tservers DESY2 /eqm=CDIEQM /os=UNIX

Servers in context DESY2 (filter: /eqm=CDIEQM /os=UNIX ):

D2KICKER.CDI
D2MAGRMS.CDI
D2VAC.CDI
D2TRIM.CDI
D2TRIMI.CDI

5 servers found

duval@mcslxterm01:~$ tservers DESY2 /sub=RF /sub=TIM

Servers in context DESY2 (filter: /sub=RF /sub=TIM ):

Timing
ZyklusGenerator
Chop.Par
Chop.Power
Chop.Push
Chop.TimeStore
Chop.PowerState
Chop.FanAnCo
ZyklusGenerator.CDI
Chop.MMM
RFSetValueCavity
RFStorage
RFSetValueRegulation

Connected to mcslxterm01  SSH2 - aes128-cbc - hmac-md5 - nc 94x41
```

[Release 4.5.8]

- Python News

- **IMAGE** datatype now included
 - Image analysis/display library to come
- setalarm/clearalarm include in python server ...
- SYNCGROUP example ...

Release 4.5.8

Python syncgroup example ...

```
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:44:40) [MSC v.1600 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> import PyTine as pt
>>>
>>> def gcb(id,cc,d):
...     if ((d['systemstamp'] % 10) == 0) : print(d['timestring'],d['systemstamp'])
...
>>> lid1=pt.attach(address='/PETRA/Idc/Buffer-0',property='I',size=1,mode='timer|syncgroup',callback=gcb)
>>> lid2=pt.attach(address='/PETRA/BLM/PU01I',property='LossRates',mode='timer|syncgroup',callback=gcb)
>>> lid3=pt.attach(address='/PETRA/FREQ-VXW/PETRA0',property='FREQ-RFist',mode='timer|syncgroup',callback=gcb)
>>>
>>> 16.02.16 17:08:14.949 CET 574044140
>>>pt.debug('get groups')
Current Group Table
> Group 0 Members :
> /PETRA/FREQ-VXW/PETRA0[FREQ-RFist] + 0 cnts
> /PETRA/BLM/PU01I[LossRates] + 0 cnts (*head*)
> /PETRA/Idc/Buffer-0[I] + 0 cnts
> number in group : 3
> number pending : 1
> current group cycle stamp : 574044178
> last group cycle stamp : 574044178
> current group cycle dispersion : 0 counts
> current group time dispersion : 54 msec
> current group synchronization : is synchronized
> effective group update interval : 50 msec
> group updating monotonically : FALSE
> most recent update : 16.02.16 17:08:18.743 CET
> current group status code : 0
>
>
>>>> 16.02.16 17:08:22.049 CET 574044190
```

Simple callback function

3 links with a syncgroup callback

debug output

Release 4.5.8

Python syncgroup example : the 'data' dictionary :

```
{'timestamp': 1455641684.963787, 'synchronized': 1, 'systemstamp':  
574061580, 'timestring': '16.02.16 17:54:44.963 CET', 'members': [  
{'timestamp': 1455641685.007451, 'id': 0, 'sysstamp': 574061580,  
'data': 499.66431990015286, 'status': 0, 'usrstamp': 0, 'key':  
'/PETRA/FREQ-VXW/PETRA0[FREQ-RFist]}},  
{'timestamp': 1455641685.035997, 'id': 0, 'sysstamp': 574061580,  
'data': [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
0, 0, 0, 0, 0], 'status': 0, 'usrstamp': 0, 'key':  
'/PETRA/BLM/PU01I[LossRates]}},  
{'timestamp': 1455641684.963787, 'id': 0, 'sysstamp': 574061579,  
'data': 0.0027472786605358124, 'status': 0, 'usrstamp': 0, 'key':  
'/PETRA/Idc/Buffer-0[I]'}}}
```

Release 4.5.8

■ acop.NET now has a transport layer !

The screenshot displays the Visual Studio IDE with a Windows Form named 'Form1' in Design view. The form contains a line graph with a y-axis ranging from -100 to 100 and an x-axis ranging from 0 to 100. The graph shows a smooth, periodic wave-like curve. The Properties window on the right shows the 'Acop1 AcopChart.Acop' control, with the 'Acop.Transport' section expanded. The 'LinkDevice' property is set to 'BPM_SWR_13'. The Output window at the bottom shows the following text:

```
Show output from: Debug
The thread 'vshost.LoadReference' (0xa90) has exited with code 0 (0x0).
'WindowsApplication3.vshost.exe' (Managed (v4.0.30319)): Loaded 'Z:\Projects\Service\VS2012\Acop.NET\Winc
'WindowsApplication3.vshost.exe' (Managed (v4.0.30319)): Loaded 'C:\Windows\Microsoft.Net\assembly\GAC_MS
'WindowsApplication3.vshost.exe' (Managed (v4.0.30319)): Loaded 'Z:\Projects\Service\VS2012\Acop.NET\Winc
'WindowsApplication3.vshost.exe' (Managed (v4.0.30319)): Loaded 'Z:\Projects\Service\VS2012\Acop.NET\Winc
```

[Release 4.5.8: MatLab News]

- clear ...
 - clear mex, clear all, clear XCOMM, etc.
 - unloads all relevant libraries
 - These need to free their resources 'gracefully' => the process itself (MatLab) is still resident and might load the libraries again.
 - Windows:
 - Several cleanup issues recently fixed.
 - FLASH/XFEL:
 - doocs_read vs. either xcomm or tine_read.

Release 4.5.8: MatLab News

```
Command Window
>> a = tine_read('/TEST/SineServer/SineGen0[Amplitude]@1000|CONNECT')

a =

    error: ''
    timestamp: '16.02.16 10:12:47.631 CET'
           utc: '1455613967.632'
    Amplitude: [10x1 double]

>> clear mex
>> a = tine_read('/TEST/SineServer/SineGen0[Amplitude]@1000|CONNECT')

a =

    error: ''
    timestamp: '16.02.16 10:13:01.335 CET'
           utc: '1455613981.335'
    Amplitude: [10x1 double]

>> clear mex
>> a = tine_read('/TEST/SineServer/SineGen0[Amplitude]@1000|CONNECT')

a =

    error: ''
    timestamp: '16.02.16 10:13:11.805 CET'
           utc: '1455613991.805'
    Amplitude: [10x1 double]

>> clear mex
fx >>
```

[Release 4.5.8: Java News]

■ CAM News:

- Control Application Manager can now span *more than one host*.
- CAM applications look for a *local console daemon* at startup time
 - And this won't exist for most of the hosts (e.g. your desktop) that run controls applications
- **Concurrency issue** (query timeout) if trying to **resolve the above address AND querying the ENS** for information (e.g. 'CONTEXTS') at the same time.

[Release 4.5.8: Java News]

- **New behavior: link exists runtime exception**

- When a **WRITE** access link is instantiated **AND** an **active WRITE link** to the same address already exists.

- /PETRA/Magnets/PSC1[Current] = 3

- /PETRA/Magnets/PSC1[Current] = 4

- What should happen?

- Worse: if same input then link2 binds to link1

- **Paradigm for some:** create some link objects at startup and use them later; change the input data with a TLink.putData() method when needed.

- Use a 'tag' if think you know what you're doing ...

- Better would be an exception at point of usage (attach() or execute()) ...

- More complicated (is this worth it?)

Simultaneous
threads/asynchronous
calls

[Release 4.5.8: Java News]

■ New Feature:

- multi-threaded equipment module (property handler) calls for registered and stock properties (a la C-Lib).
- registered properties: flag a property to run in its own thread (thread safety is your business)
 - `TExportProperty.setRunInSeparateThread(true);`
- some stock and meta- properties now do this automatically:
 - e.g. `<P>.HIST`

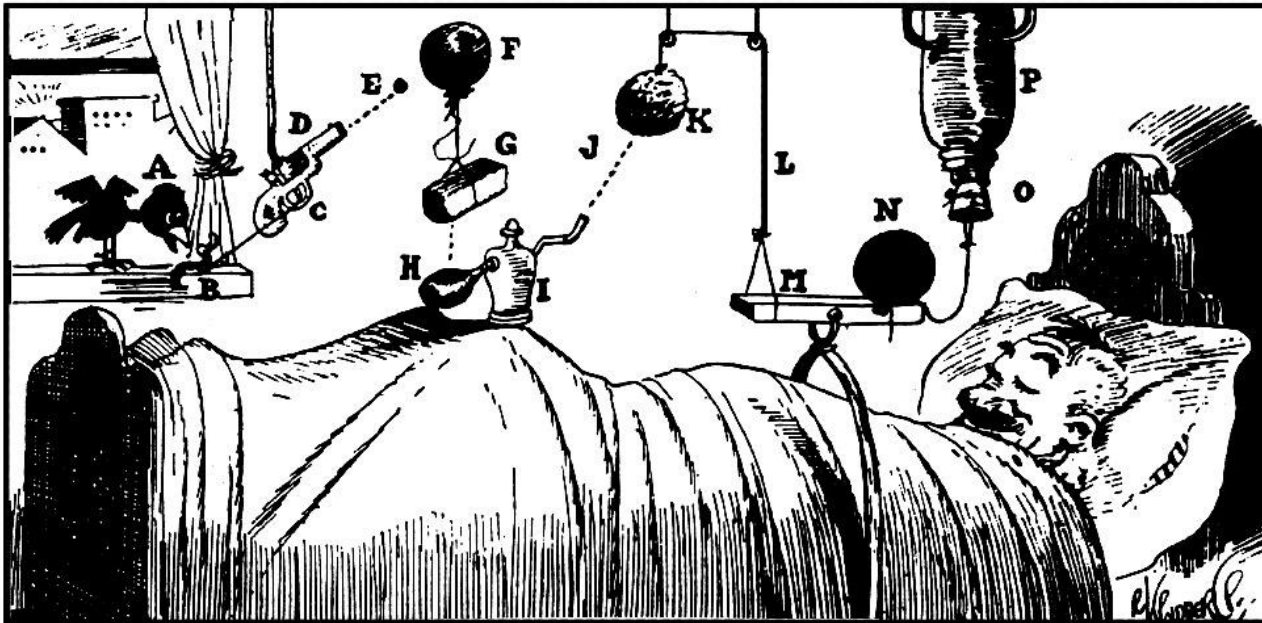
[Release 4.5.8:]

- Control Systems (everywhere) tend to evolve into *Rube Goldberg machines* ...
 - Complicated fitting together of disparate intermediate solutions to achieve an *end**.
 - are there easier ways?
 - is that middle layer really necessary?
 - is the data flow paradigm optimal?
 - The fun happens when the *end** suddenly 'stops working'.

*the control application the operators are trying to use.

Rube Goldberg Machines

Simple Alarm Clock



The early bird (A) arrives and catches worm (B), pulling string (C) and shooting off pistol (D). Bullet (E) busts balloon (F), dropping brick (G) on bulb (H) of atomizer (I) and shooting perfume (J) on sponge (K)—As sponge gains in weight, it lowers itself and pulls string (L), raising end of board

(M)—Cannon ball (N) drops on nose of sleeping gentleman—String tied to cannon ball releases cork (O) of vacuum bottle (P) and ice water falls on sleeper's face to assist the cannon ball in its good work.