

# Indexes for idle VM detection

Giovanni Franzini

# Proposed indexes

## 1) CPU Idle (EMA)

Based on `/proc/uptime` file.

## 2) Keyboard / pty idle time (Condor source code)

Based on `utmp` file.

## 3) Bytes tx and rx by the network interface (bytes per period and EMA)

## 4) lowait ticks (EMA)

Percentage of time the CPU is idle AND there is at least one I/O in progress (local disk or remotely mounted disk, NFS).

## 5) Context switches (EMA)

## 6) Paged in/out (EMA)

# Why use EMA?

- EMA = Exponential Moving Average.
- A moving weighted average, where weighting factors decrease exponentially.
  - ◆ We want to keep track of old values.
  - ◆ New records are more important than old ones.

$$\text{avg}(1) = \text{value}(1)$$

$$\text{avg}(n) = (1 - \text{alpha}) * \text{value}(n) + \text{alpha} * \text{value}(n-1)$$

$$\text{alpha} = 0.3$$

(generally  $\text{alpha} < \frac{1}{2}$ )

# CPU Idle

- Into `/proc/uptime` there are two numbers:
  - ◆ total number of seconds the system has been up;
  - ◆ how much of that time the machine has spent idle, in seconds.

- So, at time  $n$ , we compute

$$\text{delta\_idle}(n) = \text{idle}(n) - \text{idle}(n-1)$$

$$\text{delta\_total}(n) = \text{total}(n) - \text{total}(n-1)$$

$$\text{idle\_perc}(n) = \text{delta\_idle}(n) / \text{delta\_total}(n)$$

- Then we compute the EMA of `idle_perc`, at each step.

# lowait, context switches and paged in/out

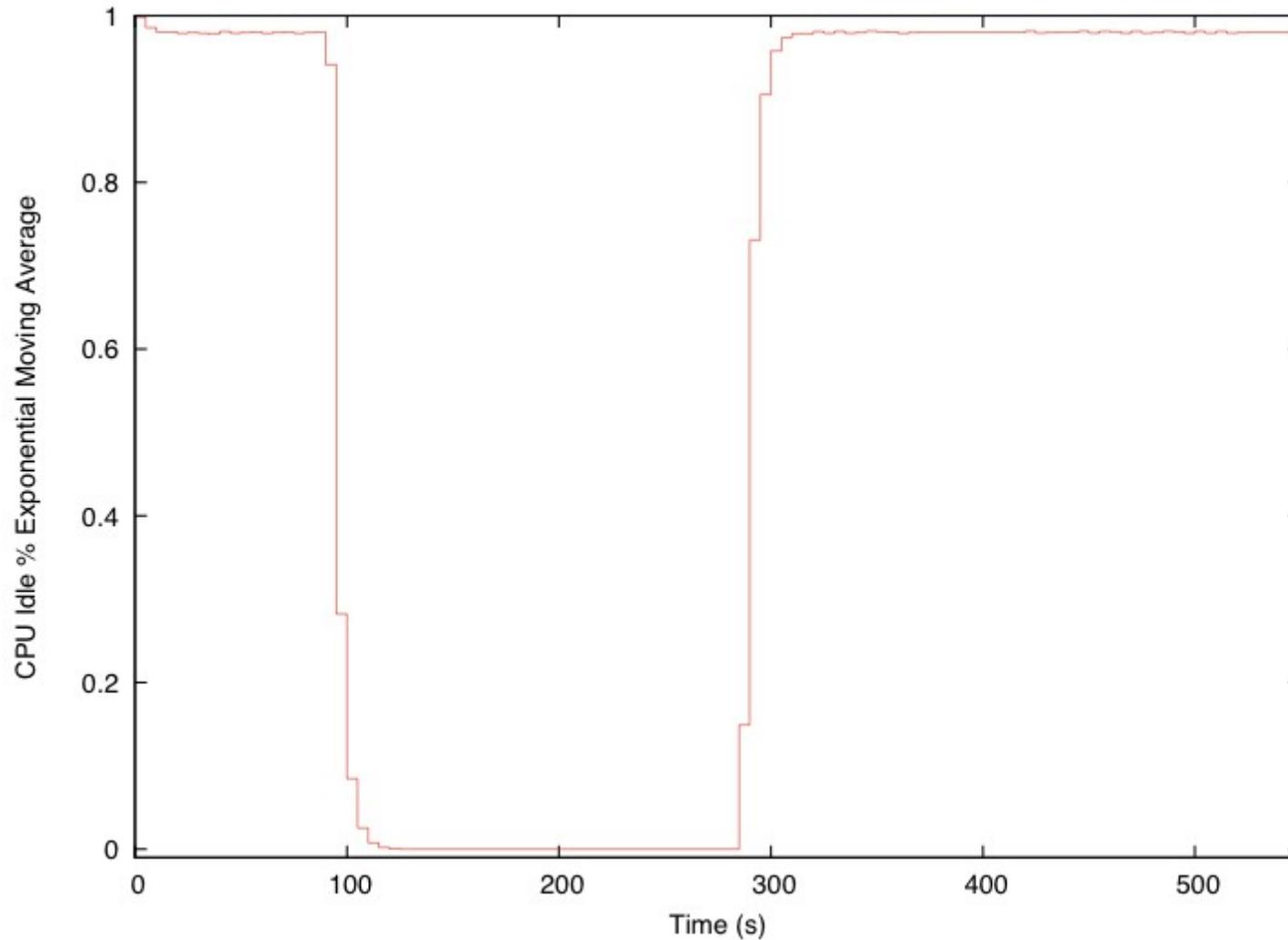
- In a similar way we can compute iowait percentage (and the relative EMA) over a sample period.
- We only need to know how many ticks there are in a second
  - ◆ `sec_to_ticks.cpp`
- Information about how many ticks the CPU has spent in iowait status, since it was boot up, can be found using `vmstat -s`.
- This command gives also information about
  - ◆ Number of context switches.
  - ◆ Number of paged in/out.
- So we only need to compute the number of context switches and paged in/out in a sample period, and update relative EMAs.

# Network activity

- In order to keep track of the network activity, we can compute the number of bytes tx or rx by the network interface (eth0) in a sample period.
- These information can be found in `/proc/net/dev`.
- Then we can compute the EMA of these two values.
- The problem is to distinguish between “idle activity”, that it is like noise, and “effective activity” (someone is using the network to tx or rx useful data).

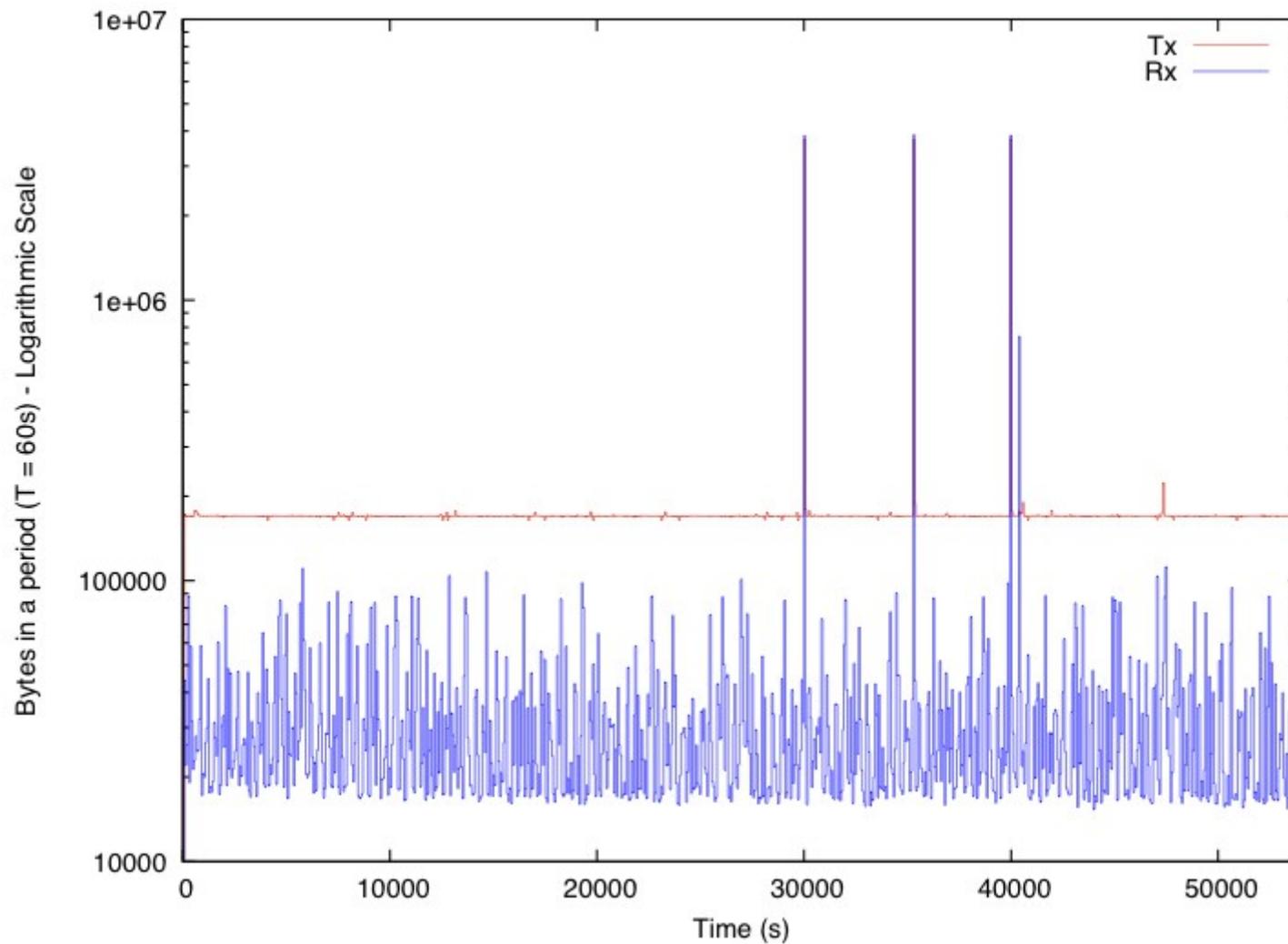
# Index in action: CPU Idle

## Prime number stress test (T = 5s)



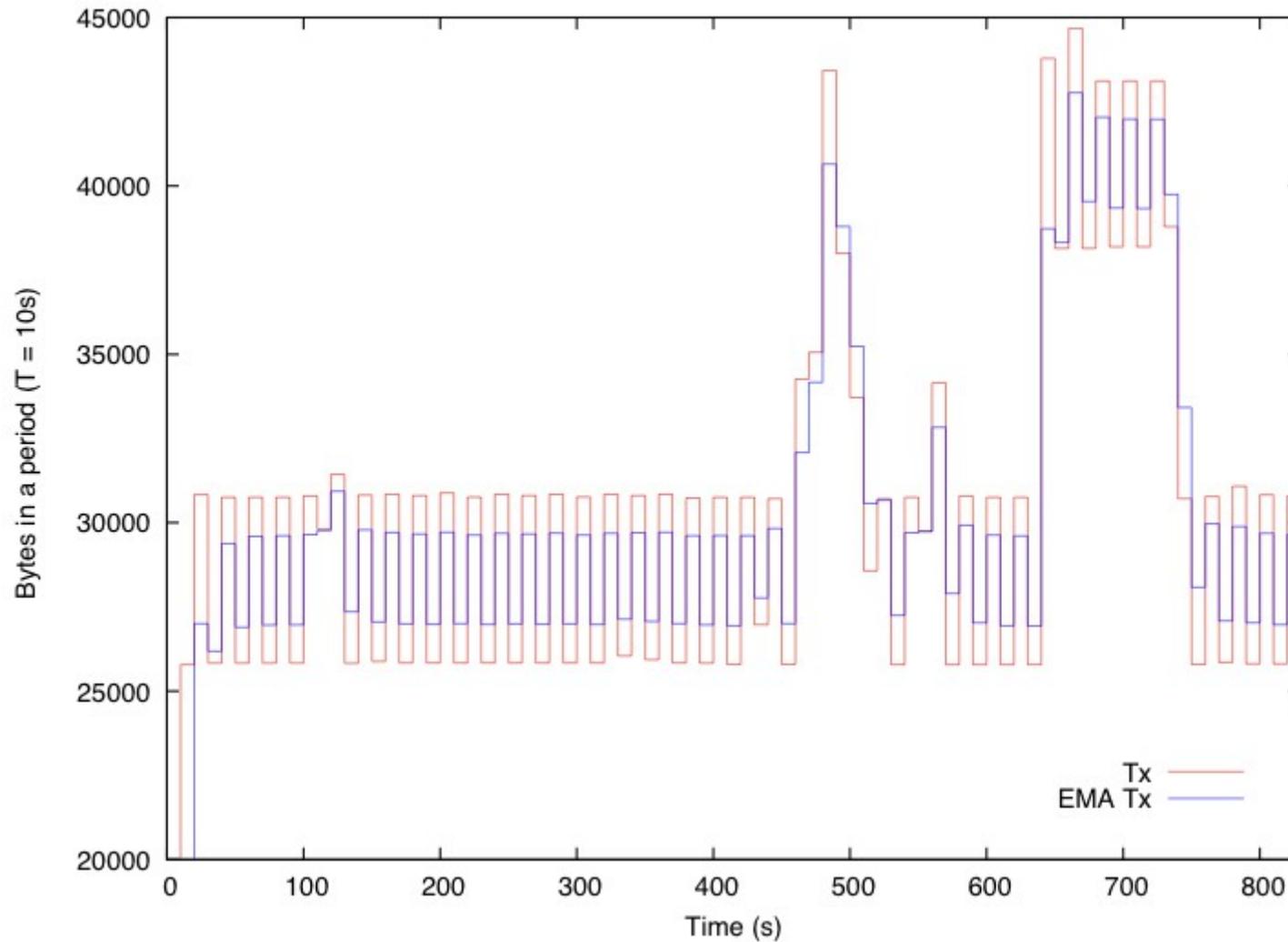
# Index in action: Network activity

## Idle activity (T = 60s)



# Index in action: Network activity

## Ping test ( $T = 10s$ )



# Next steps

- Use the indexes into a “real context” in order to understand their usefulness.
  - ◆ 24 hours tests on idle and used VMs.
  - ◆ VM users should fill a simple form, with use times of the VM (start use time, end use time).
  - ◆ Useful also to identify a correct set of values for the indexes' parameters (alpha, sample periods, etc.).
- Find an easy way to distribute and install the monitoring scripts on the VMs
  - ◆ rpm?
  - ◆ .tar?
  - ◆ ???