

The New Metrics System





Advantages of the new Metrics System

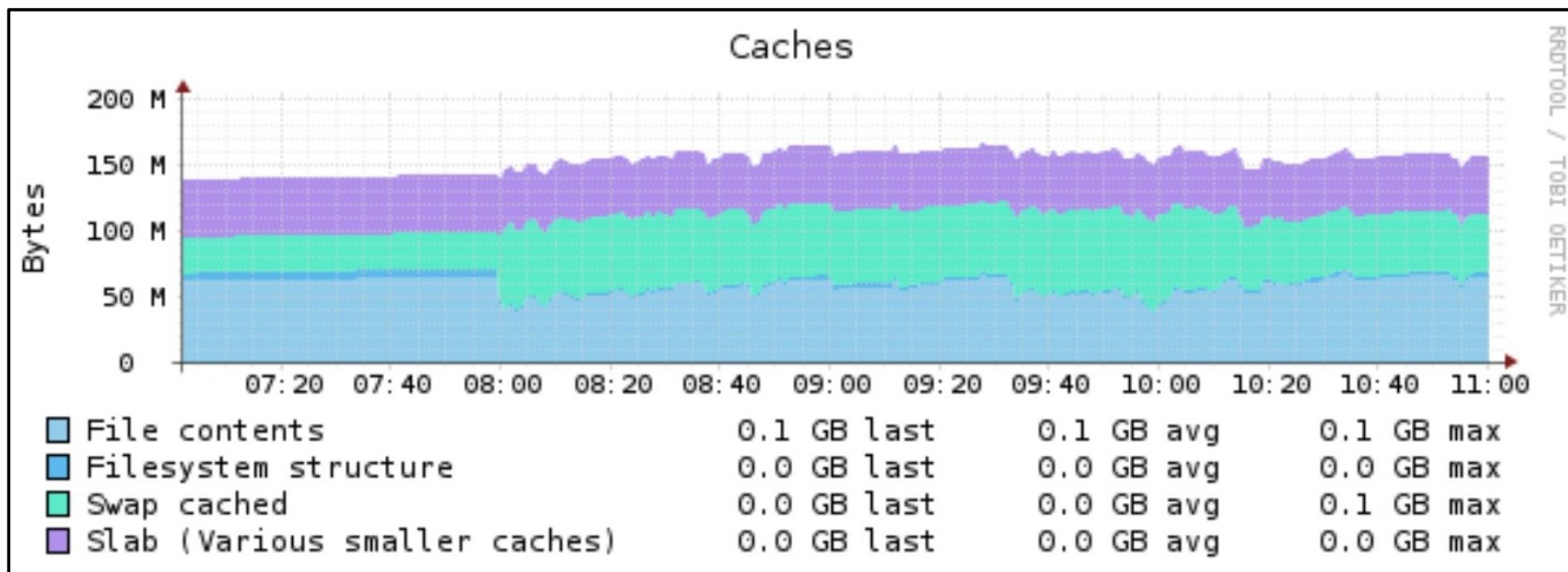
- Remote data via Livestatus, not HTTP
- Easier to create graph templates
- Also covers Perf-O-Meters
- Allows interesting future extensions
- and last but not least....
-nicer to look at



Graph rendering



PNP4Nagios

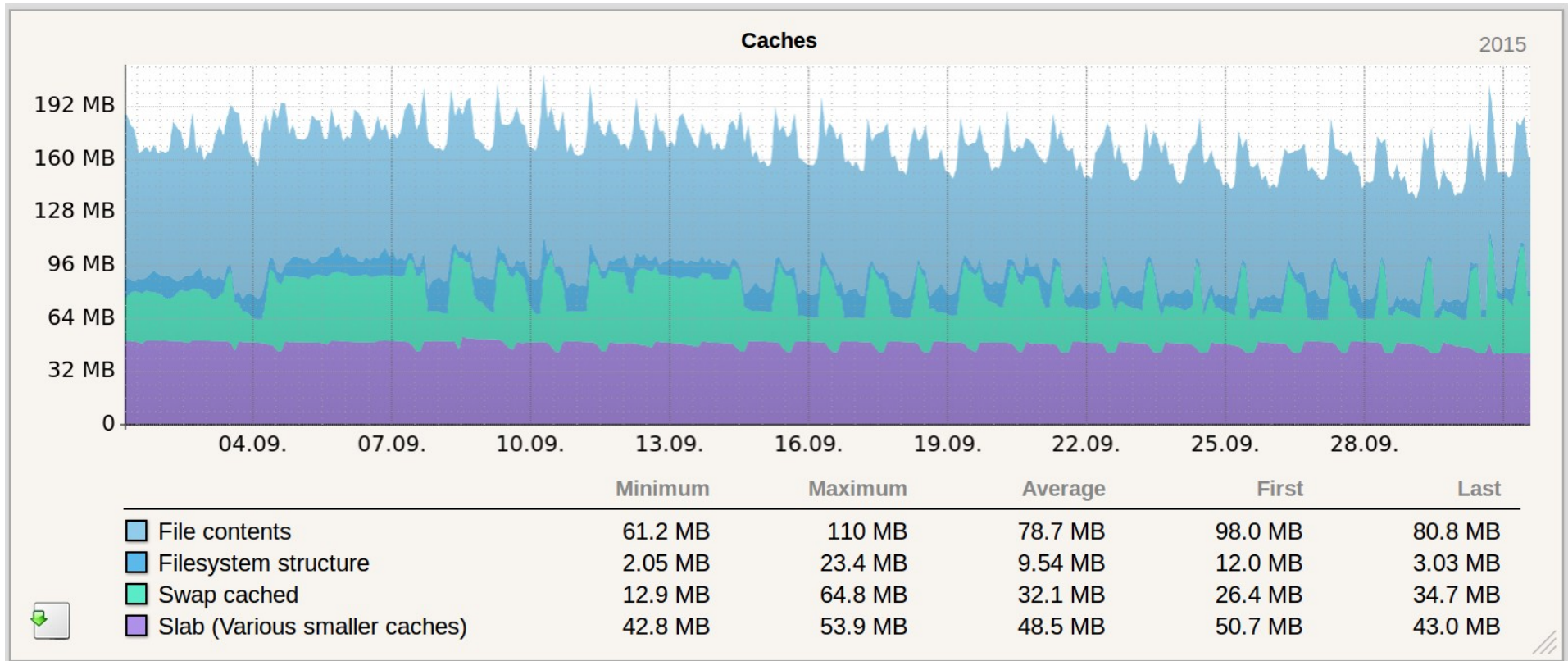


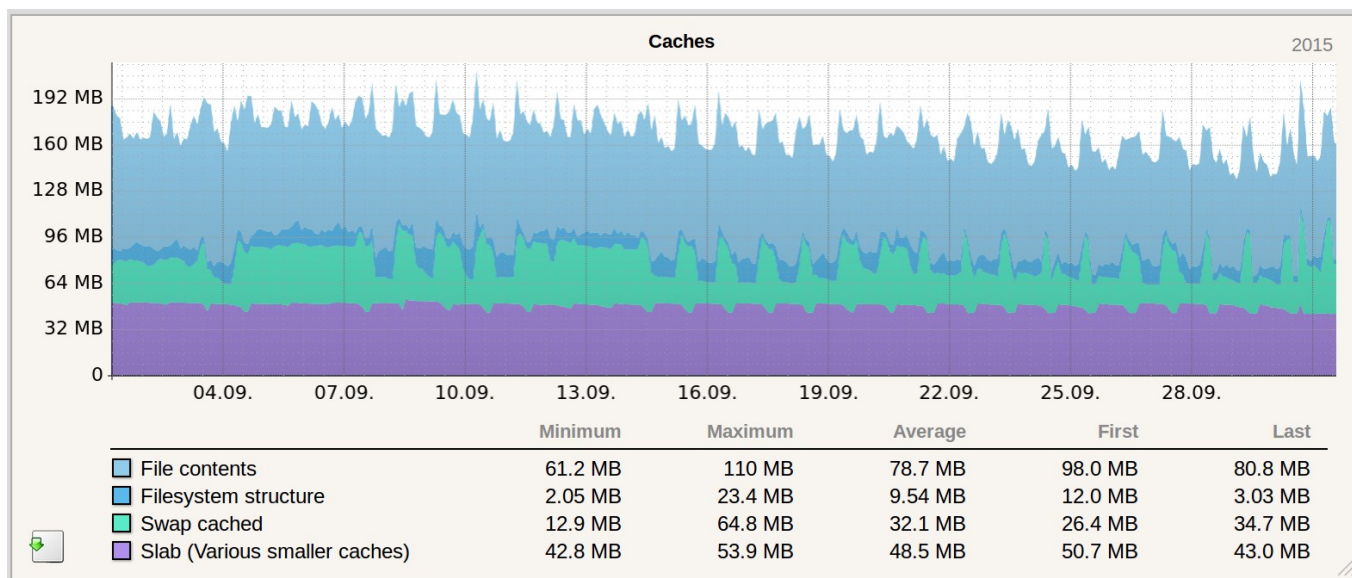
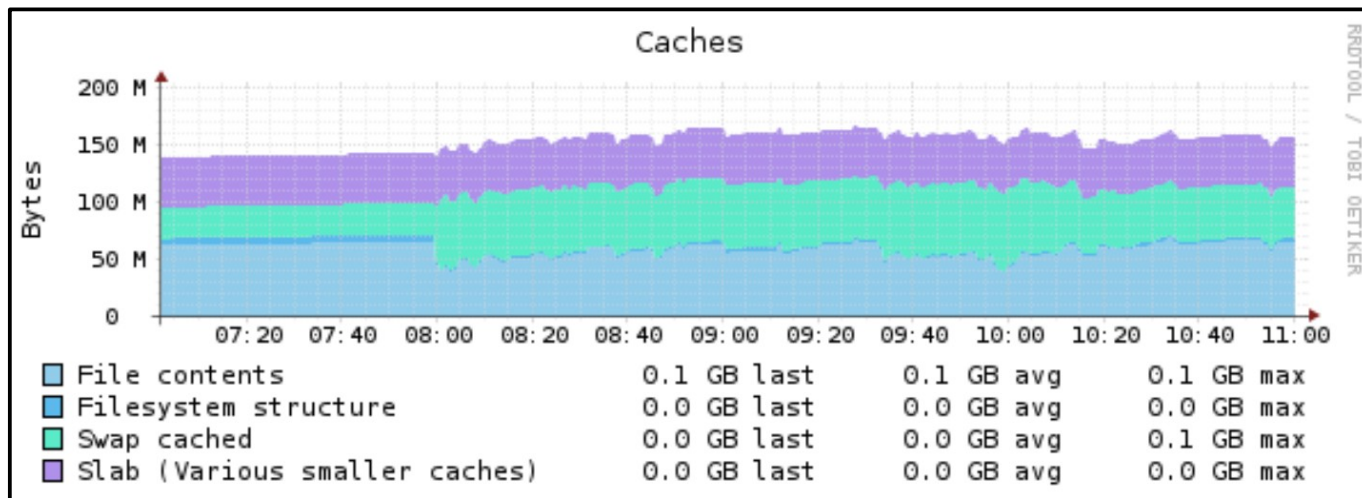
RRDTOOL / TOBI OETIKER





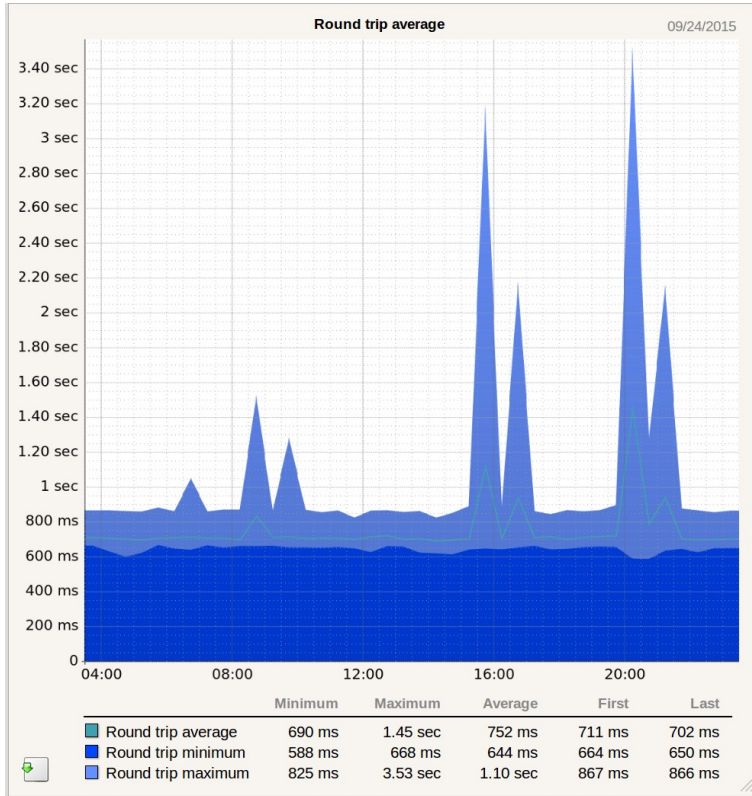
Check_MK Metrics System







Graph rendering



- Real vector graphics
- Intelligent labelling
- Interactive...
 - horizontal zooming
 - horizontal panning
 - vertical zooming
 - resizing



Graph definitions with PNP



PNP-Template for CPU load

```
$RRD = array();
foreach ($NAME as $i => $n) {
    $RRD[$n] = "$RRDFILE[$i]:$DS[$i]:MAX";
    $WARN[$n] = $WARN[$i];
    $CRIT[$n] = $CRIT[$i];
    $MIN[$n] = $MIN[$i];
    $MAX[$n] = $MAX[$i];
}

$title = "CPU Load for $hostname";
if ($MAX[1]) {
    $title .= " - $MAX[1] CPU Cores";
}

$opt[1] = "--vertical-label 'Load average' -l0 -u 1 --title \"\$title\" ";

$def[1] = ""
    . "DEF:load1=$RRD[load1] "
    . "AREA:load1#60c0e0:\\"Load average 1 min \" "
    . "GPRINT:load1:LAST:\\"%6.2lf last\" "
    . "GPRINT:load1:AVERAGE:\\"%6.2lf avg\" "
    . "GPRINT:load1:MAX:\\"%6.2lf max\\n\" "

    . "DEF:load15=$RRD[load15] "
    . "LINE:load15#004080:\\"Load average 15 min \" "
    . "GPRINT:load15:LAST:\\"%6.2lf last\" "
    . "GPRINT:load15a:AVERAGE:\\"%6.2lf avg\" "
    . "GPRINT:load15:MAX:\\"%6.2lf max\\n\" "
    . "";

if ($WARN[1]) {
    $def[1] .= ""
        . "HRULE:$WARN[1]#FFFF00 "
        . "HRULE:$CRIT[1]#FF0000 "
        . "";
}
}
```





PNP-Template for Network Interfaces

```
setlocale(LC_ALL, 'C');

# Performance data from check:
# in=6864.39071505;0.01;0.1;0;125000000.0
# inucast=48.496962273;0.01;0.1;;
# innucast=4.60122981717;0.01;0.1;;
# indisc=0.0;0.01;0.1;;
# inerr=0.0;0.01;0.1;;
# out=12448.259172;0.01;0.1;0;125000000.0
# outucast=54.9846963152;0.01;0.1;;
# outnucast=10.5828285795;0.01;0.1;;
# outdisc=0.0;0.01;0.1;;
# outerr=0.0;0.01;0.1;;
# outqlen=0;;;100000000

# Graph 1: used bandwidth

# Determine if Bit or Byte. Bit is signalled via a min `value of 0.0
# in the 11th performance value.
if (!strcmp($MIN[11], "0.0")) {
    $unit = "Bit";
    $unit_multiplier = 8;
    $base = 1000; // Megabit is 1000 * 1000
}
else {
    $unit = "B";
    $unit_multiplier = 1;
    $base = 1000; // Megabyte is 1000 * 1000
}

# Convert bytes to bits if necessary
$bandwidth = $MAX[1] * $unit_multiplier;
$warn      = $WARN[1] * $unit_multiplier;
$crit      = $CRIT[1] * $unit_multiplier;
```





```
# Now choose a convenient scale, based on the known bandwidth of
# the interface, and break down bandwidth, warn and crit by that
# scale.
$bwuom = ' ';
if ($bandwidth > $base * $base * $base) {
    $scale = $base * $base * $base;
    $bwuom = 'G';
}
elseif ($bandwidth > $base * $base) {
    $scale = $base * $base;
    $bwuom = 'M';
}
elseif ($bandwidth > $base) {
    $scale = $base;
    $bwuom = 'k';
}
else {
    $scale = 1;
    $bwuom = ' ';
}

$warn      /= $scale;
$crit      /= $scale;
$bandwidth /= $scale;

$vertical_label_name = $bwuom . $unit . "/sec";

$range = min(10, $bandwidth);

$bandwidthInfo = "";
if ($bandwidth > 0){
    $bandwidthInfo = " at " . sprintf("%.1f", $bandwidth) . " ${bwuom}${unit}/s";
}
}
```





```
$ds_name[1] = 'Used bandwidth';
$opt[1] = "--vertical-label \"\$vertical_label_name\" -l -\$range -u $range -X0 -b 1024 --title \"Used bandwidth
$hostname / $servicedesc$bandwidthInfo\" ";
$def[1] =
  "HRULE:0#c0c0c0 ";
if ($bandwidth)
  $def[1] .= "HRULE:$bandwidth#808080:\"Port speed\": " . sprintf("%10.1f", $bandwidth) . " ".
  $bwuom."$unit/s\\n\" ";
  "HRULE:-$bandwidth#808080: ";
if ($warn)
  $def[1] .= "HRULE:$warn#ffff00:\"Warning\": " . sprintf("%13.1f", $warn) . " ". $bwuom."$unit/s\\n\" ".
  "HRULE:-$warn#ffff00: ";
if ($crit)
  $def[1] .= "HRULE:$crit#ff0000:\"Critical\": " . sprintf("%13.1f", $crit) . " ". $bwuom."$unit/s\\n\" ".
  "HRULE:-$crit#ff0000: ";

$def[1] .= "".
# incoming
"DEF:inbytes=$RRDFILE[1]:$DS[1]:MAX ".
"CDEF:intraffic=inbytes,$unit_multiplier,* ".
"CDEF:inmb=intraffic,$scale,/ ".
"AREA:inmb#00e060:\"in          \" ".
"GPRINT:intraffic:LAST:\"%7.1lf %s$unit/s last\" ".
"GPRINT:intraffic:AVERAGE:\"%7.1lf %s$unit/s avg\" ".
"GPRINT:intraffic:MAX:\"%7.1lf %s$unit/s max\\n\" ".
"VDEF:inperc=intraffic,95,PERCENTNAN ".
"VDEF:inpercmb=inmb,95,PERCENTNAN ".
"LINE:inpercmb#008f00:\"95% percentile\" ".
"GPRINT:inperc:\"%7.1lf %s$unit/s\\n\" ".

# outgoing
"DEF:outbytes=$RRDFILE[6]:$DS[6]:MAX ".
"CDEF:outtraffic=outbytes,$unit_multiplier,* ".
"CDEF:minusouttraffic=outtraffic,-1,* ".
"CDEF:outmb=outtraffic,$scale,/ ".
"CDEF:minusoutmb=@,outmb,- ".
"AREA:minusoutmb#0080e0:\"out          \" ".
```





```
"GPRINT:outtraffic:LAST:\"%7.1lf %s$unit/s last\" ".
"GPRINT:outtraffic:AVERAGE:\"%7.1lf %s$unit/s avg\" ".
"GPRINT:outtraffic:MAX:\"%7.1lf %s$unit/s max\\n\" ".
"VDEF:outperc=minusouttraffic,5,PERCENTNAN ".
"VDEF:outpercmb=minusoutmb,5,PERCENTNAN ".
"LINE:outpercmb#00008f:\%95 percentile\" ".
"GPRINT:outperc:\"%7.1lf %s$unit/s\\n\" ".

"";

if (isset($DS[12])) {
  $def[1] .=
  "DEF:inbytesa=$RRDFILE[12]:$DS[12]:MAX ".
  "DEF:outbytesa=$RRDFILE[13]:$DS[13]:MAX ".
  "CDEF:intraffica=inbytesa,$unit_multiplier,* ".
  "CDEF:outtraffica=outbytesa,$unit_multiplier,* ".
  "CDEF:inmba=intraffica,1048576,/ ".
  "CDEF:outmba=outtraffica,1048576,/ ".
  "CDEF:minusoutmba=0,outmba,- ".
  "LINE:inmba#00a060:\%in (avg)          \" ".
  "GPRINT:intraffica:LAST:\"%6.1lf %s$unit/s last\" ".
  "GPRINT:intraffica:AVERAGE:\"%6.1lf %s$unit/s avg\" ".
  "GPRINT:intraffica:MAX:\"%6.1lf %s$unit/s max\\n\" ".
  "LINE:minusoutmba#0060c0:\%out (avg)          \" ".
  "GPRINT:outtraffica:LAST:\"%6.1lf %s$unit/s last\" ".
  "GPRINT:outtraffica:AVERAGE:\"%6.1lf %s$unit/s avg\" ".
  "GPRINT:outtraffica:MAX:\"%6.1lf %s$unit/s max\\n\" ";
}

# Graph 2: packets
$ds_name[2] = 'Packets';
$opt[2] = "--vertical-label \"packets/sec\" --title \"Packets $hostname / $servicedesc\" ";
$def[2] =
```





```
# ingoing
"HRULE:0#c0c0c0 ".
"DEF:inu=$RRDFILE[2]:$DS[2]:MAX ".
"DEF:innu=$RRDFILE[3]:$DS[3]:MAX ".
"CDEF:in=inu,innu,+ ".
"AREA:inu#00ffc0:"in unicast          \" ".
"GPRINT:inu:LAST:\"%9.1lf/s last \" ".
"GPRINT:inu:AVERAGE:\"%9.1lf/s avg \" ".
"GPRINT:inu:MAX:\"%9.1lf/s max\\n\" ".
"AREA:innu#00c080:"in broadcast/multicast \":STACK ".
"GPRINT:innu:LAST:\"%9.1lf/s last \" ".
"GPRINT:innu:AVERAGE:\"%9.1lf/s avg \" ".
"GPRINT:innu:MAX:\"%9.1lf/s max\\n\" ".
"VDEF:inperc=in,95,PERCENTNAN ".
"LINE:inperc#00cf00:"in 95% percentile  \" ".
"GPRINT:inperc:\"%9.1lf/s\\n\" ".

# outgoing
"DEF:outu=$RRDFILE[7]:$DS[7]:MAX ".
"DEF:outnu=$RRDFILE[8]:$DS[8]:MAX ".
"CDEF:minusoutu=0,outu,- ".
"CDEF:minusoutnu=0,outnu,- ".
"CDEF:minusout=minusoutu,minusoutnu,+ ".
"AREA:minusoutu#00c0ff:"out unicast          \" ".
"GPRINT:outu:LAST:\"%9.1lf/s last \" ".
"GPRINT:outu:AVERAGE:\"%9.1lf/s avg \" ".
"GPRINT:outu:MAX:\"%9.1lf/s max\\n\" ".
"AREA:minusoutnu#0080c0:"out broadcast/multicast\":STACK ".
"GPRINT:outnu:LAST:\"%9.1lf/s last \" ".
"GPRINT:outnu:AVERAGE:\"%9.1lf/s avg \" ".
"GPRINT:outnu:MAX:\"%9.1lf/s max\\n\" ".
"VDEF:outperc=minusout,5,PERCENTNAN ".
"LINE:outperc#0000cf:"out 95% percentile  \" ".
"GPRINT:outperc:\"%9.1lf/s\\n\" ".
"";
```





```
# Graph 3: errors and discards
$ds_name[3] = 'Errors and discards';
$opt[3] = "--vertical-label \"packets/sec\" -X0 --title \"Problems $hostname / $servicedesc\" ";
$def[3] =
  "HRULE:0#c0c0c0 ".
  "DEF:inerr=$RRDFILE[5]:$DS[5]:MAX ".
  "DEF:indisc=$RRDFILE[4]:$DS[4]:MAX ".
  "AREA:inerr#ff0000:\"in errors          \" ".
  "GPRINT:inerr:LAST:\"%7.2lf/s last  \" ".
  "GPRINT:inerr:AVERAGE:\"%7.2lf/s avg  \" ".
  "GPRINT:inerr:MAX:\"%7.2lf/s max\\n\" ".
  "AREA:indisc#ff8000:\"in discards      \":STACK ".
  "GPRINT:indisc:LAST:\"%7.2lf/s last  \" ".
  "GPRINT:indisc:AVERAGE:\"%7.2lf/s avg  \" ".
  "GPRINT:indisc:MAX:\"%7.2lf/s max\\n\" ".
  "DEF:outerr=$RRDFILE[10]:$DS[10]:MAX ".
  "DEF:outdisc=$RRDFILE[9]:$DS[9]:MAX ".
  "CDEF:minusouterr=0,outerr,- ".
  "CDEF:minusoutdisc=0,outdisc,- ".
  "AREA:minusouterr#ff0080:\"out errors   \" ".
  "GPRINT:outerr:LAST:\"%7.2lf/s last  \" ".
  "GPRINT:outerr:AVERAGE:\"%7.2lf/s avg  \" ".
  "GPRINT:outerr:MAX:\"%7.2lf/s max\\n\" ".
  "AREA:minusoutdisc#ff8080:\"out discards \":STACK ".
  "GPRINT:outdisc:LAST:\"%7.2lf/s last  \" ".
  "GPRINT:outdisc:AVERAGE:\"%7.2lf/s avg  \" ".
  "GPRINT:outdisc:MAX:\"%7.2lf/s max\\n\" ";
```



Graph definitions with Check_MK



```
unit_info["v"] = {  
    "title" : _("Electrical Tension (Voltage)"),  
    "symbol" : _("V"),  
    "render" : lambda v: physical_precision(v, 3, _("V")),  
}
```

- Units have a title
- ... that is localizable
- Example for the „render“ function:
 - 0.0055 → 5.50mV





```
check_metrics["check_mk-mem.win"] = {  
    "memory"      : { "name" : "mem_used",      "scale" : MB },  
    "pagefile"    : { "name" : "pagefile_used", "scale" : MB }  
}
```

- Translation for old existing checks...
- ... that use to wrong perf variable names
- ... or a non-canonical scale
-





- All perf variable are expected to be scaled with base **1**
- Times are in seconds (not in min or days)
- Sizes in bytes (not in MB, GB, kB, pages)
- Current is in A (not in mA)
- → The scaling is always clear
- → Old checks need to be rescaled



Perf - 0 - Meters





Perf-O-Meters: Logarithmic

```
perfometer_info.append({  
    "type"           : "logarithmic",  
    "metric"         : "rta",  
    "half_value"     : 0.1,  
    "exponent"       : 4  
})
```

Hostname	Beamer	Eiger	esx	Lastrechner
PING	3.35 ms	577 ms	151 ms	26.0 sec
Hostname	Leitrechner	printserver1	tauschzone.de	Telefonanlage
PING	93.0 ms	0.00 sec	25.6 sec	322 µs















Perf-O-Meters: Linear

```

perfo_meter_info.append({
    "type"      : "linear",
    "segments"  : [
        "fs_used(%)",
        "100.0,fs_used(%)",-#e3fff9",
    ],
    "total"    : 100,
    "label"    : ( "fs_used(%)", "%" ),
})
    
```

State	Service	Icons	Status detail	Age	Checked	Perf-O-Meter
OK	Filesystem /	 	OK - 88.6% used (354.71 of 400.18 GB), trend: +434.44 MB / 24 hours	2015-10-01 15:43:22	46 sec	<div style="width: 88.6%; background-color: #00ffcc;"></div> 88.6%
OK	Filesystem /bauwelt	 	OK - 74.9% used (693.26 of 925.25 GB), trend: +4.57 GB / 24 hours	2015-02-05 09:06:05	46 sec	<div style="width: 74.9%; background-color: #00ffcc;"></div> 74.9%
OK	Filesystem /bauwelt /chroots	 	OK - 73.0% used (143.66 of 196.86 GB), trend: -652.70 MB / 24 hours	2015-09-12 01:48:54	46 sec	<div style="width: 73%; background-color: #00ffcc;"></div> 73%
OK	Filesystem /boot	 	OK - 14.3% used (70.51 of 493.74 MB), trend: 0.00 B / 24 hours	2015-02-05 09:06:05	46 sec	<div style="width: 14.3%; background-color: #00ffcc;"></div> 14.3%
OK	Filesystem /opt/omd	 	OK - 53.2% used (49.19 of 92.53 GB), trend: +157.26 MB / 24 hours	2015-02-05 09:06:05	46 sec	<div style="width: 53.2%; background-color: #00ffcc;"></div> 53.2%



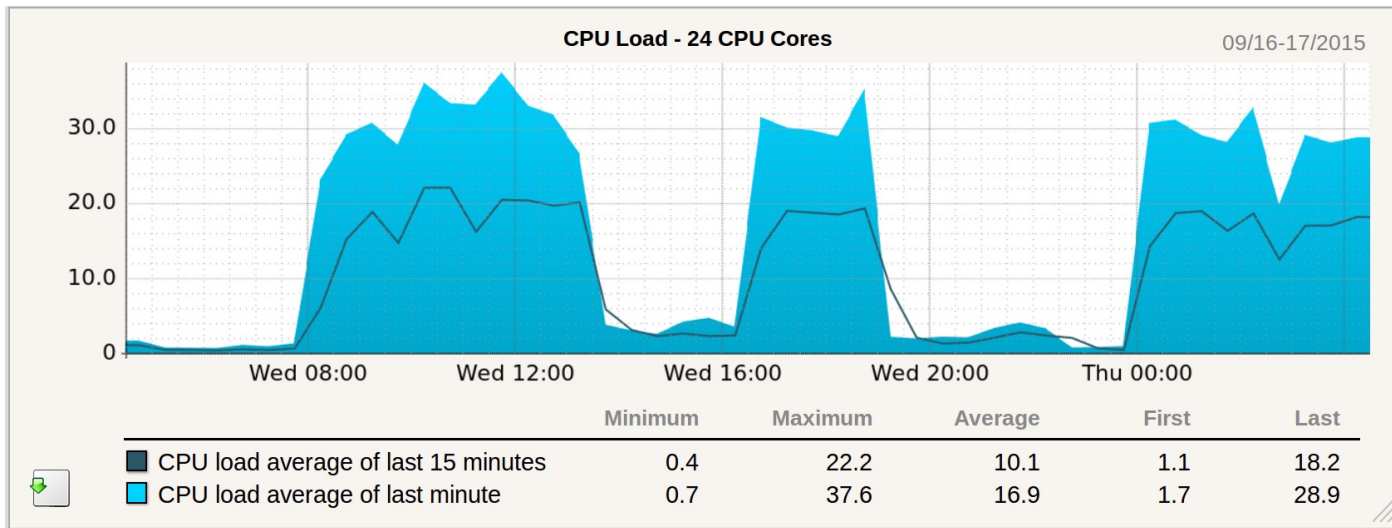
Graphs





Graph: CPU load

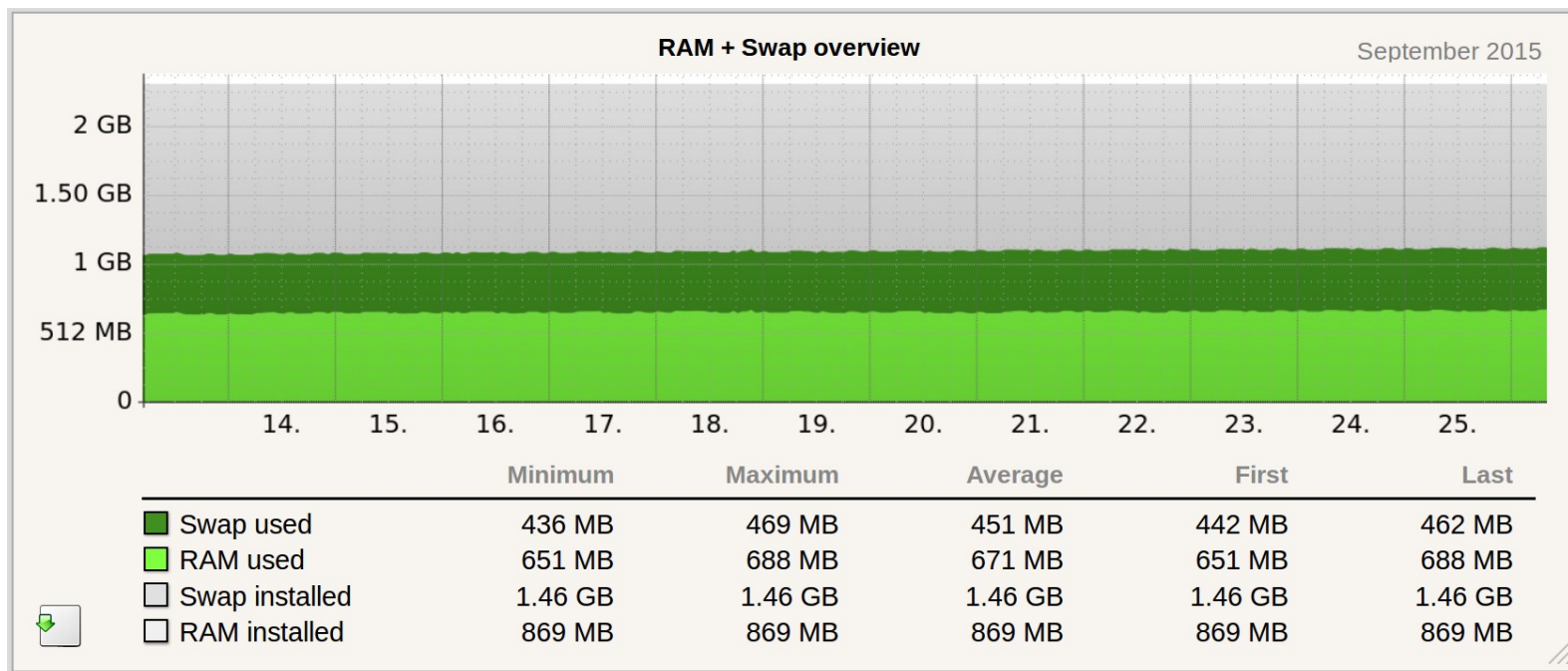
```
graph_info.append({
  "title" : _("CPU Load - %(load1:max@count) CPU Cores"),
  "metrics" : [
    ( "load1", "area" ),
    ( "load15", "line" ),
  ],
  "scalars" : [
    "load1:warn",
    "load1:crit",
  ],
  "optional_metrics" : [ "load15" ],
})
```





Graph: Memory

```
graph_info.append({
  "title" : _("RAM + Swap overview"),
  "metrics" : [
    ("mem_total", "area"),
    ("swap_total", "stack"),
    ("mem_used", "area"),
    ("swap_used", "stack"),
  ],
},
})
```





- Colors are taken from metrics definition
- Labelling and stuff automatically
- Additional features
 - Zero metrics can be dropped
 - Computations via RRDTool
 - Manual vertical range definitions
 - Manual colors/title definitions



How are graphs assigned to checks?

- Checks output perf variables
- Graphs need perf variables
- → All graphs are rendered whose variables are present

To be discussed tomorrow...



