CA APM Enterprise Manager HealthCheck

Michael Sydor

SERVICE ASSURANCE - APM

Table of Contents

Scope

A HealthCheck is a structured review of architecture and performance with a summary including specific recommendations. This HealthCheck is focused on the Enterprise Manager component of the APM solution.

Duration

A HealthCheck may be completed in a few hours or may take a week or more, depending on the number of components to examine and the availability of measurements. This best practice will how you what to do with your on-site time and how to summarize the findings.

Benefits

Completing a HealthCheck is important to the long-term success of an APM initiative and is a valuable exercise to help you improve your relationship with the client. Management of the APM infrastructure is often a significant gap which you can help the customer overcome.

Confirm correct operation and potential solution capacity

Document problems and poor practice

Understand the performance and capacity of an APM solution for the current environment

Plan for long-term APM success

PRE-WORK

Your Preparation

The primary reference for the APM product is the **Application Performance Management – Sizing and Performance Guide.pdf**. This document addresses many of the questions that will come up during client interactions but it also includes all of the background issues that will help you to understand how the EM (Enterprise Manager) works. The EM is the heart and soul of APM. If this doesn't work right – nothing works right! You also need to be familiar with the **APM 9 Network Architecture.pdf**.

Interpreting performance data takes practice. It is actually a form of triage which is discussed in Chapter 14 – Triage with Single Metrics in **APM Best Practices – Realizing Application Performance Management**. You can find specific details of EM performance situations in the best practice module **LCMM-2 EM Capacity Management.ppt**. which should be reviewed after you have read the **APM Sizing and Performance Guide.pdf**.

Setting Expectations and Scope

Day

The HealthCheck is really a post-sales activity that can be completed in about 2-4 hours. The client will have been operating an APM solution for at least 6 months. Sometimes it will be a few years before an opportunity to deliver a HealthCheck presents itself. The sooner, the better, as it is critical to block misuse of the APM solution before it gets incorrectly escalated as a product problem. As we are all interested in maximizing 'great' client experiences, you shouldn't be apprehensive about going in to see "what's really going on". This is the essence of the **EM HealthCheck**: let's make sure everything is going 'right'. And from the client perspective: "Please let us know what we need to do, to make it 'right'."

Sometimes you will get engaged with the client as a prelude to a follow-on deal. These are very important for the sales team and you will usually need to correct EM performance and stability before the client will consider a follow-on deal. They will want to confirm that their current usage is efficient and to get an idea of the remaining capacity. The engagement then becomes an **EM HealthCheck and Sizing**. The HealthCheck portion is exactly the same but you will have a few more activities to initiate before you undertake the HealthCheck, and a little bit more work to complete the sizing.

Week

If you are tasked with doing a HealthCheck over five days, it will usually be because the environment is medium to large scale, as measured in the number of clusters. A couple of APM clusters and you will have a lot of data to review and perhaps a lot of adjustments to schedule in both production and pre-production environments. Production changes are usually done after normal business hours. This is not necessary for the APM environment but sometimes agent configurations will need adjustment and this is what takes time. Specifically, you need to allow time for testing of the proposed changes, deployment and validation of the new agent configurations. Let's call this a **Medium-to-Large EM HealthCheck and Sizing**. It will have 5 to 25 clusters (25 – 250 collectors).

While the APM HealthCheck is an important activity for assessing current APM performance and capacity, it is NOT a substitute for a solution sizing. When a big deal is identified, more often for a new APM initiative, you will need to prepare a very detailed Solution Sizing and Forecast document. This is to size the initial and follow-on deployments over a 3-year (typical) period. This will involve many millions of dollars in investment, APM software and hardware, and with little or no prior operational experience of the APM environment. So the sizing document has to provide all the confidence that a scalable solution is possible and that a multi-year deployment is practical. There is no HealthCheck activity, unless there was a significant pilot period on which you could evaluate the APM cluster (or stand-alone). We will call this an **Enterprise APM Solution Sizing and Forecast**. The details of how to conduct this, as well as sample reports and tools, are covered in the *Assessment* and *Solution Sizing* chapters of the APM Best Practices book. This is a sophisticated activity and a very sensitive report. Do not attempt this until you have mastered the material and managed a couple of EM HealthChecks under your belt! You can find some more details in the next major section.

Couple Weeks

A HealthCheck over a couple of weeks is most likely going to be a certification of the APM Solution, which we can call an **APM Solution Certification**. Before your client will send in a $2M hardware order to build out 50 EM-collectors, they will expect confirmation that the solution will scale to the desired capacity. This requires a simulation of the agents and EM-Collectors – typically two or four (sometimes eight) EM-Collector instances, and then a simple multiplication to get a confirmation of the potential capacity. Effectively, you will be performing abbreviated HealthChecks every day to ensure that something hasn't slipped out of place.

This is a sophisticated activity requiring a lot of data analysis. It is all in the form of Introscope reports but you will need at least 5 complete days of data, usually in a test or temporary environment, and which will experience all manner of interruptions and strange behaviors. Add to that the provisioning, installation and configuration of the EM-Collectors and you have quite a bit of work to coordinate. You really need to allow a month or two in order to get the kinks out. This will almost always been done remotely, so you will want to have some experience conducting a EM HealthCheck remotely, as well as configuring Introscope remotely, before you undertake this type of project. You also really need a capable systems administrator on the client side to do most of the leg work. And you can also count on a few remote sessions (via LiveMeeting) to confirm or correct the configurations and operation.

Solution Sizing

For all but the most trivial of APM solutions, solution sizing requires some specifics about the client goals and delivery schedule – the number of applications and the pace of their deployment under APM. For the purposes of the HealthCheck we only want to make sure that sufficient capacity remains for growth. How much growth? That really depends on a number of factors and this is where the original plan can get out of hand. This does not mean that it is more difficult. It means that you have to guide the client through a more preparation to do.

The tool and process for this is called an *Application Survey*, which is discussed in Chapter 3 - Assessments in APM Best Practices. Your will also need to understand Chapter 10 – Solution Sizing in order to use the Application Survey Results with the Sizing Worksheet.

Unless you are already comfortable with large-scale solution sizing, this is where you want to draw the line. You can report on the available capacity and number of additional agents that may be supported. But you will not forecast the hardware needed and deployment schedule necessary to support future deployment. You can share the *Application Survey* worksheet, to start gathering this information but you will need to divide the HealthCheck from the Solution Sizing and finish your preparation or involve another resource.

Take a look at the presentation **APM Best Practices - Sizing Overview.ppt** to understand the sizing process for CA APM. If you want to know more, then look at the training module for **LCMM-1 EM Sizing and Capacity Forecast.ppt**.

DELIVERY

Access

Before going on-site you need to confirm that you will have someone to access the APM Workstation. It is very rare that you will be able to login to a production system, let alone hook your laptop into the operational network. What you may have had latitude to do during a pilot is simply not allowed with production.

In addition to APM Workstation access, you will also want to confirm that you will have PowerPoint and a browser with connectivity to the outside. You will need to screenshot and annotate – PowerPoint is best for this. You can also use Word or something equivalent. Screenshot everything! This is all you will have on which to base your analysis and recommendations. You will also be transferring files (ppt and pdfs, typically) which you can do via a browser-based (webmail.ca.com) email session. I find that it is very rare that even a USB port is open on any machine that touches the production network. Be prepared! Sometimes your client will set up a machine which includes an outlook account when they are otherwise blocking access to foreign email via a browser. Ask before and avoid surprises.

Activities

In this section we will discuss how to use you time on-site. For a cursory EM HealthCheck, half a day on-site is usually sufficient, and you will want to allow another half day to finalize the report and a short meeting to deliver the findings and recommendations. My preference is always to do this in PowerPoint, which results in the shortest time in the reporting phase. PowerPoint also has one special advantage (guideline) – your recommendations should never exceed a single slide! Some folks prefer Word – I guess they feel a lot of pages are 'better'. Of course, if you have five days for the HealthCheck a written report is usually the better choice.

The end result, in all situations, is to make sure you have immortalized your facts and findings so that the client may repurpose them for their own internal meetings and reports. If your recommendations are going to involve additional investment in hardware, services, training or mentoring – executive support is essential – make it easy for them to take your message forward.

Architecture Discussion

Allow 20 minutes to review the physical deployment with some details for the platform configurations and types of applications being monitored. You will then use the following questions to quickly assess the current teams "facility" with APM:

- How many metrics are you collecting?

- How many agents have been deployed?

- How many applications are under management with APM?

- How many folks are configuring APM dashboards?

- How many folks are utilizing APM reporting?

- How many folks are doing custom agent configurations?

A great benefit of this engagement is that you can easily show the client how to quickly get this information and start to really understand and manage their APM environment.

A related line of questioning, especially if there have been problems with the APM environment, is to ask for a list of the incidents that have occurred over the last six months. Most clients simply do not do incident reporting for the APM environment, to the same degree that they do for their monitored applications. So especially if they have been having problems, try the following questions:

- How many times were workstations unable to connect or slow to connect?

- How many times have the MOM and/or collectors been recycled? Is this increasing in frequency?

- Is the user population growing? How do they know?

- How much capacity remains for the current APM configuration?

These are also questions that you will be able to answer for them, as well as the invitation to get incident tracking going for APM.

If you find that the architecture is suspect, make a note for yourself and wait until you have confirmed, with real data, that there is in fact a problem. It is OK to say "I have some concerns", so long as you immediately follow with "Let's open a Workstation session and see what is really going on". Never jump to conclusions without some data to back it up.

Historical View

Answering the previous questions is the bulk of what the HealthCheck will do. It's really easy. Open the investigator and Navigate through the super domain to the Enterprise Manager folder. Click on the folder and you will get the following display:
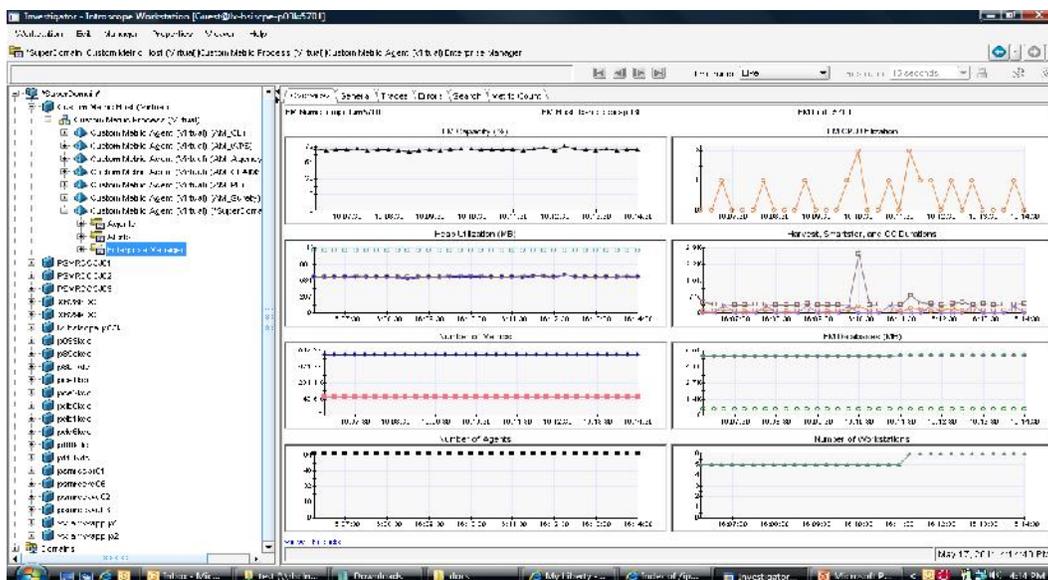
This is an opportunity for some mentoring as many clients have never seen this screen before. While that is sad, you can review what each of these metrics means, especially if they give rise to "concerns".   Screenshot this view and paste into your PowerPoint. Add notes as you describe what you see or make a checklist of things you want to look into more deeply.

What you really want is to generate are historical views at both 6 hours and 30 days. These will be more useful for capturing trends. And if the data is available, you will also want to look at specific metrics for 3-6 months, which can make for some really powerful trends.

This is also where you will find you first "anti-pattern" – there is no historical data available! The client has arrived at a useless configuration of SmartStor Tiers – typically 1-7-7. This means that they have only 15 days of data – sometimes only 7 days! Why would they do that? That's a question you need to ask. Often, they are still afraid that SmartStor data is going to spill out of their data center! You need to educate them about the SmartStor and show them the Disk Space Calculator - an easy problem to remedy. Make a note and start answering the questions we have pending. You will need some of those answers to get the Disk Space calculator working. And reserve a little time to show them how to restore the EM configuration to something more useful, with respect to SmartStor Tiers.

The details of how to interpret what you find is in the later section "What to Look For".

As you get into clustered environments you will end up with a MOM and up to eight collectors for which you will need 2 screenshots each. So this can easily become a chore but you absolutely have to look at every collector. A cluster only operates as fast as the *slowest* collector. Make sure you find out who is holding back performance and get that collector corrected!

It will be most efficient if you collect all of the 30 day views and then follow with all of the 6 hour views. This is especially important when the environment is operating very slowly. Jumping back and forth with different historical view settings puts a lot of strain on the workstation and collector queries. Also use the default data resolution, for each historical range, to further avoid over-taxing the APM environment during query.

Configuration Changes

The inevitable outcome of most HealthChecks will be serious recommendations for changes to the APM configuration. In general, if you are a pre-sales consultant, you should avoid making <u>any</u> changes in a Production environment. The client will have a process for change control and you should not offer to help them subvert it. Feel free to do whatever you want in a pre-production APM environment. Whoever screws things up is not going to be thought of well in the future. You are trying to build a relationship with the client. Do not become their poster-person for breaking production!

Our Services Consultants are the folks who should be rolling changes into production. They do it all the time. Even if you get lucky, making the change, you will have crossed a line that clients will thereafter leverage with abandon. Your time is "free". Consultants need to be scheduled and paid for. This is not the type of relationship you want to foster. Your job is to show them what to look for and how to make the change. The client should really complete the task on their own, following their own internal processes.

ENTERPRISE MANAGER

Configuration changes for the EM are really easy. You can encourage your client to make those changes directly, or drive the conversation to make production changes to the EM configuration possible. Many clients have draconian rules about making changes in production because of the potential to impact business. Making changes to the EM are simply <u>not</u> going to impact business. You want to establish two levels of change control. One level for the APM configuration, and another level for everything else. This is really critical to, of course, correcting a mis-configuration but more importantly, this can negatively affect the overall success of the APM initiative, as we will discuss later.

AGENTS

Configuration changes for the agents are a bit more sensitive because you know that a typo can prevent the agent, and sometimes the application, from starting correctly. Especially dangerous is making a change in the tracing configuration without testing it in the QA environment, and ending up with a high overhead situation.

This is an opportunity to emphasize the necessity for a mechanism (process) to change agent configurations, including the testing phase. An APM deployment is simply not a one-time event. Visibility needs change, as well as the need to correct inefficient monitoring configurations. Your client really needs to understand that there will always be some small change that should be accomplished sooner, rather than later. When folks can't get the visibility they need, they just lose confidence in the technology. APM has to respond rapidly and precisely to keep the initiative successful.

Chapters 11 and 12 of the ***APM Best Practices*** will cover what your client needs to know about useful load generation and the baselines that will allow them to deploy with confidence into production. The *Configuration Baseline* is what confirms that they have a safe APM configuration. The *Application Baseline* will confirm that the APM visibility is appropriate.

Wrap-up Meeting

Resist all attempts to deliver full recommendations that same day you are delivering the HealthCheck. For sure, fix the SmartStor tiers, and anything else that can be done quickly. Otherwise, you are creating a sink-hole into which you will fall! Take the time to review your results and recommendations. Setup a follow-up call and use LiveMeeting if the schedule is tight. You want to make this exercise something that is useful and predictable, in terms of time commitments by the customer. You will build a much better relationship if you keep scope and show up periodically. Don't try to do everything in a single visit.

We've included a variety of reporting outlines for you to follow, when you get to that section. So long as you have got plenty of screen shots – it will be pretty easy. In general, you don't want to say anything that you cannot also show them why and how you know that to be true.

You should allow 30 minutes meeting time for a 'good' report and 1 hour if there are problems. Typically, there is always something to be adjusted. The main questions you want to address are as follows:

- Is the environment stable? What can we do to fix it?

- Will the current architecture scale? How much longer before we need additional capacity?

- Is migration to a newer release necessary to meet current goals?

- What can we do to make better use of our APM investment?

Written Report

For longer engagements (5 days) or for initial APM deals, you will very likely need to prepare a written report. This is where your discipline is most important – do not say ANYTHING that you cannot support with a screenshot or other summary data. This will come back to bite you, if your recommendations are not successful. In general, you want to report on what you have observed, what the product recommendations are or what generally-known performance criteria demand, in terms of supporting hardware. If you make a sizing recommendation, include all of your artifacts. The accuracy of your sizing is very dependent on what the client has supplied so you need to make sure that everything is traceable doc or a CA APM doc. If you can't trace back a requirement, just be sure to say so. Every solution architecture has some wiggle room and some 'soft spots'. Just make these points clear.

You will also find a sample 'large-scale' sizing exercise in the Artifacts chapter of this cookbook. You will need to do these for deals in the 2M$-10M$ range, especially if this is their first implementation of APM. Many of your screenshots will be repurposed for the doc but try really hard to keep the discussion "high-level". In this case the HealthCheck is a summary of a deployed environment and this will be used to check the sizing forecast, with the operational reality.

Please also consider having a more experienced practitioner review your document if you have any concerns. Use the Team-WIU alias – that's what it is there for!

What to Look For

As you conduct the HealthCheck, you are going to 'see things'. What do they mean? This section will give you some starting points to consider. The ultimate arbitrator will be the APM Performance and Tuning Guide (see the Reference chapter) and the experience of you and your peers. About 80% of the time, the two primary screenshots (6 hrs and 30 days) are all you need to share (on the WIU alias) in order for folks to give you some insights.

Cluster Characteristics

Using the search tab you can quickly bring the cluster into perspective. Make sure to expand the name column so that you can read off the full collector name. Alternatively, you can enter values into a spreadsheet as you find them.
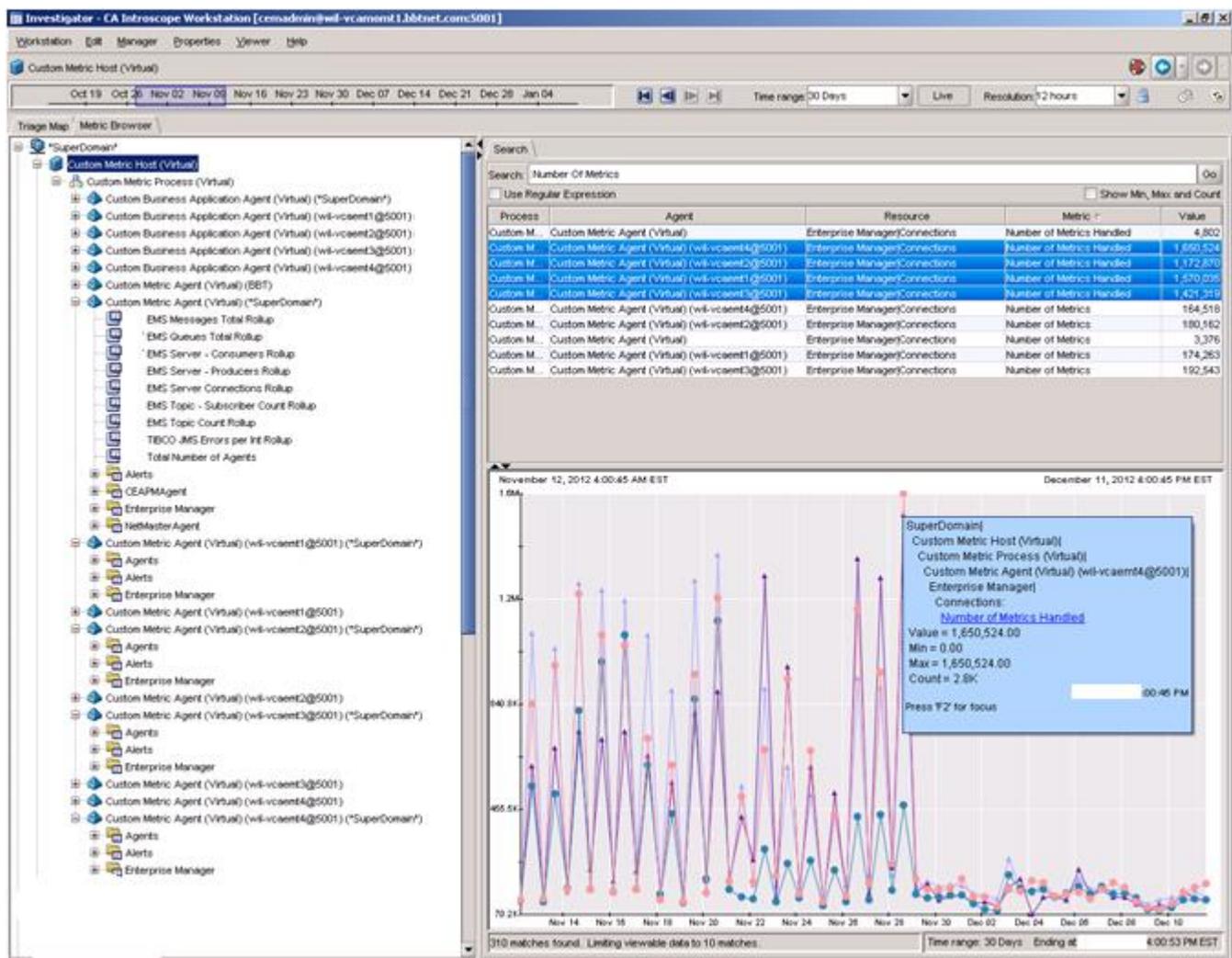
The easiest way to see how well balanced a cluster is, is to look at the number of metrics (live, historical/handled). You will later contrast this with other performance metrics.

Also note how all of the collectors are opened in the Investigator view. This will facilitate quicker navigation as you collect the initial screenshots of the Enterprise Managers at 6 hours and 30 days. Do all of the collectors at each time range so you do not have to suffer needlessly! This is helpful when the APM environment is already suffering. It seems easier to do all the 6-hour views and then follow with the 30 day views.

SUPPORTABILITY METRICS

While the Enterprise Manager Overview is most convenient for communicating the relationships among various performance metrics in overview, the Supportability Metrics will be more useful when you are drilling down to document which collectors are behaving 'exceptionally' (in a bad way!). However, not all of the metrics in current use will have useful values.

When at 30 days historical view, get screen shots of the following:

- Connected Collectors

- EM Overall Capacity %

- GC Heap Stats, MOM only

- GC Heap Stats including Collectors

- Metrics By Agent

- Number of Agents

- Number of MOM Metrics

- Number of Metrics, All Collectors

- Number of Workstations

EM Incident Reporting

Frankly, this topic is beating a dead horse. Most APM problems come about because simply NO ONE is managing the performance and capacity. They have to start tracking incidents. Someone on the client-side needs to have their feet held to the fire!

If you are getting incidents, there are three situations to look for. First, has the growth of the environment simply consumed all available capacity? This can be due to lots of deployments or extremely wasteful agent configurations (too many metrics). Make sure you find out the root cause. Second, is the environment under-sized, usually evidenced by low number of metrics and agents but otherwise poor performance? Third, are the number of workstations in use increasing, decreasing or on a plateau? This can sometimes reveal unreported incidents or performance problems.

As to why this situation comes up, it is simply due to lumping APM as analogous to the experience with any other monitoring tool. Very few folks would every use the 'older generation' of monitoring tools, they never needed any attention, and no one certainly could deploy an agent that would product thousands of metrics. APM changes the game with dozens and often hundreds of users, very dynamic and high capacity considerations, and the odd chance of someone configuring an agent with 50,000 metrics!

EM Health

When the EM is working correctly, everything is in balance: metrics, agents, I/O, queries, persistence. When performance is problematic you need to find out which performance parameter is out of balance.

There are a lot of EM metrics but the following seven metrics really give you all you need to know.

NUMBER OF METRIC GROUPS

Metric Groups are the fundamental element necessary to support Dashboards and Reporting. A growing APM initiative will have a smoothly increasing number of metric groups. Any sudden spikes are likely due to weak training and process, or other problems. A failing initiative will have a static number of metric groups. It means that no one is really using the APM system, often because all changes are prohibited in production! It sounds crazy but when you can't build a metric group or even run a report, folks just don't see the value in using APM.

Of course, one huge problem with metric groups is when they are composed of wildcards, the query times can be excessive. It is not the number of metric groups but the number of metrics that a metric group can match. Sometimes this is because of weakness in the QA testing – the production environment just gets a lot more metrics matching. Other times it is just poor training/education.

Chapter 12 – Baselines, in *APM Best Practices*, will show them how to identify the specific metrics that matter, so that they can break their dependence on the 'wildcard', and get more efficient use of their APM.

Professional services has a tool that will measure exactly how many metrics a metric group will match which can help you identify the worst offenders of 'wildcard' abuse. In reality, you only need to open a couple of management modules to find the evidence. Look for metric groups like `JSP(.*)` or `(.*):Stalled Metric Count`. The solution – training – or at least a lunch-n-learn!

Here is an example of BAD:
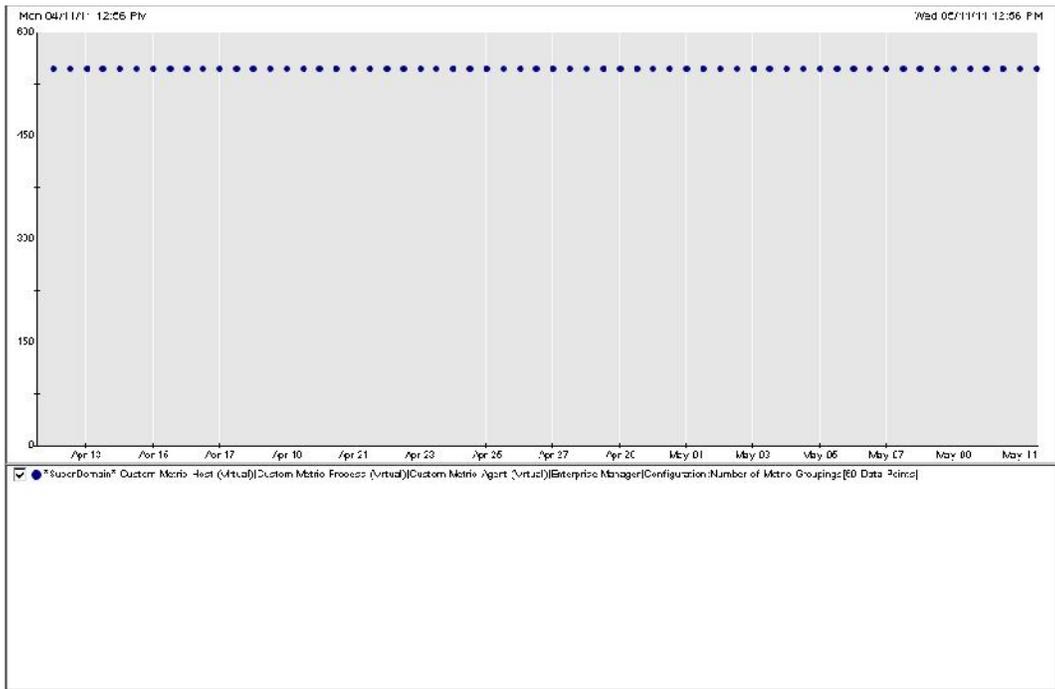
NUMBER OF HISTORICAL METRICS

Historical metrics will always be higher than live metrics. When these get to 5x or greater than live metrics, this when you will often begin to have problems.

Whenever an Introscope agent comes on-line, it generates a temporary name until the app server name service comes on-line. Whatever metrics are generated will live on forever because the EM really hopes that agent will come back, even as it has adopted a new name. So we always expect that the historical metrics will be at least a little bigger.

But when an agent comes up multiple times, and each time with a slightly different name, the historical metrics reflect each and every instance. Sure, we may age the metrics out after 60 minutes but they will always be in the EM metadata, as the EM hopes and prays that that specific agent will come back, someday. Of course, they never do come back and what we end up with is a metadata leak, of sorts.

Applications that are doing batch processing are particularly strong offenders. The client may think it is real-time request processing but it is actually a unique batch process, each and every time – until the EM collapses under the weight of all that metadata. Other abuses are possible.

There are some metadata purging tools but this will often require correction of the abusers and a new SmartStor.

Here is an example of TYPICAL:
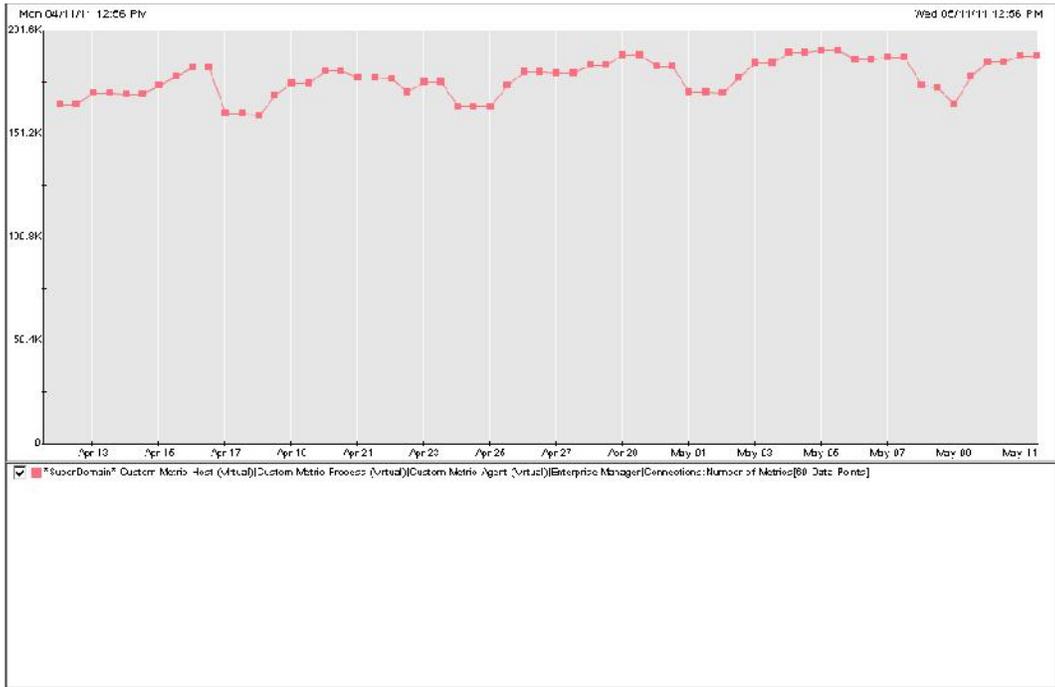
This curve varies a little and you can just detect an increasing trend. For an actively growing APM initiative this is expected. At some point it should stabilize but as long as new agents are being added you can expect the historical metrics to keep increasing.

NUMBER OF METRICS HANDLED

This is the number of metrics that the agents are currently sending. The first part of EM performance is to store all of the incoming data, resulting in the *Harvest Duration*, discussed later. This is the significant number in terms of overall EM capacity. You can get up to 400k metrics per collector – providing that the hardware and configuration are correct. The doc will say 500k metrics but you will find that about 100k metrics are consumed with the Heuristics and Application Map features. If you have CEM deployed then this will also generate metrics and consume capacity but usually at a much lower rate than a typical JVM agent.

When a collector cannot handle the incoming metrics then this will start to put pressure on the query time and workstation update time. Users will complain about slow queries or slow updates or slow logins. That's what you need incident reporting to expose because the EM will find a way to keep everything moving – until it can no longer function and comes crashing down. The warning signs are always there, weeks before. It is just often the case that nobody is looking.

Here is an example of TYPICAL:
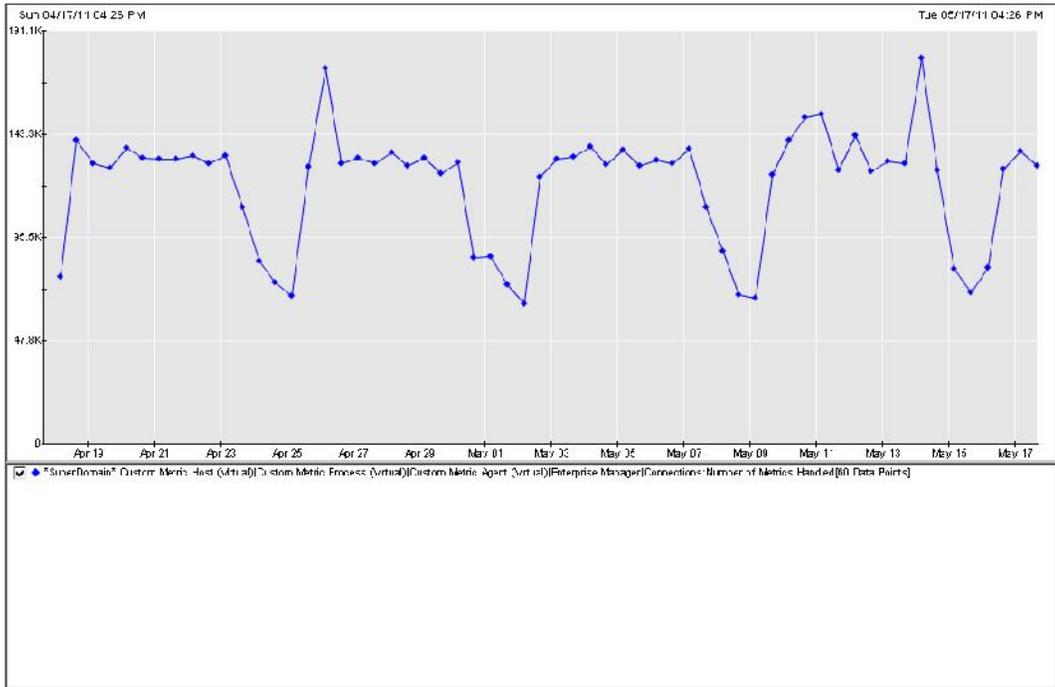
Here we can see some weekend behavior where many applications are taken down and restarted. The peak metrics are not consistent, which seems a little odd. I would expect the peak metrics to be the same at each start–of-week but that is another question. For now, nothing is unsafe – be happy!

Here is an example of a GROWING INITIATIVE:
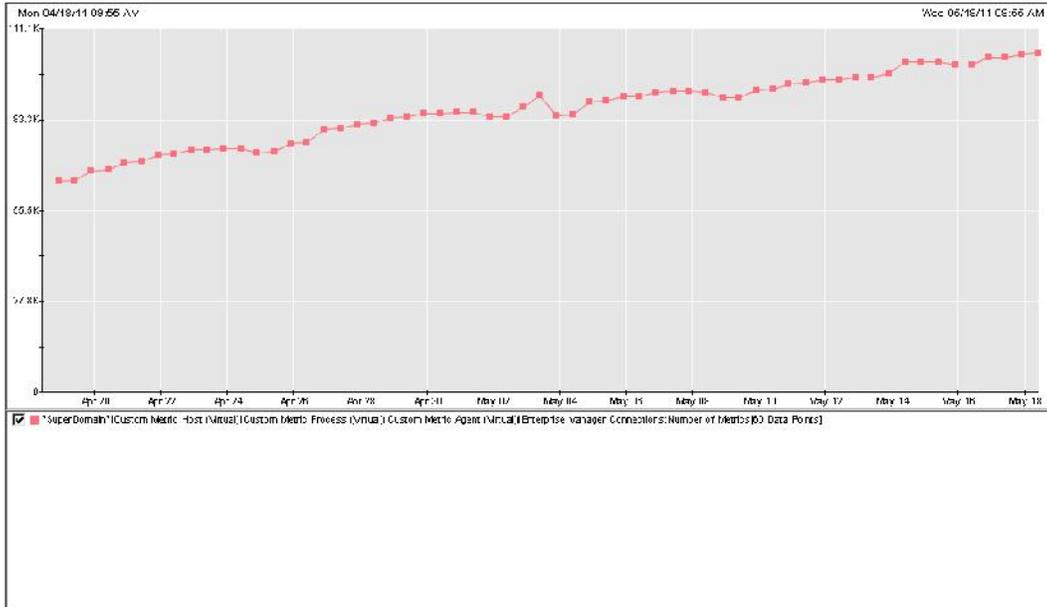
Here new agents are arriving at a consistent pace, which would be confirmed by looking at the growth in the agents. If there is no change in agents, then this is likely a slow burning historical metrics leak. The usual suspect here is with the backend metrics – lots of unique SQL that is never seen again! Look ahead to the section on metrics leaks for more discussion.

NUMBER OF METRICS, CLUSTER PERSPECTIVE

A cluster requires a different set of views to efficiently overview what is going on.

Metrics Load per Collector

From the MOM, navigate to Custom_Metric_Process_(Virtual)|Custom_Metric_Process and execute a search on "Number of Metrics". Sort the column "Metric" to get the following view:

Here we can immediately see that the cluster is unbalanced. You should then investigate if the load balancing has been disabled and talk with the client as to why they are using this configuration.

Make sure to expand the Agent column (see below) in order to capture the full metric name. This will help you out when it comes time to summarize the various environments. See the Reporting section for more details on what to put into your spreadsheet.

Agent Load per Collector

From the MOM, navigate to Custom_Metric_Process_(Virtual)|Custom_Metric_Process and execute a search on "Number of Agents". to get the following view:

Here we can immediately see that the cluster is unbalanced. You should then investigate if the load balancing has been disabled and talk with the client as to why they are using this configuration.

If the cluster is composed of disparate hardware it is important to realize that the performance of the cluster is limited by the *slowest* collector.

NUMBER OF WORKSTATIONS

When an initiative is growing and successful, the number of active workstations will also be growing. When the initiative begins to experience problems with stability or performance, the number of workstations will begin to decrease. This is because folks are frustrated and abandoning the project. This is sad and very bad for the long-term success of the client. And this is why a periodic HealthCheck is so beneficial to getting an APM initiative back on track. You want to get in and root out the problem before users start abandoning the technology.

Here is an example of TYPICAL:
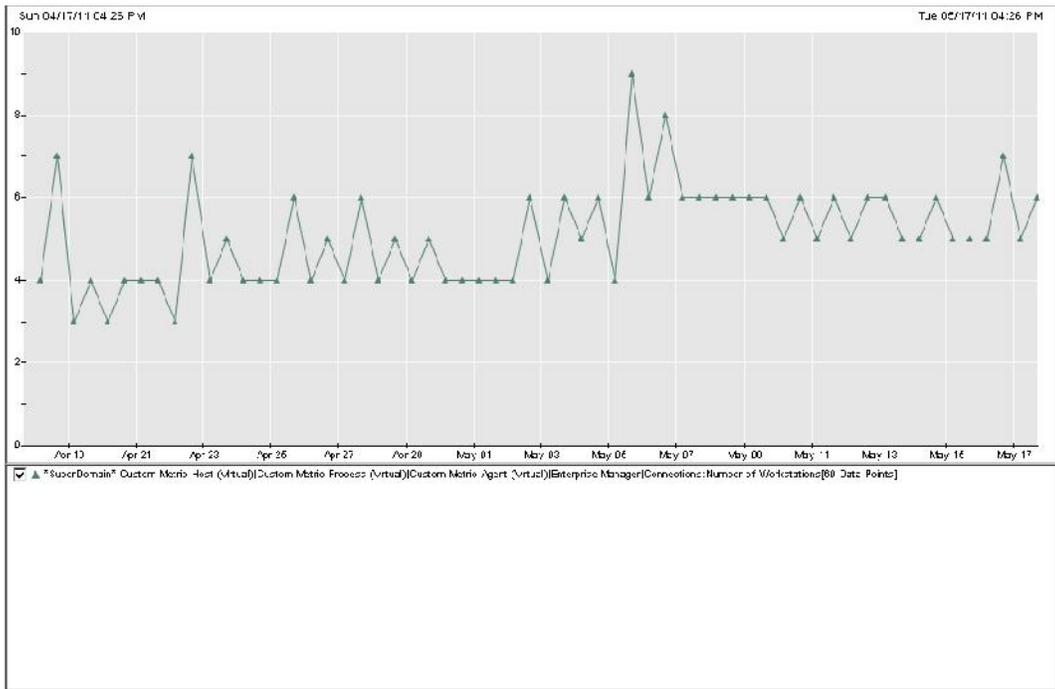
When the initiative is mature, having finished its deployments, you can expect the number of workstations to plateau. Sometimes the novelty wears off and many casual users also depart, resulting is a slow decrease of active workstations. This is normal, provided that there are no performance incidents occurring. Folks may be relying more on reporting, than dashboards, and this will be reflected in an increase of workstations users after business hours when it is more efficient to generate the queries underlying the reports.

GC DURATION

The EM is an example of a real-time application. It has to complete a variety of tasks without a specific interval. When one of those tasks goes long, all the other tasks are compromised. The amount of time spent in GC is the lesser of the durations. The Harvest Duration, discussed later, is more significant. But if there are configuration difficulties, or excessive metadata, then the GC Duration will increase significantly.

Here is an example of TYPICAL:
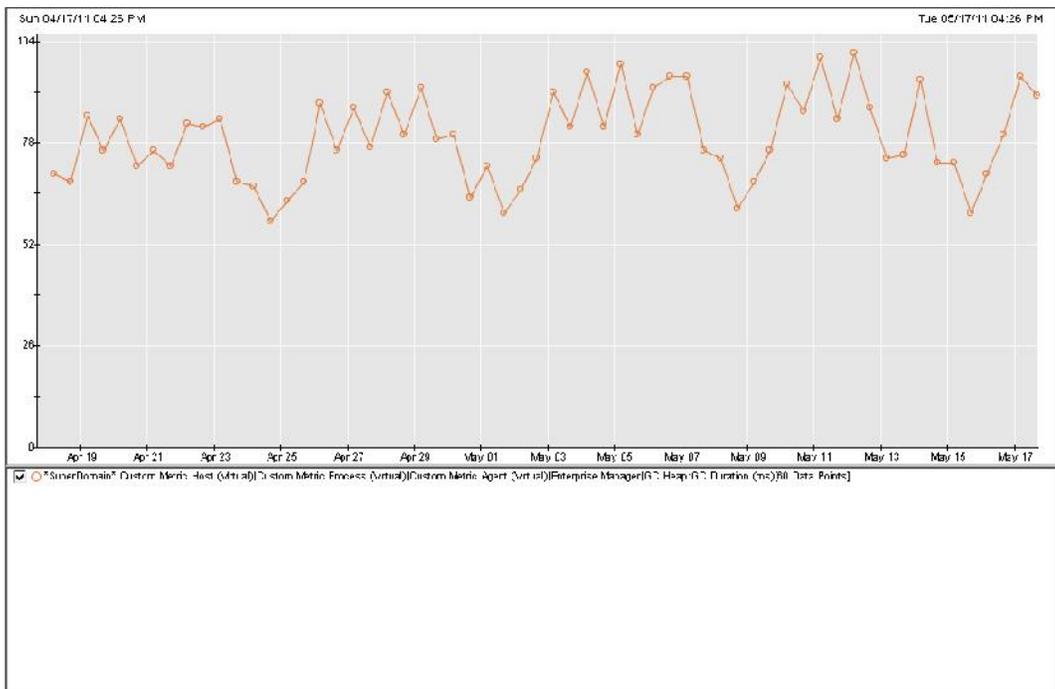
Here the dips are due to the agent population changing over the weekend. We can also see a slight upward growth and when this is accompanied with a growth in the number of agents, this is a 'good thing'. If the number of agents is static, this becomes a 'bad thing' and you will want to look for historical metric leaks.

Just like any other application memory usage should be consistent without wide swings in the amount of time managing memory. If wide swings are present, and the number of agents consistent, then tuning of the JVM parameters is indicated. If the number of agents is increasing, the memory footprint will be increasing as well.

Here is an example of TYPICAL:
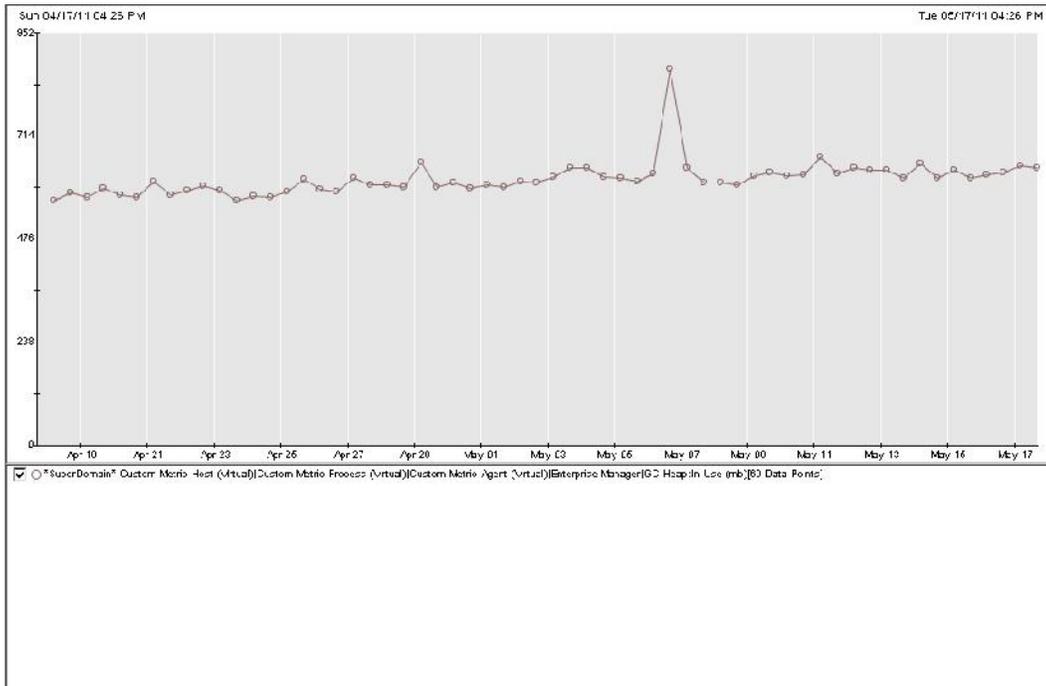
This doesn't seem problematic but if you look carefully, there is a spike occurring every 8 days, and it is getting bigger. I don't know why. Hopefully, as more folks complete HealthChecks in the field, we will start to get insight into some of these on the 'anomalies'.

HARVEST DURATION

This is probably the most important metric in assessing a performance or capacity problem. This is the duration it takes to process all of the incoming agent connections and metrics. When this exceeds 3.5 seconds, very bad things are going to occur as all other activities are subordinate to the Harvest and this means less time for the remaining EM activities.

Here is an example of TYPICAL:
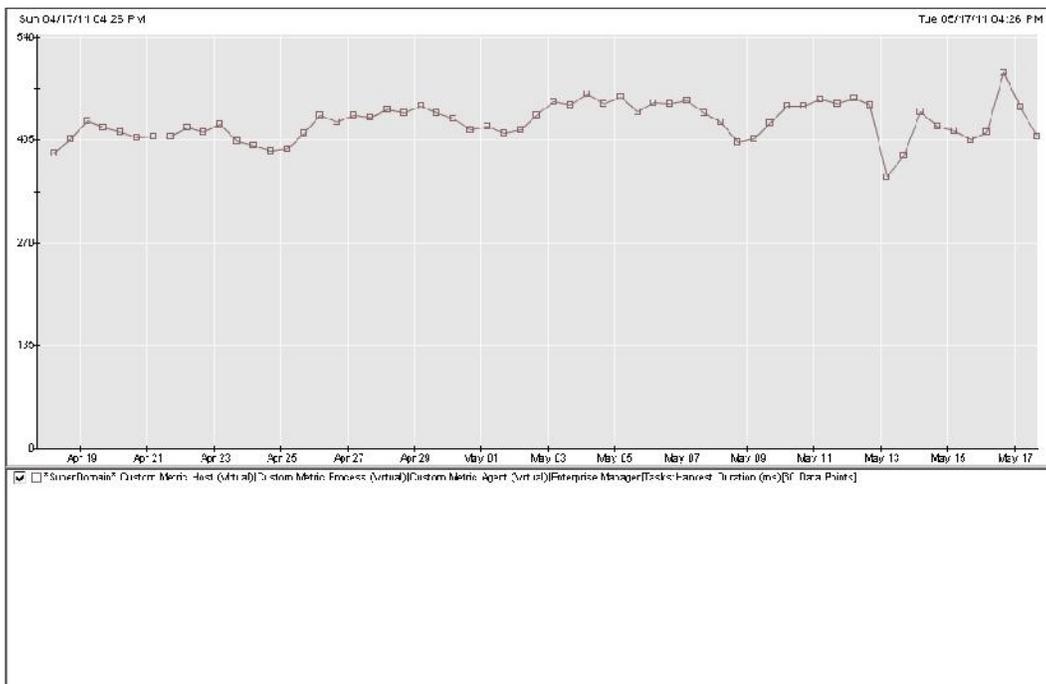
HARVEST DURATION PERIODIC SPIKES

This is a more interesting situation which shows that the platform is approaching saturation but this can be due to a combination of platform sizing, metrics load or Calculator load.

The six hour view of the collector reveals the following:
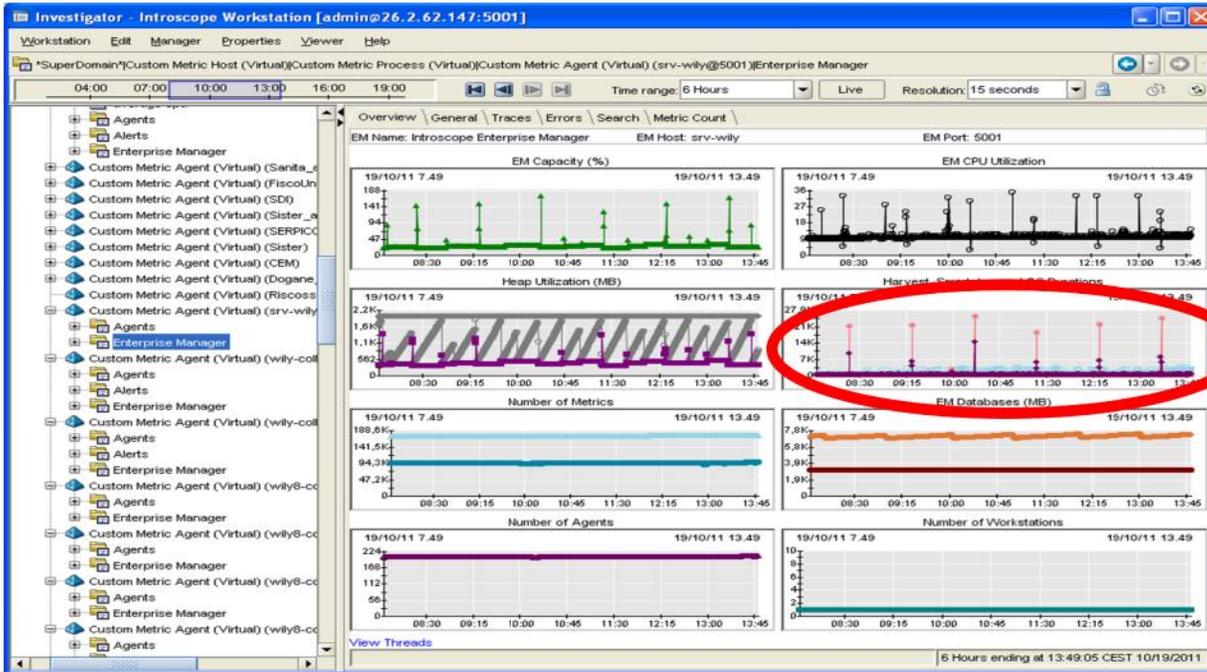
And in this case, here is a screenshot of a fully loaded (500k metrics) but otherwise correctly sized platform for comparison (from QA testing with simulated agents):
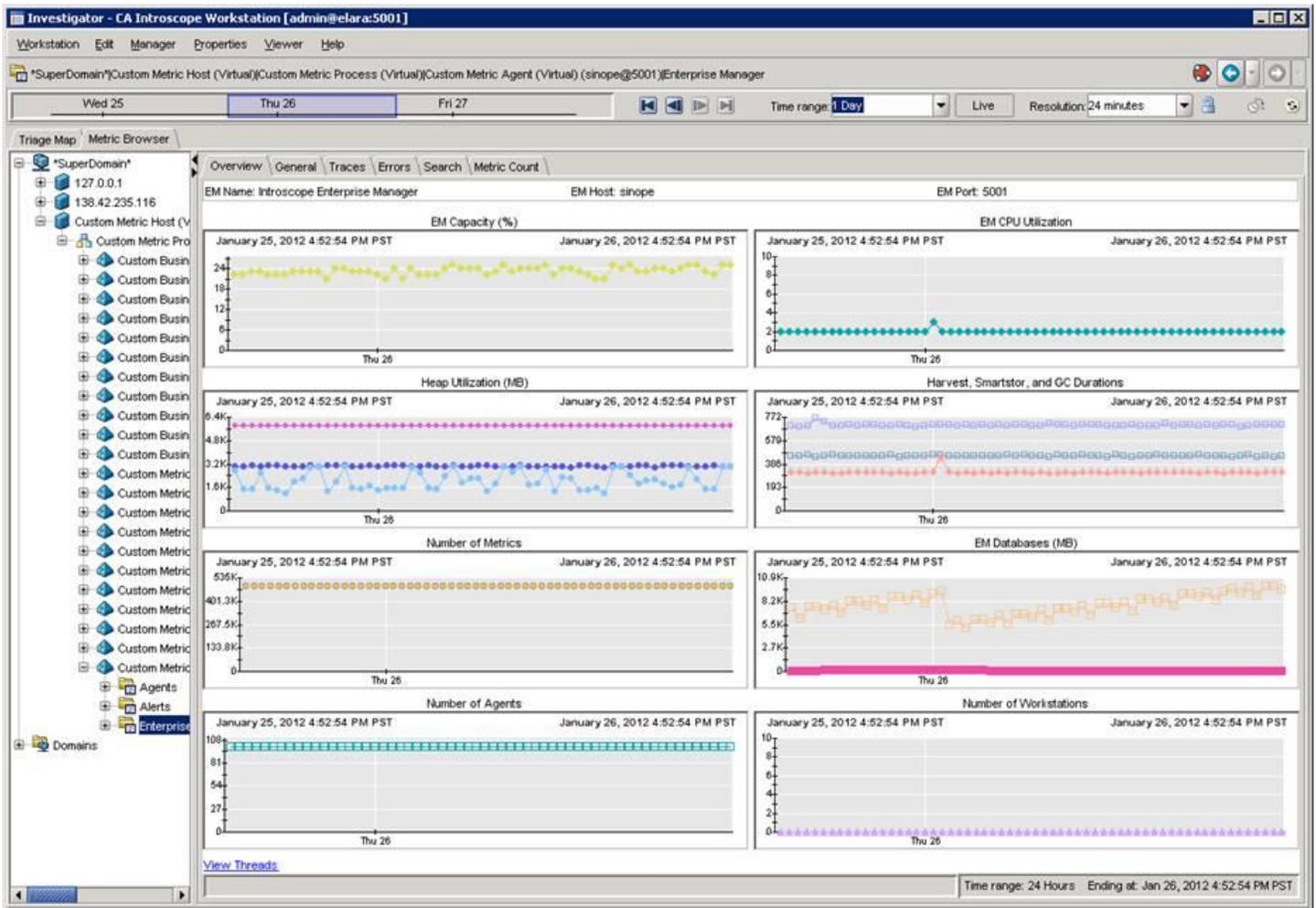
Here is the detail of the SmartStor activity that is happening hourly, corresponding to the above collector:

Smartstor Disk Write Bytes/sec
500K metrics (2-hour zoom-in)
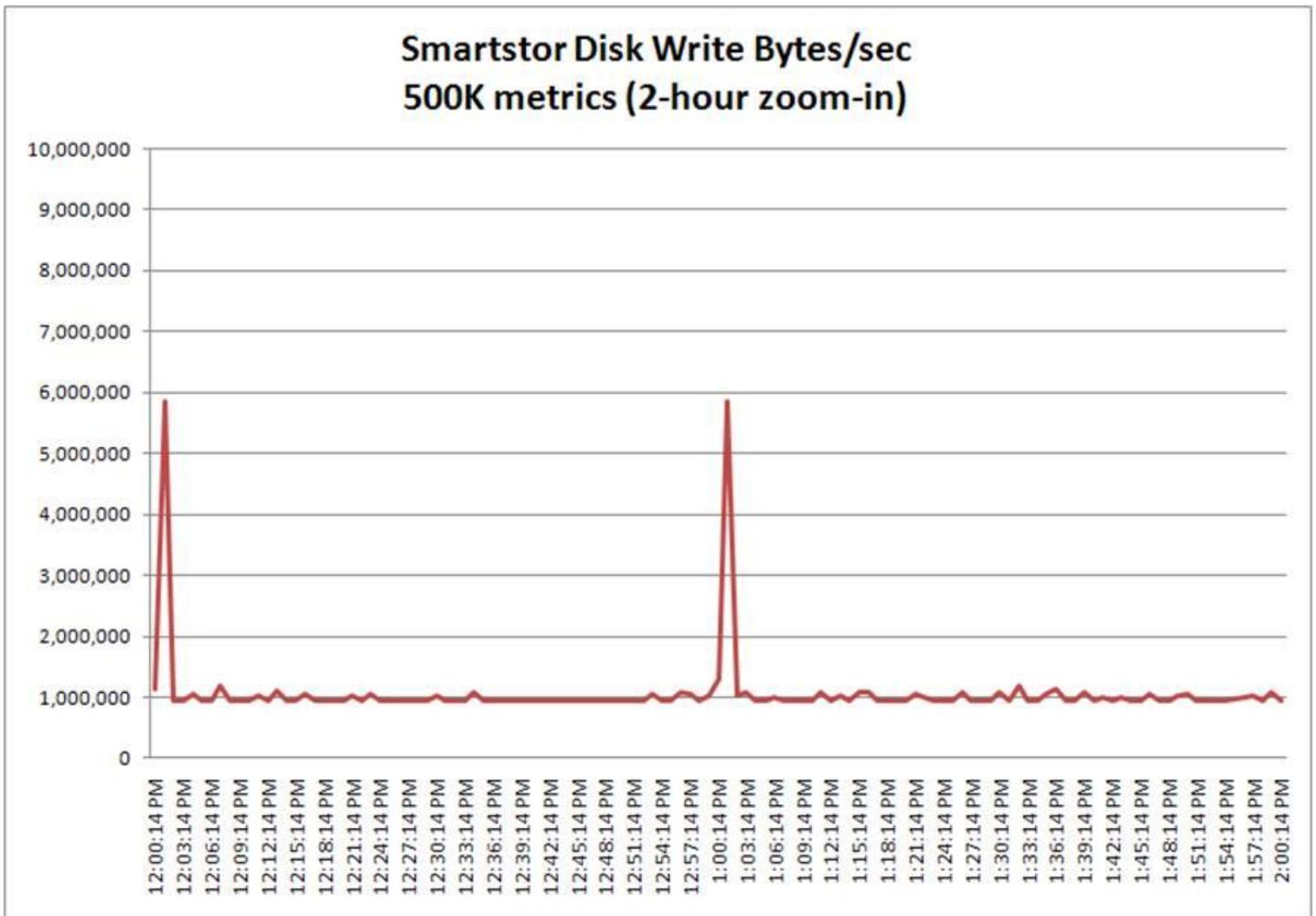
Over-instrumentation

For a new APM initiative, over-instrumentation is the most common problem, in terms of wasting or obliterating EM capacity. There are a number of situations where it is allowed to occur and they are all self-inflected. Processes like the *Pre-production Review* or the *Configuration Baseline* and *Application Baseline* will help correct the problem. Again, the EM gracefully accommodates these abuses for months but the end result is either poor capacity utilization, performance degradations or instability.

So this is activity number three. At this point we have reviewed the APM architecture, assessed the current performance and capacity (6 hr and 30 day screenshots) and now it is time to dig into what is responsible for the overall health. The technique is to simply follow the Baseline practice (Chapter 12), starting with the high invocations metrics. No need to build a metric group (which would be impossible to evaluate anyway), just set your view to 'live', hit the 'search' tab (looking for "Responses") and click you way through the Investigator hierarchy. Make sure you sort 'highest to lowest'. You can easily inspect 100 agents in about 30 minutes. What you are looking for is anyone who is exceeding 5000 invocations (Responses per Interval) in 15 seconds.

Of course, make sure your view is 'pinned' to 15 seconds.

Here is an example of BAD:

Here are the details of the offending methods:

```
e002            LMIG|AMIT|Struts|ClassInfoAction$1|compare  Responses Per Interval    255830.0

AMprde002B      LMIG|AMIT|Struts|ClassInfoAction$1|compare  Responses Per Interval    214460.0

AMprde002C      LMIG|AMIT|Struts|ClassInfoAction$1|compare  Responses Per Interval    146024.0
```

What do we see? First, it is customer tracing (self-inflected). Second, it is probably the result of a wildcard in the custom tracer. Third, this EM is performing queries quite effectively (no one else is using it!) as I am able to view 15 second data for a 2 hour period – nice!

Sometimes you will not be able to generate queries efficiently because the EM is already seriously overloaded. In that case it is likely you already have enough insights from the EM screenshots alone. If not, here is another technique to uncover excessive instrumentation:

This is the, always handy, transaction trace. As the EM is today sampling transactions then it is easy to go through what is already available. This is, of course, subject to an appropriate configuration for preserving the traces. You only need 5-7 levels of stack to do effective triage. Anything more is 'excessive' even if it does not violate the 15 second rule for the number of responses (invocations).

Here is an example of UNDER-INSTRUMENTED:

None of this should be a surprise to the client, once they acknowledge that they never looked at it or understood what they should be looking for.

Consistency of Agent Naming

Now that you've got their attention, we can start with the nit-picking. Agent configurations go wild when there is no single authority for what is expected and how to get it done. A runbook and preproduction review will keep this in check. The simplest way to track progress is to look at how consistent agent names are.

Here is an example of INCONSISTENT:

It's pretty simple. Stick the app server version into the agent name. Now everybody knows.

Agents with Excessive Metrics

When the APM environment is functioning reasonably, and you've got less that 150 agents (the workstation metric clamp limit) then looking at the supportability metrics is the most direct way to identify the problematic agents.

Here is an example of BAD:
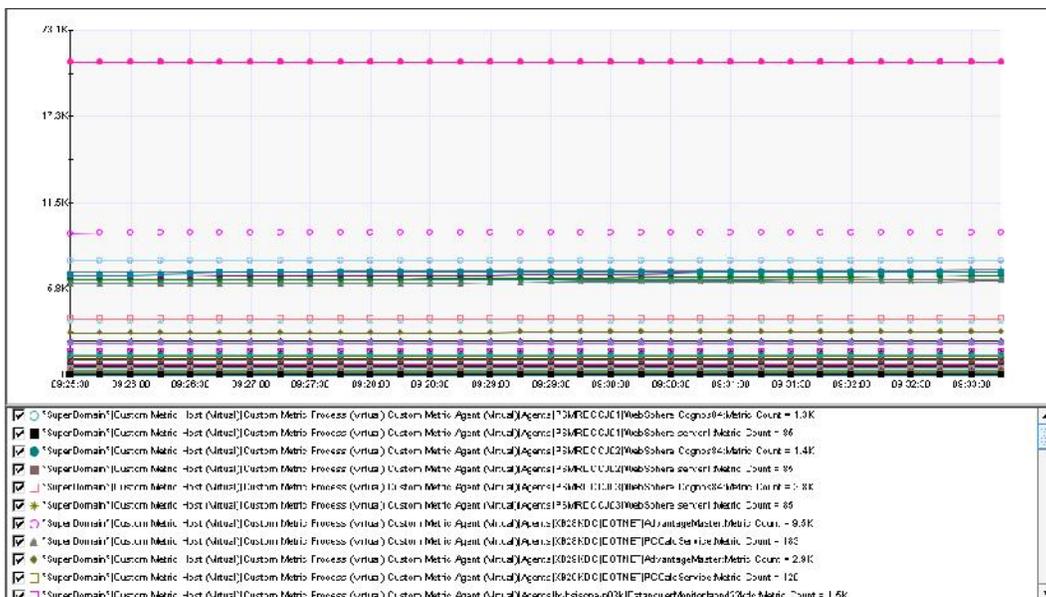
This is a live view, locked to 15 seconds. Mouse of the offenders and right-click to copy the agent name, and stick into the notes for this screenshot, after you have pasted it to PowerPoint.

This next category of trouble breaks into two parts: metric leaks and metric explosions. These configuration problems can result in an unstable EM even when you are well within the capacity guidelines.

METRIC LEAKS

Metric leaks are due to unrestrained creation of unique metric names. It happens most often with backend metrics, such as SQL statements but can also be a problem with JSP metrics. The JSP is the first design pattern for developers when they move as legacy app to the web. And it almost always goes poorly. The variety of SQL statements also reflects characteristics of a legacy migration but in this case, we do have the *SQL Normalization* to correct it.

Here is an example of a metrics leak, indicated with the red circle:
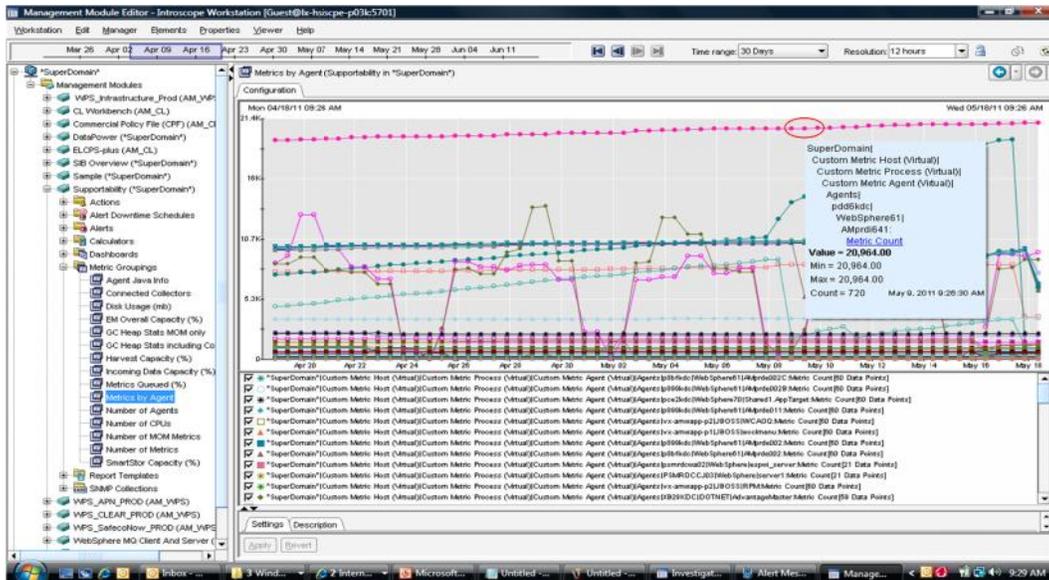
A nice, slow burn - apparently forever. This is not terrible but if you can find these inefficiencies, then you can correct them. It just takes commitment to the process of tuning the monitoring configuration.

METRIC EXPLOSIONS

When things leak really, really fast, or have otherwise suddenly changed their growth characteristic, then the leak becomes an 'explosion'.

Here is an example, indicated by the red circle:
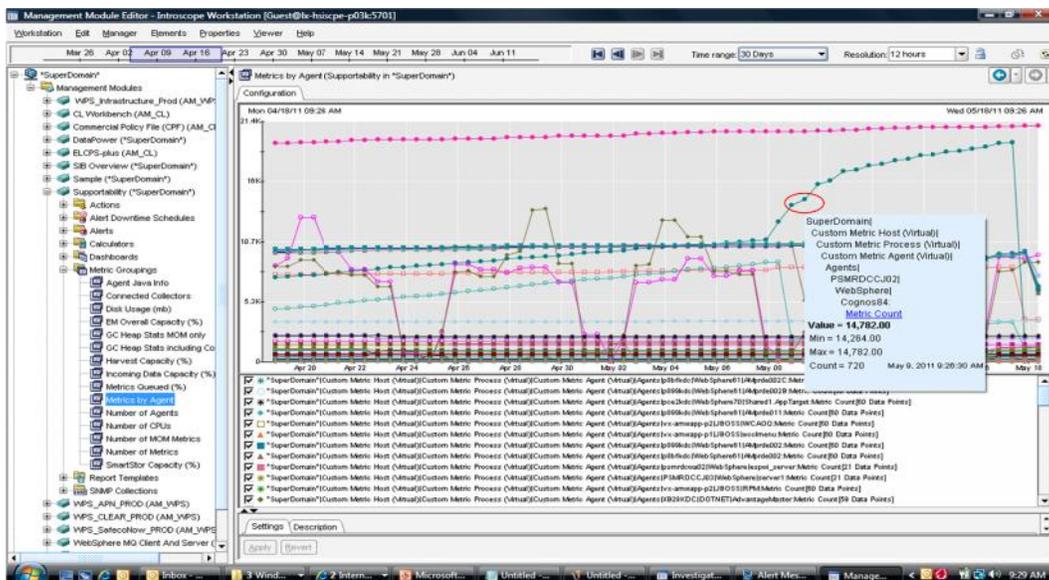
What starts out as a leak, for about 15 days, suddenly goes ballistic, over the next couple days and results in the server crashing or being restarted. Usually metric explosions are over a shorter interval but once you lose linear growth (straight line) – then I say it's a 'boom'! Which metrics contribute to the explosion? What use case triggered the change in leaking? These are all interesting questions and an excellent exercise for the monitoring team to start to show that they can actually use the APM technology. Try very hard to get the APM team to do the root-cause analysis because this is what they should have been doing all along. Your job is to deliver the HealthCheck. Plunging down this potential rat-hole means that you will likely run out of time before completing the HealthCheck. Other findings may cause you to make some recommendations requiring a service engagement. If you start here, you won't finish and the client gets even more frustrated. And it may be beyond your abilities. Just say "no".

Types of Metrics

Here is an example of socket tracing that is wasting metrics capacity:
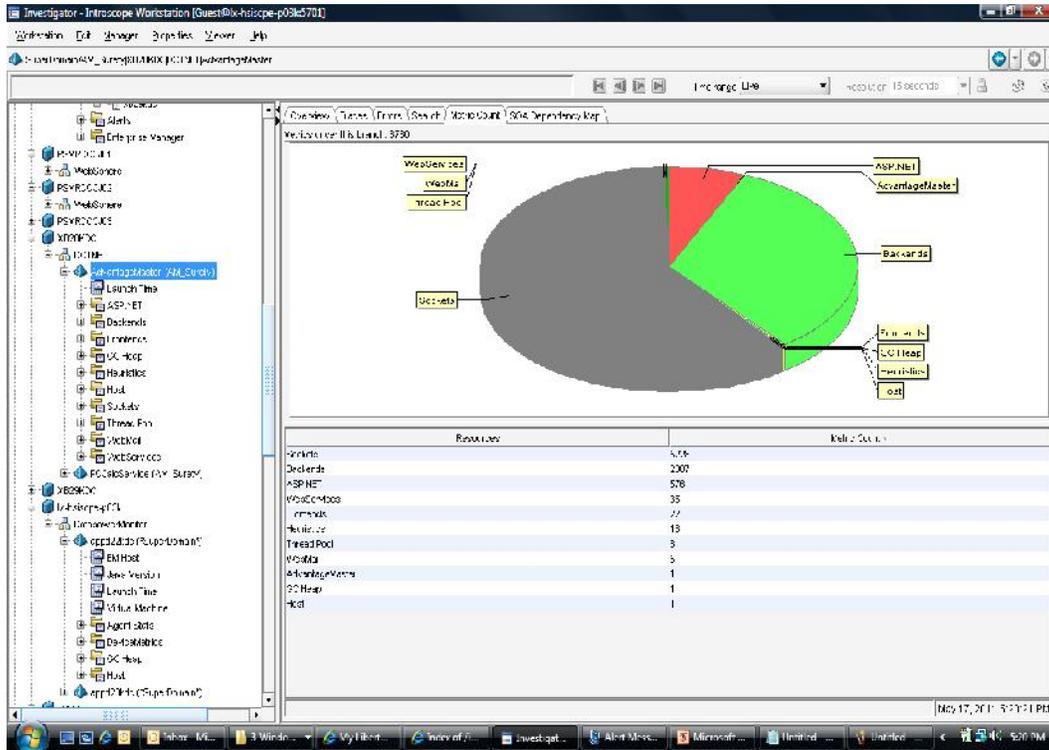
Sockets are just not very useful for triage. Nice to see during QA testing and then you need to turn them off – and 5000 metrics eliminated. In general, 1500 metrics are sufficient for reliable triage. You just need to make sure they are the right ones. That's what the *Baseline Process* is for – know before you go (to production).

Here is an example of REASONABLE:
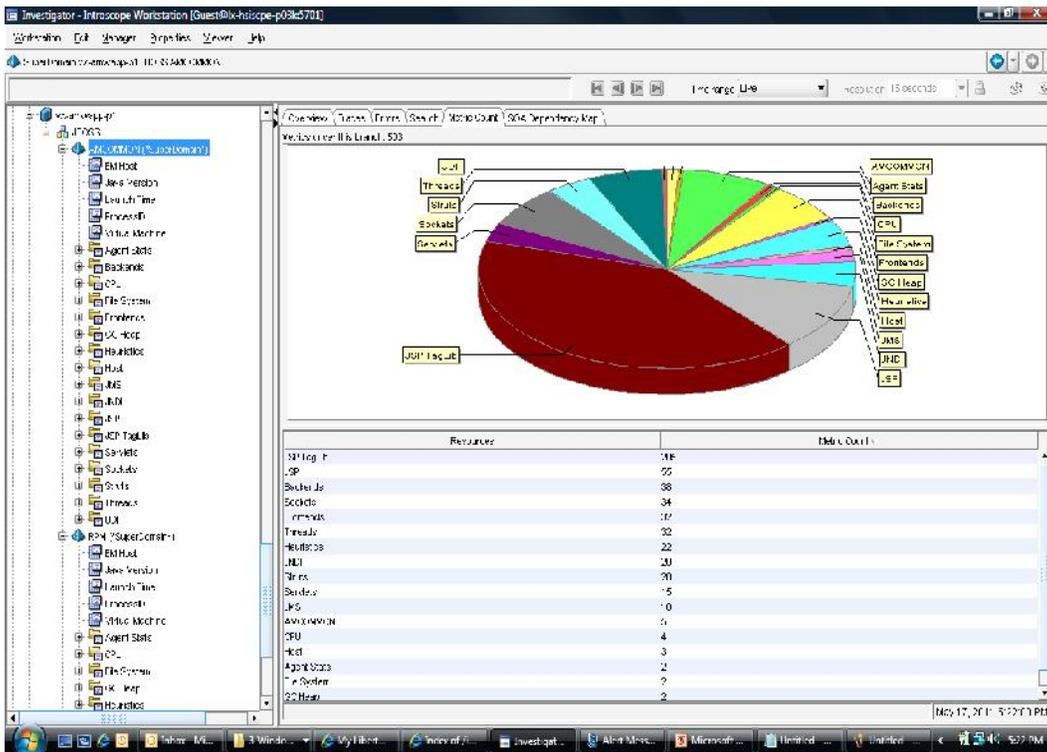
Here we have some 500 metrics, mostly JSPTagLib, which are not terribly useful. You would want to tune this configuration as well but it is a much lower priority.

Here is an example of excessive custom tracing:
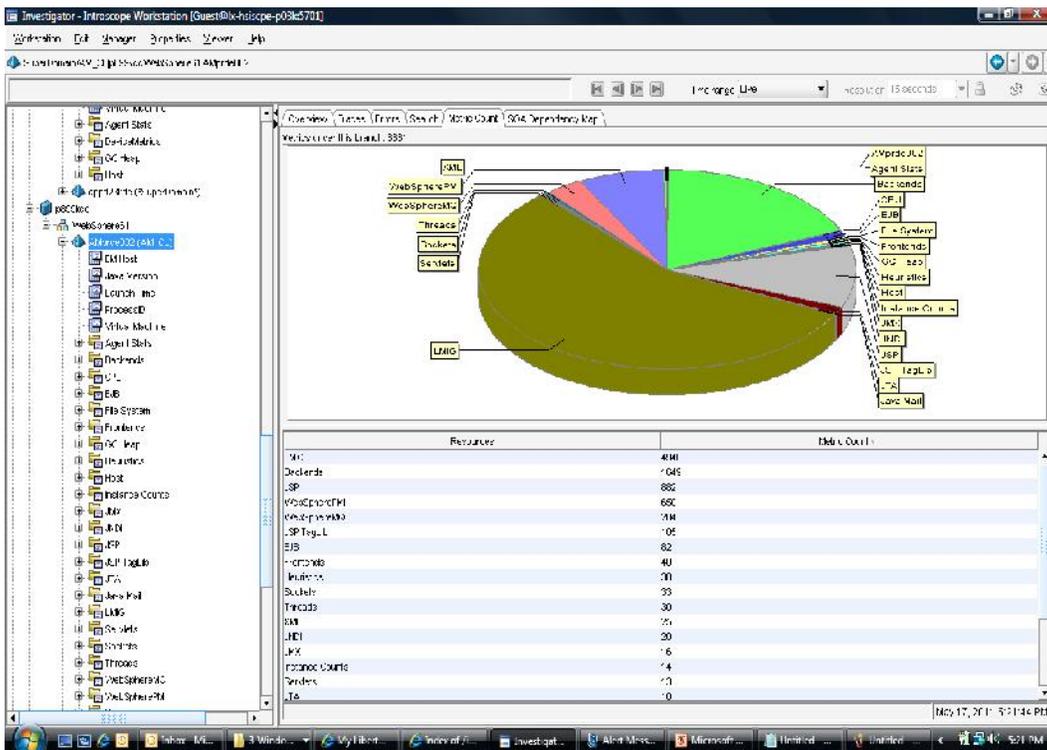
This is easy to see, provided that they have followed convention and named the custom metrics so that they all come under one folder. But having 4900 metrics being all useful – not likely. Tuning is needed here and the baseline process will at least show you which ones matter and which ones are not useful. If you are doing this is QA – that's cool. Just make sure you have a process to determine which metrics goes forward and be prepared to show a Production Review Board why this configuration will be effective (the next time).

Here is an example of excessive PMI metrics:
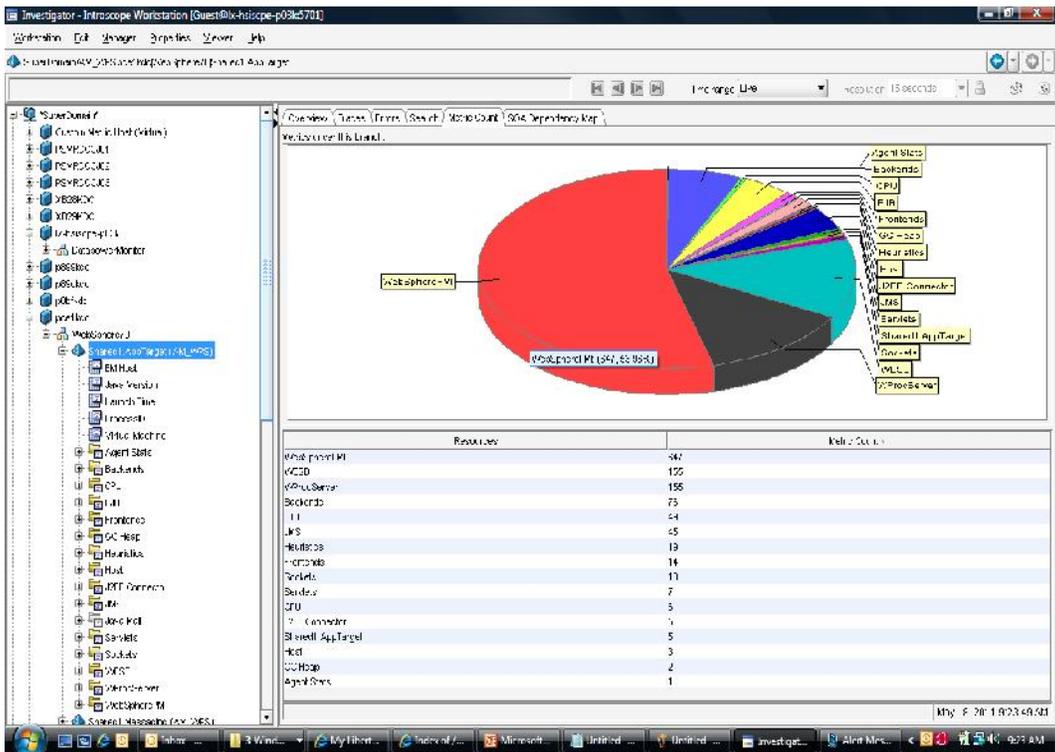
Only 650 metrics are generated here so it is not really a priority. PMI is a deprecated interface. It's has shocking overhead and really limited utility for problem solving. OK for QA, where it helps in tuning the app server. Thereafter, just turn it off.

Trace Quality (Application Baseline)

Evaluating if you have sufficient visibility is what the Application Baseline is for. See the full details in Chapter 12. As long as you have 60% of the bottom covered with a component trace, you are in good shape.

Here is an example of GOOD visibility:
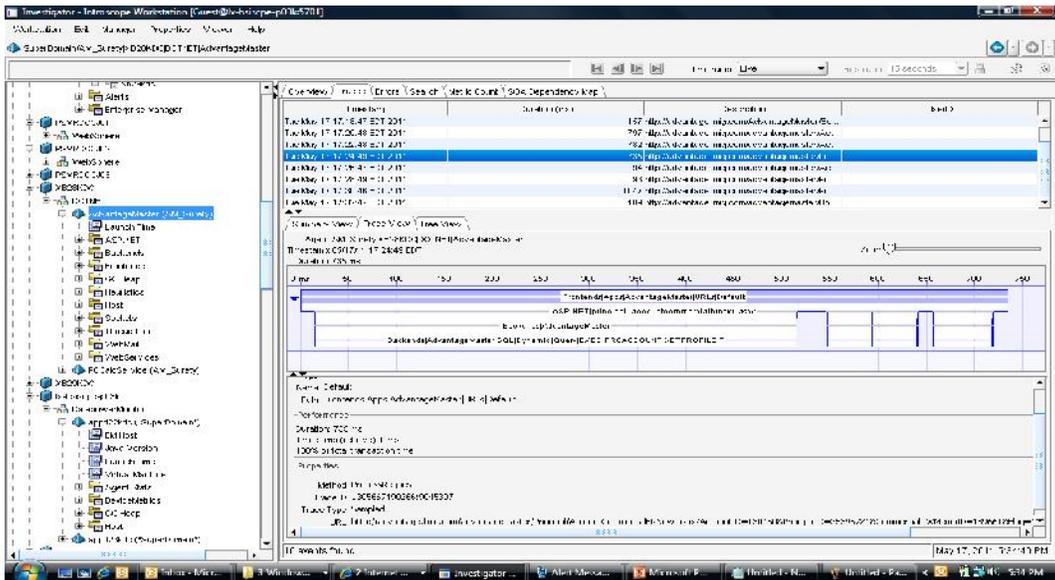
There are a couple of gaps but in general, the longer acting components are the best prospects for really compromising this transaction overall.

Here is an example of POOR visibility:
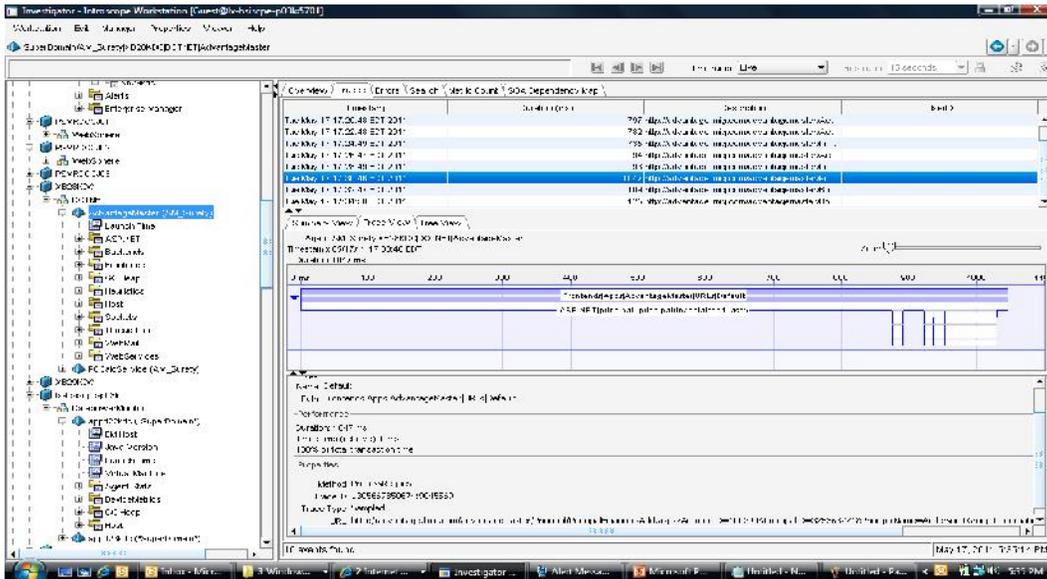
Here we have about 10-15% of the bottom covered. Any problems are likely going to be in the stuff we don't have instrumented.

EM Load Testing

Another form of "healthcheck" is when a large physical environment is being proposed – and the client desires to validate the following:

- Will the hardware support the target number of metrics?

- Is all that hardware actually necessary?

- Will the architecture support SAN, virtualization, large numbers of users, etc.

The details for this activity may be found in the Best Practice Module "LCMM-1 – EM Capacity testing.ppt". This is an advanced topic and is not a trivial undertaking but your HealthCheck skills will be used to evaluate the data coming out of the cluster, just the same as would be for a 'live' installation. The focus however will be limited to the MOM and Collectors, as the agents are simulated.

You should also expect to do a bit more reporting through Introscope, in order to directly compare different collector configurations, as when you compare physical versus virtual, for example.

REPORTING

Screenshots

Screenshots are always more efficient then writing down enough details to actually reproduce the view you are currently interested in. It also makes it completely simple to annotate and later communicate and mentor the APM team. It also makes it easy to get help, via the WIU, via email. Everybody loves a puzzle and a picture says a lot – compared to your pressured written description of the problem.   The APM workstation today also presents a couple of features to copy just the graphs and this will make for a neater presentation or report.

Spreadsheets

When reviewing a large cluster, or numbers of clusters, it is often useful to summarize the EM characteristics. This allows you to focus on the exceptions – collectors that are not at all the same as the other collectors. As we know from EM Performance and Sizing, the cluster will perform as fast as the *slowest* collector. So it really helps to know which collector is compromising the overall cluster performance.

QA

	# Metrics	# Historical	Durations	# Agents
MOM	3000	45000	190	
col1	175000	1355000	1300	115
col2	180000	1440000	900	90
col3	192000	1500000	940	75
col4	165000	1700000	900	125
	715000			
Target	1880000			
Capacity	38%		steady	

Earlier TIBCO config caused a lot of problems
Recommend to drop SmartStor (archive) and start fresh.
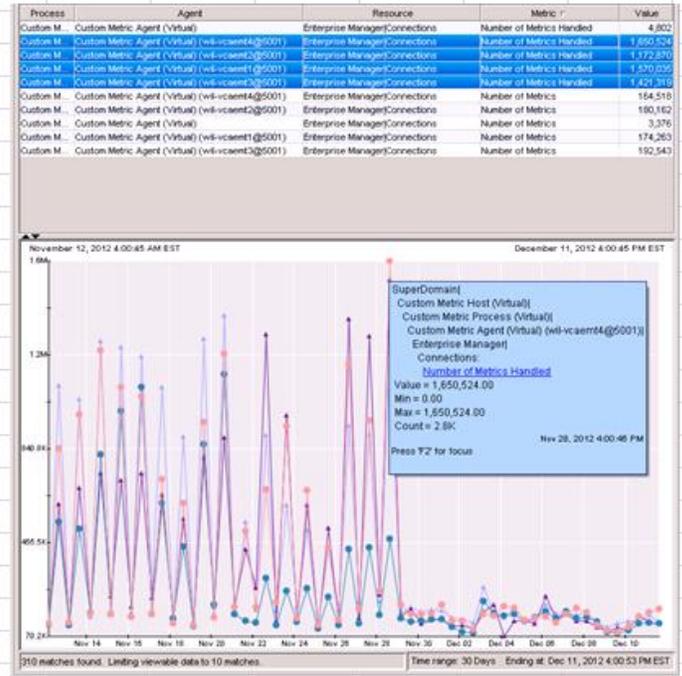
You will not need an in-depth analysis - just enough to put things in perspective. In this case, col1 is a little out of balance. You can also throw in a screenshot to help you appreciate how things got into trouble.

Introscope Reports

For deeper analysis and comparison of events, the Introscope report is preferred. Here you can collect numbers in a tabular form and then use spreadsheets or scripting to make the analysis. This is absolutely necessary when conducting the EM Load Testing (Solution Certification) as we are actually planning to do testing and will take the time to set up the appropriate metrics groups and automate the report generation. For most clients, the significant gap is any structure around testing and the use of reporting in general. They are so sold on the use of dashboards that they set-aside the fundamentals of basic, reliable reporting.

The real gap is that they do not know what to report on. Now, with the *APM Best Practices*, there isn't an excuse anymore. They follow the baseline process and they end up with a manageable set of metrics and some basic understanding of the application performance characteristics. They now know what to trend and compare.

Wrap-up Presentation

You will want to cover the following topics during your wrap-up:

Slide 1 – What was done

- HealthCheck of EM
- Which environments (for which EM is deployed)
- Which EM servers
- Total time spend on-site
- Any impediments

Slide 2 – APM Overview

Looks like this:

APM Overview

EM	Metrics	Agents	Metric Groups	% Excessive Configurations	Dashboard Complexity and Utility, Reporting	Overall Rating
5701	147,492 •Low growth •Some decrease	65	546 •Static	30% •Metric leaks •Metric explosions •High overhead instrumentation	•Linked dashboards with thresholds •Some reporting	D-
5601	103,439 •Growing	32	129 •Static	18% •Metric leaks	•Inconsistent use •No linking •Many components inactive •No reporting	D

This is for 2 stand-alone Ems. I tend to grade rather harshly but consider the following grade scale:

A: Best EM I have ever seen. Well-balance for performance and about _____ agents capacity remaining.

- Full incident reporting

- Pre-production Review of agent configurations

- QA Tunes agent configurations prior to production

B: Pretty good. Well-balanced performance and about _____ agents capacity remaining.

- Adequate incident reporting

- Aware of agents with heavy configurations

- No agents with dangerous configurations

C: No major incidents. Within capacity but opportunities for scalability look poor.

D: No major incidents but numerous abuses and few people actually using the data (production lockdown).

F: Lots of incidents but no tracking. Manifold abuses and users are abandoning the platform.

Slide 3: Concerns

Looks like this:

Concerns

- Excessive instrumentation (5601, 5701)
- Excessive use of PMI
  - Little diagnostic value, adds overhead
- High overhead due to incorrect instrumentation (dotNet 5701)
- Incomplete visibility (5601, 5701)
- Ineffective Dashboards (5601)
- Limited use of reporting (5701 is a little better)

Just try and get all of your concerns on one slide.

Slide 4 : Recommendations

A list of what you recommend they should do to correct the problems.

Slide 5: Details

A summary of what you observed, that lead to each of your recommendations.

Written Report and Recommendations

Within reason, please do not recommend ANYTHING that you cannot support with a graph, screenshot, page number or spreadsheet. Written docs can come back to haunt you. If you are not sure of something, then say exactly that. If your source data is suspect or incomplete – tell them and then go ahead and do your best with it.

You will want to cover the following topics in your written report:

Summary Points

APM Environment Health Review

Covered in this CookBook

Gaps and Risks

Organizational Maturity

Covered in the Skills Assessment Cookbook

Recommendations to Ensure APM Initiative Success

(optional) Detailed Recommendations

Covered in presentations and documents around APM and On-boarding Use Cases. Basically a 1-2 slide outline of what they specifically need to undertake, to resolve a performance or adoption problem.

For a scrubbed example of a final report, look to the Appendix.

FOLLOW-ON ACTIVITIES

After a HealthCheck has been completed and accepted, there are a variety of activities a client will want to pursue.

The obvious starting point is a **Services Proposal** to remediate the current environment. Maybe it is just a staff augmentation to help with a version migration. Or it could be an opportunity for custom tracing or other advanced or complicated configurations or functionality.

When you have uncovered that there are a number of short-comings with how the technology is being employed, you may want to schedule a formal **Skills Assessment**, to further explain the best practices and document the specific gaps and plan the client may want to pursue. Look to the "Cookbook – Skills Assessment.pdf" for guidance.

If the client is interested in improving their use of APM, especially towards testing and dashboards, then an **Application Audit** is indicated. An app audit will use load testing to characterize an application, establish the three baselines and help identify the KPIs (Key Performance Indicators) for the application. This in turn identifies which metrics to threshold, assign alerts and assemble a set of dashboards, as well as prepare report templates – really everything the client should be able to do, for a given application, as needed. With the App Audit, we do it for them and we use that exercise as a mentoring activity. The ideal case is where they have a couple of applications. We do the first one and then we manage their efforts in doing the next couple – but they are doing the bulk of the work. This ensures reliable skills transfer. Look to the "Cookbook – Application Audit" for full details.

When there is sufficient capacity remaining, the client may be interested to undertake a deployment planning and solution sizing. The first question is always how to find the next set of candidate applications. We start with an **Application Survey** (see Chapter 3 in APM best practices) to identify candidates, prepare an estimate with a Solution Sizing (see Chapter 10 in APM best practices), and then prepare a **Phased Deployment Plan** (see Chapter 7 in APM best practices).

REFERENCES

Artifacts

Application_Survey_May_2010.xls

EM_Performance_Assessment_Guide.xls

GENERIC_EM_Sizing_Forecast.doc

Sizing

Apmdiskspacecalculator.xls

EM - Application_Sizing_ALL.xls

EM_load_testing.zip

GENERIC_EM-ProdCapacityFindgs.doc

GENERIC_LARGE_EM_Sizing_Forecast.doc

Presentations

APM Best Practices - Sizing Overview.ppt

PDFs

APM9 Network Architecture.pdf

apmsizingperformanceguide_9000.pdf

HealthCheck – EM.pdf (Introscope report example)

Best Practice Modules

LCMM-1-EM Sizing and Capacity Forecast.ppt

Books

APM Best Practices – Realizing Application Performance Management   ISBN 978-1-4302-3141-7

ABOUT THE AUTHOR

Michael Sydor is an Engineering Services Architect specializing in Best Practices for APM. He advises and leads client teams to establish their own APM disciplines to deliver effective triage and manage performance across the application lifecycle. Michael is also the author of "Application Performance Management – Realizing APM", available from APress and Amazon.