

JENNIFER Tech Note

Enterprise Application Performance Monitoring with JENNIFER

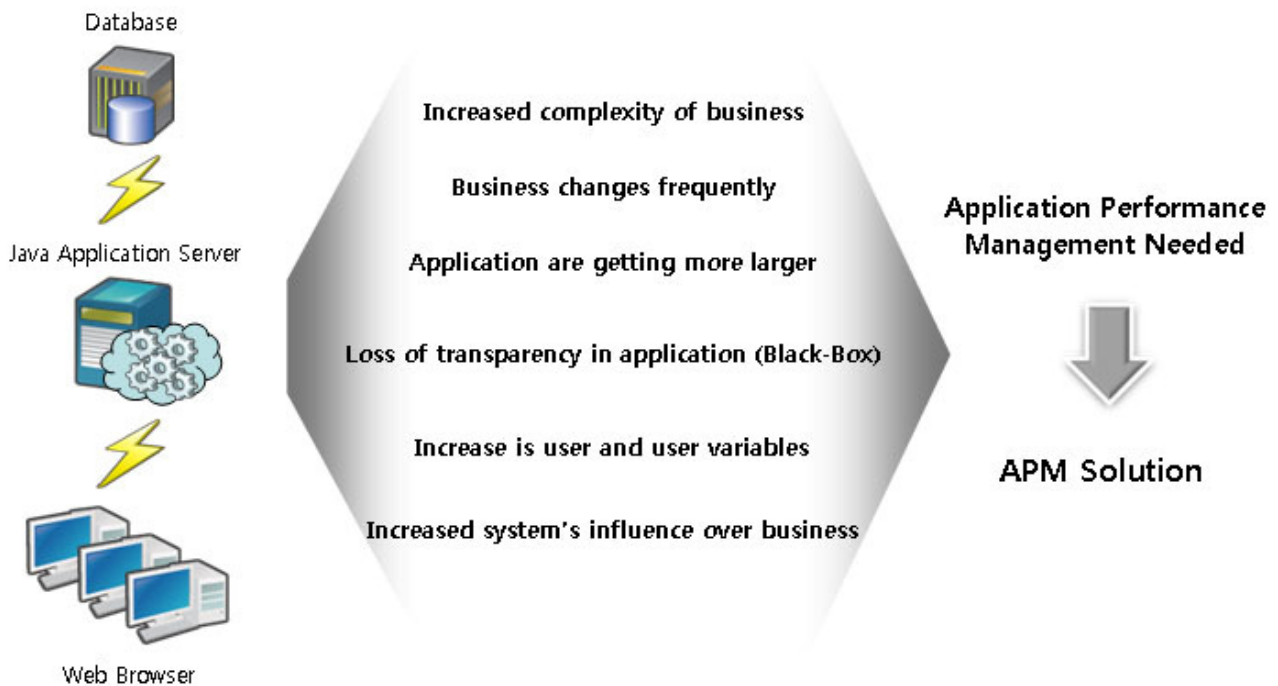
Date: 2009-03-30

Autor: S.J. Kim

In this article, I would like to discuss about the Enterprise Application Performance Monitoring (EAPM) process and how JENNIFER is involved in the EAPM process.

Today's enterprise applications often use internet as its medium to provide valuable services to the end-users. Using internet as its medium provides applications the advantage to offer services all types of end-users around the world by allowing the end-user to use their own personal computing system to connect to the applications via web browsers. But using internet opens up the certain possibility of constant system errors and performance problems for the applications.

Today's web based applications are getting bigger and more complex to match the increasing complexity of the business. Customers using their own PC result in increase in user numbers and user variable, requiring scalability to be a big concern for today's application. Also, as more and more customers rely on internet to receive their service, web based application hold significant impact on the business, making its stable operation a top priority for many companies.

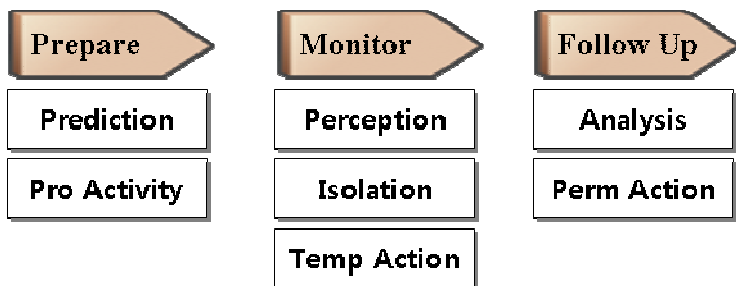


As enterprise applications play more significant role in the business environment, how applications are managed and maintained also must be improved. A proactive approach to managing the application performance and resolving errors and exceptions in real-time is necessary for today's IT team to ensure stable system operations and customer satisfaction.

EAPM process provides a guideline on how to IT team show manages enterprise application. This guide provides a quick look at how a system administrator or application performance engineer should identify, diagnose, and resolve application performance in today’s enterprise application environment.

1 Enterprise Application Performance Monitoring

The Overview of EAPM process is illustrated in the below diagram.



PREPARE

The cycle of EAPM typically starts with expectation of an application performance problem by system administrator or application performance engineer. System administrator may expect enterprise application to experience performance problem when a new application is deployed or when a business event which may influence the application is scheduled to start. Seasonal sales event in an online store or beginning of new semester for school registration system are good examples of such business events. Expecting application performance problem originating from business event or change in enterprise system is called “PREDICTION”.

Let’s take a look at a busy online store during the beginning of Christmas season as an example. On Monday, popular online toy store “X” announces that they will be starting a big sales event to attract customers for the Christmas shopping season. The sales promotion may attract a flood of shopper far beyond the number that their enterprise system can handle during off-season times. Increase in customers opens up the possibility of system performance problems, and at worst, system down due to overwhelming service requests. An experienced system administrator must be able to recognize the events which may result in system performance problems and make the necessary preparation to start EAPM process.

Remember events associated with PREDICTION may not always be as obvious as the above example. To an experienced system administrator, an unexplained increase in memory usage or unknown error messages can be used to predict potential performance problem.

When a performance problem is predicted, system administrator starts to perform a variety of activities to ensure that the enterprise system is protected and expected performance problems do not cause damages to the business. Some of the example preparatory steps are temporarily expanding hardware resources, re-check the parameters of the application environments, backing up unnecessary log files to free space in servers, and installing a monitoring solution. System administrator may also set up modules to block or limit excesses incoming services requests to prevent system from crashing under it. Also, in order to identify the root cause of the application performance problem, administrator may prepare some shell scripts that takes dump of system status in log files. These activities that an administrator may perform to prepare for expected performance problem is called PRO-ACTIVITY.

PRO-ACTIVITY is an important step because doing a good job in this phase protects the business from application performance problem in the first place, and it also determines how fast the performance problem is identified and resolved if the protective measures have failed to prevent business from taking damage.

1.1 MONITOR

Once the preparation for expected performance problem is finished, MONITOR phase comes next. On the day of the application performance problem is expected to occur, the system administrator starts to monitor the applications and enterprise system for any sign of performance problem. Administrator may keep their eye on the system resources usage such as CPU utilization or he/she may scan the alert panel for any application errors detected. Administrator is ready to react at any sign of performance problem; this step is what is called PERCEPTION.

When a performance problem is perceived, system administrator starts investigate whether the performance problem is real and narrow down at what level of system the performance problem is originating. This step is called ISOLATION. Typically, one cannot find the exact cause of the performance problem during ISOLATION stage, but narrowing down to the general area is the goal. For example, if performance problem is perceived to be originating from the Application Server, Thread Dumps are immediately logged. If the performance problem is perceived to be in the database, the DB administrator is notified so that additional information can be collected. During the ISOLATION phase, collecting sufficient logs and performance data is the key since the diagnosis and resolution for the performance problem can be accomplished only by analyzing the data collected during this phase.

Often, data that are used to identify and resolve performance problem are collected while the performance problem is affecting the system. This means that while performance problems are being recorded in order to create the solution, the business may be being affected by the performance problem. An experience system administrator may perform activities to temporary alleviate the effect the performance problem has on the business. For example, if the throughput suddenly increases, system administrator may limit the number of incoming request and redirect the requests to an alternate webpage so that the applications and servers are not overloaded. The actions that are done to temporarily patch the performance problems while an actual solution can be created are called TEMP-ACTION.

1.2 FOLLOW UP

When a performance problem occurs, that day becomes the longest day for the administrator. He/she must generate all kinds of reports for root-cause of the performance problem and its effects on the business. System administrator must also produce plans to address the performance problem effectively. Some performance problem can be very difficult to resolve and may require assistance from external consultants. The activities that are involved with identifying cause of performance problem and creating the resolution for performance problem are called FOLWUP.

Analyzing performance problem is

2 EAPM-Process and JENNIFER

In this chapter, we will discuss how JENNIFER is used during each stages of the EAPM process. JENNIFER offers features which help system administrator in each stages of EAPM process. While JENNIFER does not address all issues revolving application performance problem, it helps system administrator by offering series of tools and features which can help facilitate through the EAPM process by resolving performance problems quickly and minimize its effect on the business.

21. PREDICTION

During the prediction phase, JENNIFER offers variety of different tools to identify up upcoming performance problems. The tools that are used during the PREDICTION phase is as followed.

Response Time Scatter Graph

Response Time Scatter Graph is especially intuitive and helpful tool in predicting performance problems. In a normal application environment, the dots which represent individual services

transaction should all pool at the bottom of the graph (approximately under 2 seconds). If administrators see some of the dots to start moving up the graph then starts forming a certain type of patterns, that can be a prediction indicator that performance problem is at the horizon.

JDBC Object Checker

Memory Leak Alert

Memory Leak Alert monitors for change in heap memory usage. If the heap memory usage is progressively increasing over few or more days, administrator would predict a memory leak problem. JDBC Object Checker monitors whether JDBC object has been unreturned to the pool after its use for a long time. JENNIFER can create customized alerts so that if the heap memory usage or JDBC connection meets or exceeds the threshold, admin is alerted appropriately.

2.1 PRO-ACTIVITY

Once a performance problem is predicted, the user must start to prepare to impede the performance problem first, and then ultimately create a fix for it. Thus during the PRO-ACTIVITY stage, it is important to device preventative measure against damage to the business as well as collection of system data to analysis and diagnosis of performance problem. Taking thread dumps and heap dumps should be taken and other system snapshots and logs should be collected. Because the data used in analysis can be only collected while the system is experiencing performance issue, administrator must do the due diligence to get ready for data collection at appropriate time.

Following features are available through JENNIFER for use during the PRO-ACTIVITY Stage.

Adjust Profile Level

Exclude methods that is unnecessary to monitor from profiling to prevent unnecessary overhead (ex. Simple utility methods)

Limit the number of methods to be profiled from any one specific transaction.

Set threshold for whether the profile data is collected from a transaction. (ex if transaction response time is below 100ms, profile data is not collected)

Re-check the parameter of transaction when the application is changed.

In production environment, profiling configuration should be minimized. When a new application is deployed or application is changed, the configuration should be check to see if they fit the change in environment.

Peak Load Control (PLC) to limit number of incoming service requests

Expand System Monitoring (REMON)

Check if there is any system resources that need to be monitored in addition to the application data and use Shell scripts and REMON to bring it in JENNIFER's dashboard.

Application Server Monitoring Expansion (ExtraAgent)

Out of Application Server Process internal resource, if there are any that need to be collected for monitoring, use ExtraAgent to monitoring data. Data that JMX is used to collect the data in included here.

Business Data Monitoring (REMON)

All data that is considered meaningful that are stored in the database should be monitored. For example, for a class registration system, number of total class registration or number of available slots would be useful if monitored in real-time.

Create New Dashboard

Every Use has different requirement for what want to monitored and see in the Dashboard. Use the above methods to create monitoring displays according to once specification then create a Dashboard that fits user's needs.

Set Alert Level

Set threshold for JENNIFER's Alerts including CPU utilization, max response time, etc... as needed.

Also, preparation should be made for Active monitoring. SMS, NMS, APM monitors the core part or standardized part of the application or system. However, administrator may often find himself in need of monitoring other information such as business data or various log files or data from NMON, VMSTAT, and other system utilities.

Administrator must be prepared for sudden influx of incoming requests. If the system crashes due to sudden increasing in service requests, the business impact of such event will be catastrophic and admin will suffer the consequences. Foreseeing the arrival rate of throughput and being prepared is a very important for any app/system administrator.

2.2 PERCEPTION

Once JENNIFER is placed to monitor the applications, any sign of performance problem is immediately recognized and communicated to the Administrator. An action can be applied to the performance problem only when the administrator correctly perceives the problem at hand thus the

monitoring solution must be able to clearly convey various performance and system data so that the Admin is provided with enough information to quickly understand the circumstance.

JENNIFER offers 4 different features for use during the PERCEPTION stage.

44+ error/exception alerts and additional custom alerts

Monitor main system resources, services and set custom thresholds limits and trigger alerts

Monitor overall status of Active Service status by Instances

Monitor for backend system status quickly using response time scatter graph (x-viewer)

JENNIFER provides 44 different alerts for the system errors or exceptions that is most commonly associated with system performance problem. 44 alerts are not very many compared to number of different errors in existence some monitoring solutions offer many more error/exception types, but JENNIFER purposely provides only 44 default alerts to prevent administrators from relying on alerts because relying on alerts cause administrators to monitor their environment reactively rather than proactively. Of course JENNIFER offers module for administrator to add more alerts as necessary. The threshold for default alerts and customer alerts can also be adjusted. Setting threshold is not an exact science and often required trial-and-error approach to get the alerts just right for efficient monitoring. With JENNIFER, administrator can adjust the threshold restart the application easily, thus alert setting is made much easier.

Alerts are great but it takes history data and trial-and-error to setup an effective alert, which means relying alerts to identify problem in new system is not possible. In new system environment, response time scatter graph and Active Service graph can be used to intuitively monitor system performance. Active Service monitoring by intuitively monitoring service status and associated resource usage, response time, and other critical information all at the same time.

Active Service Module and X-View is very effective in picking out abnormal application service transaction. When a service transaction is processed, if anyone of the system resource is lacking or has problem, it will cause a direct effect on the service response time. Thus, if individual service is monitored using a response time scatter graph, service performance can be easily an intuitively monitored and perceived by the administrator.

2.3 ISOLATION

Once it is determined that a performance problem exists, next administrator must start isolating the location of the root-cause. An important point that I would like to make is that all problem

concerning system resource unavailability results in increase in application service response time. Thus monitoring for service response time of each section of application is very helpful in isolating root-cause of performance problem.

When a performance problem is perceived, administrator must first identify where the problem has originated. All resource problems can be perceived by monitoring the changes in the service response time. Thus, by monitoring the response time by different parts of application service process, administrator can narrow down and isolate where the performance problem has originated.

Using system resource usage monitoring to diagnose performance problem is not practical since many performance problem does not result in noticeable change in system resource. Also, Resource usage monitoring is limited by the fact that the resource usage can never exceed 100%.

The relation between system resource and transaction is not 1:1 or 1:N but N:M. Thus analyzing one transaction does never resulting in identifying which system resource is having problem. It is more efficient to indentify transaction that is having delayed response time and analyze them for a common trait.

JENNIFER collects transaction and profiling information per individual transaction and displays them in graphical form to make this process easy.

Detailed data for Active Service and service in process.

Response Time Scatter Graph (individual transaction and profile data)

In ISOLATION stage, quickly narrow down where the performance problem is coming from. Is it JVM/AS, Application, DB, or Network? Throughput is too high overall?

TEMP-ACTION

Once the performance problem has been isolated to a specific area, apply a temporary fix in order to provide relief to the users then work toward a permanent solution. This will keep the user satisfaction level up and maintain the business value of application. The most common temp fix is to simply restart the application server process. But in a clustered environment, restarting certain server process may affect other functioning services like a falling domino, thus restarting is not advisable in that case. Thus devising ACTION PLANS for various common system problems in advance is the key. For example, being able to limit the incoming service request or quarantining applications that are having performance problems can be very helpful ACTION PLAN in certain situations. JENNIFER provides a way to limit incoming service request by forcefully rejecting overflowing service requests and diverting them to another location in order to prevent system crashes from excessive service requests. Administrator can limit the service request by thread or

network but using such criteria may not be appropriate in certain business situation. JENNIFER can limit service request by service name, thus allowing service requests for business critical applications and limiting non-critical applications.

Peak Load Control(PLC)

In TEMP-ACTION phase, temporarily stopping the problem is the main goal but in order to do so, collecting system data and analyzing them must be done first. APM gathers application performance data and most performance problems can be diagnosed and solve by analyzing data APM solution collects but a system has more data than what an APM solution can get. But in most cases, this data are not stored for future analysis store these data for future analysis can be very useful for resolving performance problem then thus should be backed-up in a database. JENNIFER stores performance data of wide variety in its own database for future use. This action is called Service Dump.

Service Dump

2.4 ANALYSIS

Once the immediate problem is put out from TEMP ACTION and system is stabilized, administrator will start to identify the root-cause of the performance problem and create a permanent fix for it.

The Analysis of performance problem is very similar to ISOLATION phase, but the difference is the objective. If ISOLATION is focused on resolving the immediate problem, ANALYSIS phase focuses on finding out the root cause of the performance problem. Time is of most importance in ISOLATION phase since the TEMP ACTION require immediate fixes, but accurate diagnosis is less important. ANALYSIS phase requires in-depth diagnosis so it can take more time.

Another characteristic of ANALYSIS phase is that it may involve outside consultants, thus reserving sufficient time and gather as much information possible before creating the final fix is important.

JENNIFER provides many different tools for statistical analysis of past transaction data. Backing up performance data for the date when the performance problem occurred and replaying it while analyzing the root cause if important.

Daily X-View Summary

Statistic Summary

Summarizing database performance data using SQL

Provide summary of in individual transaction performance data sample.

If resolving performance problem fails, the cycle of EAPM must be started from the beginning. Administrator must wait until the problem at hand occurs again, and then go through the cycle of EAPM from the beginning.

PERM-ACTION

Perm-Action is the phase when a permanent fix is created for a performance problem and it is applied to the target system to finally resolve the performance problem. Typically, Perm-Action can involve adjusting options, applying bug-patches, configuring programs, and increasing hardware resources.

Progressive system improvement must be performed by application manager for system administrator. Monitoring solutions can be used to check whether the final solution has effectively resolved the performance problem after the final fix has been applied.