The Guidebook to the SPE BoK

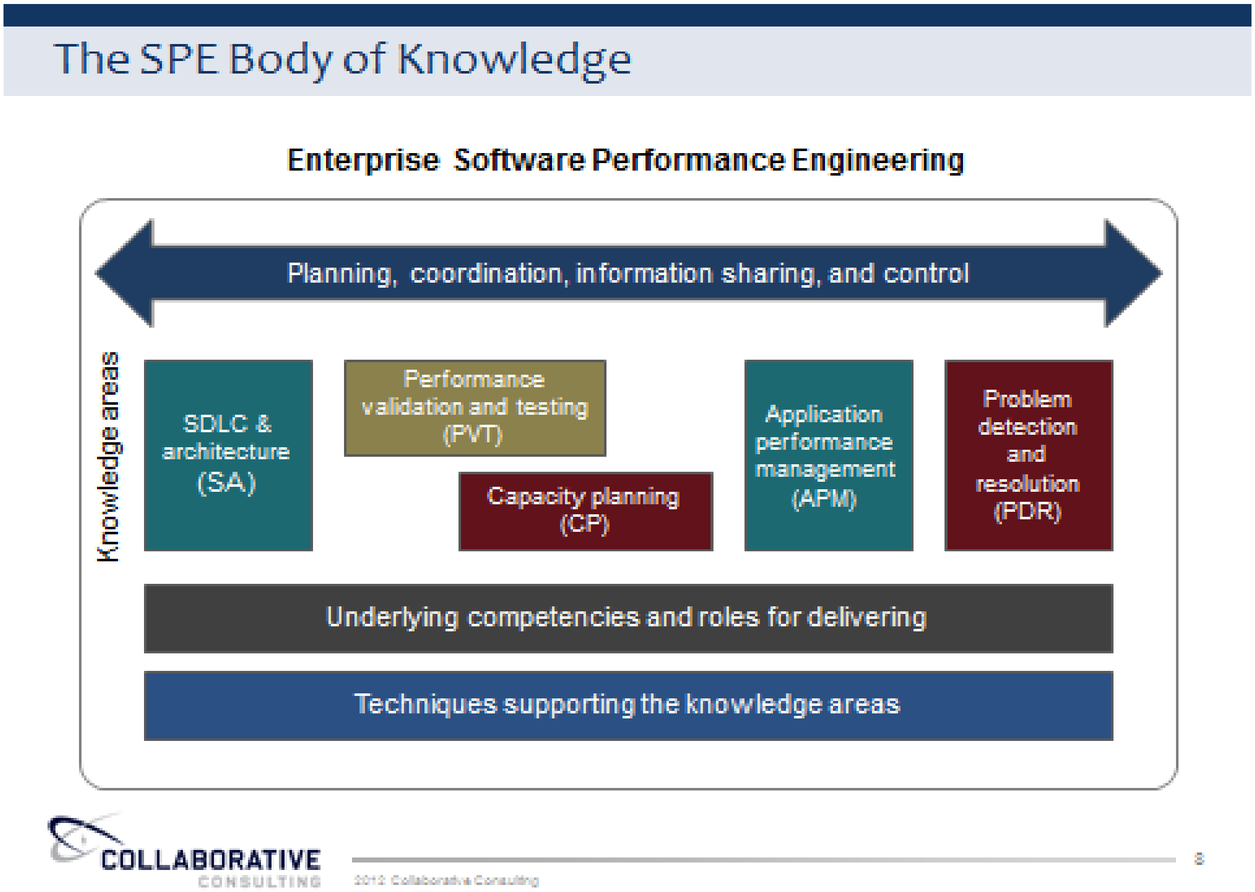
DRAFT - Software performance engineering body of knowledge

[Pick the date]
Collaborative Consulting, LLC
Walter Kuketz

The Guide to the Software Performance engineering body of knowledge

Table of Contents

Overview of the software performance engineering body of knowledge



The SPE Body of Knowledge

Enterprise Software Performance Engineering

Planning, coordination, information sharing, and control

Knowledge areas

SDLC & architecture (SA)

Performance validation and testing (PVT)

Capacity planning (CP)

Application performance management (APM)

Problem detection and resolution (PDR)

Underlying competencies and roles for delivering

Techniques supporting the knowledge areas

COLLABORATIVE CONSULTING

2012 Collaborative Consulting

8

Figure 1 - SPE BoK diagram

Introduction

1. What are the Industry trends

The response time of software systems is always a key concern of business people and IT people. The user experience can directly influence the revenue of the business. Your IT systems and business applications have become more exposed to your customer over the past many years. There are a few key industry trends that stress the need for an Enterprise approach to software performance engineering.

Focus on user experience

How do you measure the user experience? The mobile device is everywhere and your customers need an outstanding experience on you web site or mobile site to purchase items. The user is now untethered and can access your site from a high bandwidth location to a low bandwidth location. The teams designing the application must be aware of this. You need to communicate across many different groups within your company to make everyone aware of the impact the user experience has on profits.

Shifting applications to PaaS and a SaaS model and cloud

If you are shifting more of your applications and businesses functions to a PaaS, SaaS or cloud service, how do you enable SLA measurements to hold others accountable for performance of critical business transactions? What are the new rules or guidelines your development team need to use to design and develop software that will run on PaaS or a Cloud platform. These choices impact the entire software development life cycle, it impacts the operations team.

Wide-spread adoption of Virtual Systems

How much CPU does your application need and how does it consume them during a peak lead? Virtual systems allow a new level of sharing system resources and present a challenge to the capacity planning team. The results of performance testing can be confusing when mapping the virtual and the physical to the resource needs of the application. In production, the logical partition may be assigned a minimum number of logical CPU's, the may or may not equate to a full physical CPU. There can be several logical partitions sharing the same pool of Physical CPU's and each on can be allowed to borrow from the pool. On the surface, moving to a virtual environment can provide great flexibility to the operations team.

Highly distributed teams and systems

The further apart the teams, the longer it takes to share the right information. Development teams can be distributed across the US, and then across India or China. Now add the architecture teams, the business team, the performance testing team, the capacity planning team and the operations teams. The application performance goals and objectives have to be communicated across these teams. If the critical business transactions and performance goals are not spelled out from the beginning, they will get lost along the way to production.

No clear definition of a software performance engineer

There are many roles within the software performance engineering domain, a key role is that of the performance engineer. Many companies do not even have that listed in the Human Resources job roles. The ones that mention it are too focused on performance testing. There are other roles as well, a performance architect, a Sr. and Jr. performance engineering, a performance test developer, capacity planners, performance modeling, and technology specific performance engineers. Typically the career path is not even defined for becoming a Sr. Performance engineer.

The roles and responsibilities are not clearly understood by the business and even by Sr. IT management. Many times they equate performance testing with performance engineering. Performance testing is a key part of the overall performance engineering framework. In many organizations Capacity planning and performance engineering are loosely coupled, sharing little information. They must be tightly coupled to provide maximum value to the business. Compounding this is the separation of systems engineering, operations and application architecture.

The role of Capacity planner tends to have a more common understanding of the responsibilities. A performance test developer is not confused with a capacity planner. A performance architect is not mistaken for a capacity planner.

Adoption of agile methods

Businesses are adopting Agile methods for larger and larger software development projects. The projects are moving from department focus to Enterprise focus. The teams are getting larger and distributed. The Non-functional requirements of systems are becoming more stringent. The end user experience is critical.

How do we implement performance engineering and performance testing tasks and activities in the Agile methods? How do we prepare the Performance Engineer for this, is it part of his career development plan? The introduction of PE/PT cannot compromise the original intent of Agile, faster and partial delivery of software. The goal of Agile methods is to produce frequent working copies of the system, with incomplete features. Features are introduced for each release. Traditional performance testing occurs after all the features have been developed and the application is nearing production.

2. What is the need for the BoK

We can take these industry trends as the catalyst for introducing a Body of Knowledge. Using the framework of a body of knowledge will allow the profession to formalize and define a standard for language and terms, critical job descriptions with career paths, and a consistent method to communicate performance engineering value to the business. This can start in your company.

SPE activities not visibly aligned to business value

You must be able to demonstrate how SPE activities provide value to the business. In some cases it is easier; designing, coding, testing and monitoring activities can be aligned to show an increase in shopping cart conversions on the web site, due to stability and performance. The SPE can directly and positively impact the user experience.

Converting data from one system to a new system faster will result in business value. SPE can make sure the non-functional requirements are defined early, in the case of a large data conversion, this can make all the difference in the business case. If the business can convert the data in five days instead of 15, then there is real measureable business value. Having a goal of five days upfront is critical. Often times many people back into the conversion schedule, and that appears only to be an estimate. Of course, if it goes longer then the business case gets worse. The design activities and testing activities will align under this as well. Does it add value to plan and execute a number of pre-conversion performance tests to measure the throughput of the conversion?

Providing accurate and timely capacity forecasts offers great value to the business.

No clear definition of the SPE profession and roles

In most companies there is no formal title or job description of a performance engineer. Many people arrive at this point in their career on their own. They attained deep technical knowledge in two or more subject areas, they have great problem solving skills, and they understand the business goals. Their formal title may be architect, database architect, lead tech, etc. Many of the performance engineering roles are split across the IT organization. Another critical role is the Capacity Planner; where they monitor the current system, forecast the future needs and track the costs associated with the system.

Often times the performance engineer and the capacity planner work in different groups, they share many similar competencies and techniques. They track the same information, however the key split has been the production focus of the CP and the pre-production focus of the PE.

The performance test developer role tends to align with the Quality Assurance group or even be part of Center of Excellence team (these CoE's typically do not add much value, they know how to test, but do not know anything about your application), the capacity planner with production operations, performance architect with the development group, database performance in the database engineering team.

The most common cross role confusion is that of the performance engineer and the performance tester. There are Sr. testing people who are also Performance engineers. The performance engineer is a more Sr. technical architect who understands a large body of performance practices across the SDLC, performance testing, application performance management, troubleshooting, capacity planning, etc. They are also a leader and great communicator and they understand the business goals. They understand the connection between the SLA's, for example, the login transaction must be completed in 1.5 seconds for the 95th percentile under month end peak load conditions. The batch process must achieve 34 TPS in order to meet the SLA of one hour.

The terminology varies by industry and company

There is a large vocabulary for performance engineering terms; from the business terminology to the design and coding terminology, performance testing, and monitoring and measuring. Performance goals start with business terms; one business is concerned with Shopping cart conversions, another is concerned with throughput of a message hub that communicates sales information to the distributors and wholesales in real time. An insurance company is concerned with the number of policy quotes, throughput, and the time to get the pricing (response time).

How is a critical business transaction defined in your company, who decides that? Does everyone in your PE group have a common understanding of them? Is this used as the starting point for the capacity planning process?

There are the core competency technical terms;

- Physical and Virtual CPU utilization, CPU entitlements, shared pools, CPU run queue,
- Database wait events, full table scan, left outer joins, buffer gets,
- number of IO operations per seconds,

- transaction arrival rates, system throughput, component throughput
- response time, service time & queue time,
- heap size and garbage collection,
- queuing, capacity planning and forecasting, closed system, open system, standard deviation
- page load time, event correlation, etc.

A body of knowledge can help you capture these terms and share them across your Enterprise.

Introduce Agile techniques to SPE activities

Everyone is tackling this on their own, wouldn't it be nice to reach out to a peer group working on the SPE BoK to define and implement a standard approach for companies to start with? This can be applied to any new technology or methodology that rolls through the IT industry.

The performance team must approach the Agile project that same way the Development team does; People over process, multi-disciplined, learning and adapting. The performance engineer and the performance test developer must understand the performance test process extremely well. During an Agile Performance Sprint, the Performance team will have to adjust and adapt to meet the timeline. They will have to communicate performance issues to the development team so they can get the fixes into the next Sprint backlog.

Develop a Performance Theme for the project. This is a project wide message to let everyone know the project has a performance and scalability focus, and the risk associated with the project require PE tasks, activities, and people. This must convey to the Architect and the lead technical person, that best practices, tips and techniques for performance and scalability must be used for this project.

- Performance Stories – The performance tests for the project, the workload models, and the test types. How do you add performance needs to the Story?
- Performance backlog – The performance tests that need to be executed.
- Spike – How will the performance tests be executed in context of the Release schedule and Sprints? Will there be focused out of band set of performance tests?
- Definition of Done – Clearly defined exit criteria for the Performance tests.
- New role on the Agile team. A performance engineer for architecture reviews, code reviews, database reviews, and test planning.

Since the project has been set to a performance theme, the development team must use best practices for performance when developing code/services.

Addressing career path and organizational challenges

The goal of the BoK is to overcome the organizational challenges you and other companies have. The key task you must undertake is to understand the gaps in your organization for performance engineering, the plan out how to define the roadmap to implement the BoK.

3. What is the business value of the BoK

- It plays a key role in aligning the IT investment in SPE to the business goals
- Helps IT define performance test results in business terms
- Enables Service Level Agreements for internal and external systems
- Improves the professional development of key IT people
- Ensure proper organizational placement of the SPE team
- Introduces a common language of understanding performance engineering; its more than testing.

Overview of the framework

SA:SDLC Design and build for performance	**PTV: performance testing**	**CP: Capacity planning**	**APM: Application Performance management**	**PDR: Problem detect and resolution**

1. How does the BoK work as a whole

There are two components; the development of the BoK and the implementation of the BoK. The current development state of the BoK is in a draft version. However, you can still use the draft to begin to plan the implementation for your organization.

To achieve a world-class performance engineering and capacity team and to maximize the value the SPE BoK will provide, you need to align the following items;

1) CIO Sponsorship and HR support
2) Define the role Capacity planning and PE plays in the organization and how it delivers value
    a. Improving shopping cart conversions improves revenue
    b. Ensuring a large data conversion is done early or on time reduces costs
3) An Enterprise PE Director as the point person for delivering value
4) PE tasks and activities must be embedded in the SDLC methodology and operations
5) PE and capacity roles and responsibilities defined
6) PE and capacity best practices repository is established for each knowledge area

The point person, planning and coordination, Enterprise owner

Your IT organization must identify the person/role responsible for Enterprise software performance and capacity. This person has a critical role in driving the value of adopting PE practices and they have a key role in the future direction of the team. It is a multi-disciplined role, for companies that are now adopting the Enterprise PE approach, this person must be a self-motivator, great communicator and be in good standing with the business. By good standing, I mean respected. This leader has a great track record in the company. Sometimes, this person comes from outside the company, because the role is new. In this case, like other new people, they need to bring instant credibility to the role. This role calls for the non-technical skills and the technical skills. They must understand the business goals and how the team aligns to them.

More than likely your company has an Enterprise Architecture team, has an Enterprise Security team, Enterprise Network team and Program and Project management office. Where is Performance engineering and capacity planning team in all this?

Establish cross organization communication for PE (Sharing experience)

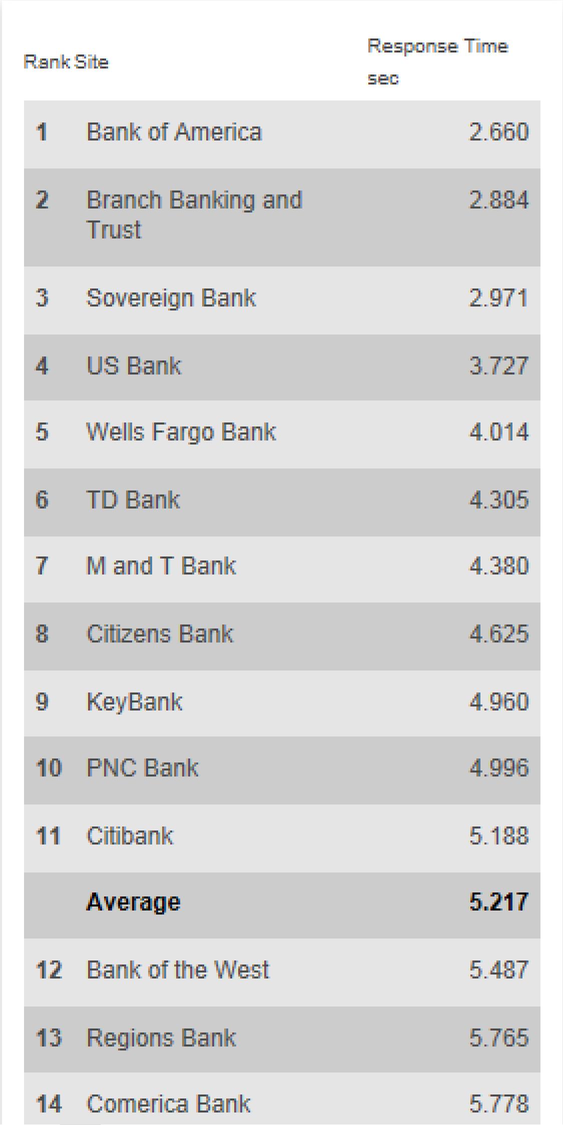
To implement the BoK, you must assess where the gaps are today in your organization, then create a roadmap to the desired future state based on the business priorities. The business priorities vary by company and industry. Performance is more important to some than others. You must know your businesses approach; some use a good-enough approach, or if it's not broken, don't fix it. So, you must know what you are up against.

- You must describe the knowledge areas to your team
- Identify the critical production issues and challenges; align them to the KA's
- Never let a crisis go to waste; if your company uses a new product that cannot scale, did they do a performance assessment first before they bought it?
- assess the current inventory of PE projects ; are there any POC's or testing?
- identify where the key people are; if this is new to your company, they will be scattered across the groups
- find where the existing knowledge is hiding (formal and informal),
- who has the rules of thumb,
- and where any guidelines for performance and capacity may exist.

Make PE tasks and activities visible to the Enterprise

This is where the performance and capacity team could use a good marketing person. Always make sure you communicate and publish a scorecard of successes. For instance, in some industries like Brokerage and Retail, an industry group measures the web site response time and publishes the top ten list of faster web sites, as the Enterprise performance engineer, you must make sure everyone is aware of the part your team played to get your company's web site into the top ten list. See the chart from Compuware;

Response Time



Rank	Site	Response Time sec
1	Bank of America	2.660
2	Branch Banking and Trust	2.884
3	Sovereign Bank	2.971
4	US Bank	3.727
5	Wells Fargo Bank	4.014
6	TD Bank	4.305
7	M and T Bank	4.380
8	Citizens Bank	4.625
9	KeyBank	4.960
10	PNC Bank	4.996
11	Citibank	5.188
	Average	**5.217**
12	Bank of the West	5.487
13	Regions Bank	5.765
14	Comerica Bank	5.778

Figure 2 - Web Site performance, Compuware

2. The Knowledge areas

SA:SDLC	**The software development lifecycle, design and building applications that exceed performance goals.**
PTV: Performance testing and validation	Defining, developing and executing a full suite of performance tests for periodic Release cycles, product validation and project level testing.
CP: Capacity planning	Bringing the right resources at the right time.

APM: Application performance management	Defining and monitoring the ensure application performance and the end-user experience, enabling faster root-cause analysis.
PDR: Problem detection and resolution	Systems still encounter problems, this defined the approach, tools and support roles needed to manage and triage issues.

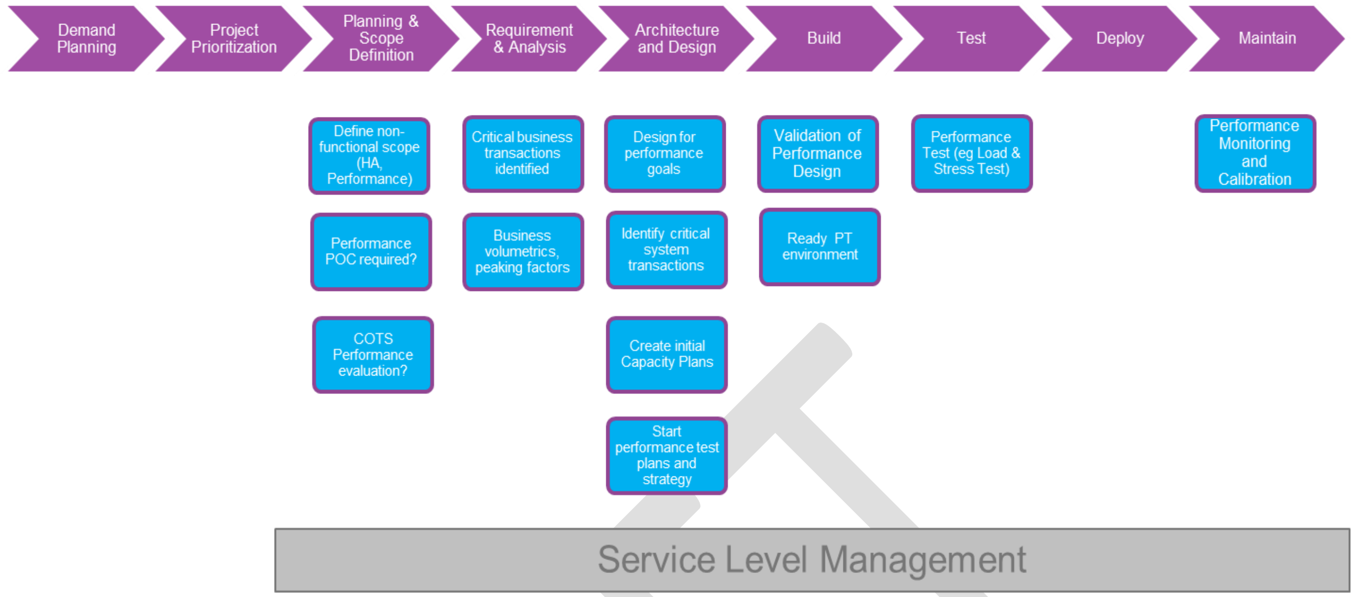
SA: SDLC Software development lifecycle

There a three types of SDLC, waterfall, iterative, and Agile. They all have one or more requirements phase. This is a critical phase for performance, the non-functional requirements are defined and setup for traceability throughout the project.

Some software Architectures enable performance better than others. There are many different types of application architectures; messaging, reporting, analytics, high volume online retail web sites, batch processing, etc. The Enterprise performance engineer and the Enterprise architecture team must work together to inventory and describe the architectures for performance. The Architect or the project Performance engineer can specify performance budgets for the critical business transactions. A performance budget is a time allocation for the transaction, you can communicate to the developer that they have 100 milliseconds to execute the transaction.

Many companies do not create custom applications, instead they buy a commercial package and work with the vendor to make customizations. This is where many performance failures occur, the customizations do not meet expectations. It is critical the Performance engineer work with the customization team to enforce performance and scalability design goals.

Performance is physical and in this KA's where performance is designed and built into the application. The development process is where the performance techniques are applied by the programmers and software engineers. They implement the algorithms of the application. There is a heavy dose of tools during the development process that can help build better code; static code profilers, and code analyzers, code profilers, and coding practices. The Enterprise performance engineer must work with the development teams to capture coding practices for performance. For instance; caching practices, use of abstractions, layering, parallelism, etc.


Demand Planning
Project Prioritization
Planning & Scope Definition
Requirement & Analysis
Architecture and Design
Build
Test
Deploy
Maintain
Define non-functional scope (HA, Performance)
Performance POC required?
COTS Performance evaluation?
Critical business transactions identified
Business volumetrics, peaking factors
Design for performance goals
Identify critical system transactions
Create initial Capacity Plans
Start performance test plans and strategy
Validation of Performance Design
Ready PT environment
Performance Test (eg Load & Stress Test)
Performance Monitoring and Calibration
Service Level Management

Performance testing and validation

Planning, defining, designing, executing, troubleshooting and interpreting the results. This knowledge area is focused on the performance testing and benchmarking capability.

Every performance test must clearly answer the “why”, why are we testing, and the “what”, what is the success criteria? The pre-requisites for a successful round of performance tests are;

1) The business goals and non-functional requirements are defined. Sometimes, the response time goals are not finalized, or the application is new and they are not sure what the goals should be. In this case, the PE/PT team can bring value by providing guidelines on the goals. For instance, a two second response time for 95% of a particular transaction. The goal of the performance test can also be a performance characterization, how do the response times change as the load increases?
2) Workload and testing scenarios are defined. What are the key business transactions and user profiles for testing? The testing team must create a simple spreadsheet model listing the steps in each user profiles, or messages, or batch processes. The testing scenarios include; response time characterization, stress testing, component testing, business peaks, etc. The testing scenarios are then mapped into testing scripts.
3) Properly defined and implemented testing environment. The performance testing environment must be as close to production as possible. A key success driver is the size and distribution of the database. An exercise must be done for the data analysis, if the database is incorrect, then the test results will be incorrect. You should also make sure you can simulate a remote client. Where are the users of the application? If they are in Europe, you must know the network bandwidth.
4) Application in a stable state. The features being tested must be in stable release state. Often time work is lost when the PE/PT team uncovered functional issues. For Agile approaches, the

team will start off testing incomplete functionality, as the Agile team builds each component. The performance team must be able to build out the database along the way for agile.

5) Configuration and release management. These processes can be the cause of many headaches and lost time. The deployment process must be automated and repeatable. The configuration in the testing environment must be locked down and changes tracked.
6) Core performance team. The model for a PT team is to have Performance Engineer in the lead role, with proper scripting and technical support. Depending on the size of the application, there may be Java developer on the team, an Oracle technical DBA, etc. The goal is to make the team independent and productive. If the team has to fill-out a form to get DBA help, then they will be waiting too often. The team adds value when they can trouble-shoot the problems to a technical component level. This way the application team is involved when there is a real issue, not because the PT team cannot run the test. The skillset on the PT team is critical in order to gain the respect of the development and production teams. Being test scripters is not enough.
7) Supporting team commitment. To be successful and manage the schedule, the support team must be aware of the testing schedule and ready to help triage issues. The support team can be a Websphere technical admin, or the application developer, or from any of the key supporting technologies. The key idea here is that they are ready to respond.

Capacity planning

The goal of capacity planning is to help plan what a new application will need in production and to help monitor the production application. The CP considers the application workload, the technical environment and the money (the cost to implement the system). They must take the output of the performance testing team, validate the testing workload and estimate the resource requirements.

The capacity planning team will develop various levels of models, queuing models, to forecast resource utilization, based on the workload. The CP is a consumer of APM metrics, collecting the workload profile for web transactions, messages, batch, etc. As well as the system resources usage, CPU, memory, disk, network, etc. The people in this group must understand linear and non-linear system behavior.

There three key areas for capacity planning;

1) Determine capacity requirements. This can be done for different environments, certainly for production. CP has role in sizing the performance testing environment and if they are familiar with the production workload, they can provide information to the PT team. The capacity must be determined for the database sizing requirements and growth plans as well. The CP team must have reference list of all the production applications and their workload/resource consumption. This information can often be the starting point for estimating a new application.
2) Analyze current capacity. The CP monitors the current production systems. They must notes trends for the usage of the system, has the users behavior changed over time? This can be critical to communicate to the performance testing team. They should monitor how the resource consumption changes with each Release? Often times, the operations team is caught off guard, because the new version of the application uses significantly more resources than the prior release for the same workload. (Possible chart here would be good)

3) Plan for the future. How will the resources be used for the next quarter? What significant business changes are coming to change the workload? The use of models can help here, but it depends on the nature of the business, does your business need complex model to predict resource usage as early as possible? Is it difficult to add servers to the mix, will the application support more servers? For instance, what is the business predicting for the holiday season? The CP team brings in the costing model for the over all business ROI from new or enhanced applications.

Application performance management

This is how you define the plan and execution for monitoring complex distributed applications. This is about monitoring the end-user experience, and the technical components that work together to service the request. It is also used for message bases applications and batch type applications or transactions. APM tools are essential to the root-cause analysis process, there is tremendous speed-up on finding the issues. If you are not using it now, you need to get an APM tool.

APM is composed of the following five subject areas;

1) End user experience measurement. More and more business transactions are being accomplished with a mobile device, in addition to browsers. The mobile device has a wide variance in the type of network connection is has. The monitoring tool must be able to track this.
2) Create a model of the run-time environment. When the APM tool is introduced to the environment, it can discover the technical components by monitoring the transactions. It will provide a real-time transaction map of the technical infrastructure. This will allow you to visualize the transaction performance, performance trending and analysis.
3) User defined transactions. The APM tools will automatically start tracking and monitoring web requests. They also allow you to define your own business or set of transactions to monitor, trend, and report on.
4) Deep dive performance analysis tools. The APM dashboard will provide you the overall health of the key transactions, and as a problem is found, then allow you to drill into the technical component. They cover Web Servers, Weblogic servers, Java Servers, database servers, etc. You can find where the slow part of the transaction is.

Problem detection and resolution

There will be production problems and there will be problems in pre-production environments. You need to be able to find problem quickly and correct them. This knowledge area makes use of the APM tools combined with techniques for root-cause analysis. Typically, this will reside in one or more groups. The operations team tends to be skilled at this, with support from the development team. The performance engineering team will make use of this KA during performance testing process as well.

The Enterprise performance engineering can help facilitate this experience and help capture the tools and techniques. Traditional problem solving activities have relied on the 20 person conference call and ad-hoc approaches to finding the root-cause, after the fact or while it is still happening in production.

Many people are involved, they look across all the technical components, scanning logs for key words from the Weblogs and application logs, maybe even heap dumps.

Formalizing the root-cause analysis approach provides great ROI, it helps reduce the time to find and fix the errors.

Supported by Competencies

Each of these knowledge areas require technical skills and expertise and non-technical skills for problem solving, project management, and communication. The performance engineer should both improve their ability in their existing competencies by going deeper, and continue to increase the number of competencies.

In the SA:SDLC knowledge area the Architect should continue to increase his use of patterns for performance, should understand what design patterns are best suited for high volume web sites. The architect should continue to grow his/her skills across multiple technologies.

In the PTV: Performance testing knowledge area, the performance engineer should have a solid understanding of the challenges of test data management. They should have a competency for creating synthetic data for performance testing. The lead performance engineer must be able to manage a small team and manage a schedule.

Supported by techniques

A technique is a proven approach for solving a particular problem.

As the aspiring performance engineer, you need to inventory your organizations techniques. In large companies, many times the same problems are solved again and again without creating a corporate performance engineering memory.

Examples:

SA:SDLC Requirements phase

What technique do you have to gather the non-functional requirements?

For instance, you might be competent when it comes to developing performance testing plans and test scripts. You will need a technique for the data required to support your test plan. What technique will you use to provide test data to the synthetic users? User id's are needed, account numbers may be needed, what's your technique?

PTV: Performance testing

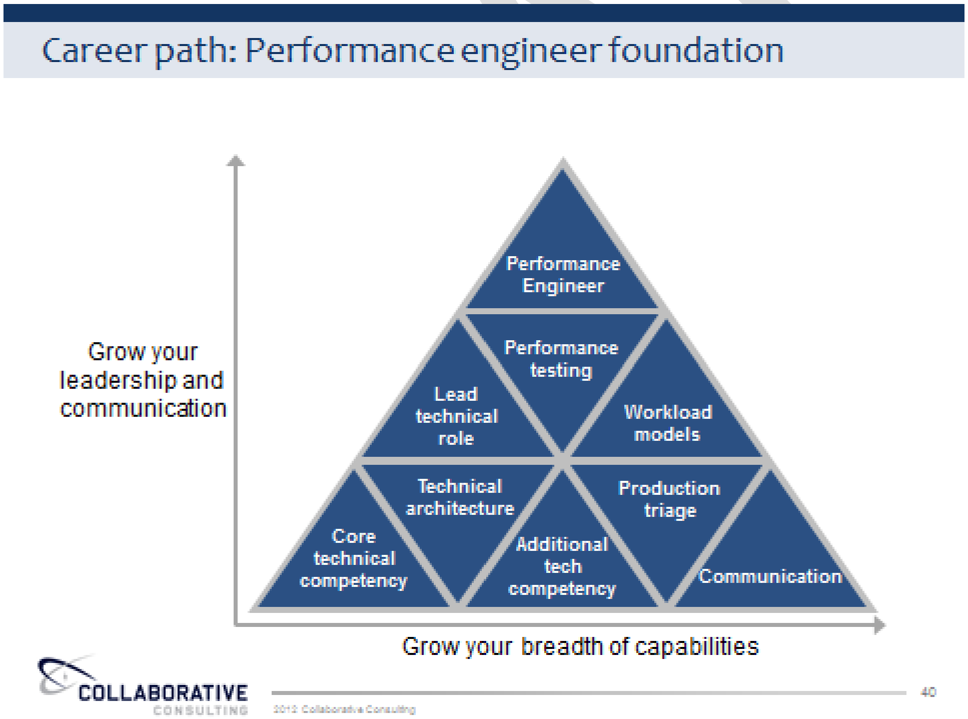
What technique do you use to load the performance test database?

Knowledge Area	Technique Short name	Description

3. What are the Competencies supporting the KA's

What does it mean to be competent in a subject?

It is a combination of knowledge gained from research and hands-on learning, developing abilities to implement and the supporting skills to gain increasing mastery of the topic. When you put together a number of competencies, you can start to define a career path. You can have senior people who are new to the topic, they may have 12 years of experience; however they are just learning the topic. You would expect them to master the topic fast than a junior person right out of college. There is a warning here, the senior person may move too fast and make too many assumptions about the new topic, the wind up causing problems when implementing it. You need to select two perhaps three core competencies that you become a recognized expert in. You should be recognized by your company and management team, and then become recognized by your industry.



Career path: Performance engineer foundation

Performance Engineer
Performance testing
Lead technical role
Workload models
Technical architecture
Production triage
Core technical competency
Additional tech competency
Communication
Grow your leadership and communication
Grow your breadth of capabilities
COLLABORATIVE CONSULTING
2012 Collaborative Consulting
40

Ultimately, you can look at your career as a pyramid where you continue to widen the technical base, and grow the height of you leadership and communication skills.

Non-technical competencies

Analytical thinking and problem solving

The performance engineer works in complex technical environments. You need to be able to work through complex problems and arrive at the root-cause with a list of facts supporting the conclusion. You must be very careful to make one change at a time to identify the benefits for the changes introduced. Many times people chase the wrong issues and do not understand how the entire system interacts. Or they find one issue, correct it and do not realize they have moved the bottleneck to the next component.

Project management and communication skills

Managing schedules is a critical part of performance testing. The testing schedule can change frequently based on the outcome of each test. Reporting status to Sr. IT management is critical to success and managing expectations on the outcome of performance tests. Managing expectations for your customer or client is important during a production troubleshooting exercise. The situation changes quickly and new information is introduced quickly. Being pragmatic is a core trait of the Sr. performance engineer.

Statistics and forecasting

This is very helpful for test results analysis and workload forecasting. Many people become stuck on the average response times for the business and system transactions. Averages are very misleading. The response time distribution provides more value to the business. The 90th or 95th percentile provides more value than the average. Understanding the transaction arrival rates is critical as well.

These are key competencies for the capacity planning KA; forecasting the resource consumption for the next quarter based on the businesses forecast for growth. Knowing how to select the proper trendline for the forecast is important with a variance or degree of confidence. How much weight do you give the most recent measurements? Predicting system resource usage is a tricky task. If you are not accurate or reliable then people will no longer listen to your recommendations.

Business knowledge

The performance engineer must be able to understand the business and the applications. Understanding the context for each Knowledge Area adds the most value. It is difficult to run a set of performance tests with little business knowledge, for instance the user profiles and the workload models are key to success. The more you know about the application the more value you add to the business and the development team will take notice as well. A performance engineer adds value to himself when he the business.

Software development methods and the SDLC

Understanding the SDLC becomes more important as you grow into a Performance Engineer. You should become an expert in your own company's chosen methodology and understand the typical issues your company has with the SDLC. Some companies always have stability and performance issues during the first 2-3 weeks after production deployment. You must be able to put a plan in place to get them out of this habit.

You are the person who will integrate PE tasks and activities into the development process. You will have to provide guidance on what the criteria are for a project that requires these PE activities and what ones do not. You must establish a solid relationship with the Enterprise Architecture team.

Eventually you will gain a solid understanding of Waterfall, Iterative, Agile, and hybrid approaches. A key item here is when working with third parties, how will you hold them accountable for the performance and scalability of the solution they deliver?

Technical competencies

These are a few examples of key technical competencies.

Monitoring and tuning

System and application monitoring is a must have for the Enterprise today, the systems are distributed and complex, with many working parts. You need to be able to define a monitoring plan for all your environments. The performance testing environment should have the same mentoring plan and tools as the production environment. This will allow the PT environment to better predict the production environment performance.

A performance engineer should know what to monitor and how to monitor it. For each Enterprise application you need to have an end to end view of the transaction(s) and the supporting technical components. You should monitor;

- the browser for response time of the requests,
- the network for bandwidth utilization and latency,
- the Web Servers,
- the application servers and operating systems,
- the database servers.

Proper monitoring allows you to create trending reports. To create trending reports, you must establish a performance database. The reports must correlate the workload on the systems to transaction performance and system resource consumption. This is a critical step to establishing a Capacity Planning process and capability.

How do you roll-up the measurements? Under normal conditions in production, you might measure every five minutes (12 samples per component per hour). During a critical performance test, you might measure every 10 seconds.

Core skillset – A Database engineer, a software engineer

The Performance engineering should have a core technical skillset. Many people accidentally become a performance engineer. Initially they start as a key problem solver and troubleshooter. However, they started in a role as a; software developer, or database administrator, or operations support member. These are hands-on roles. The Java Developer knows how the Java Virtual Machine works, they know how the threading model works, and they understand how the database connection pools work via JDBC.

The Database administrator learns to understand how the application is using or abusing the database. They understand how the performance tables and catalogue tables are used during real time troubleshooting. They can interpret the query plans, they understand the physical database design and can develop improvements for performance.

Because of their innate ability to learn new technologies, coupled with their ability to ask great questions, they become the de facto performance engineer.

The key item here is that you have a deep technical skill, continue to work on it, and start to cross into the other areas to being to put the big picture together. To solve the problems of todays complex systems, the performance engineer must understand the entire system end to end. You need enough understanding of the other components to ask the right questions.

For example, if you are the Java expert and starting to understand the database, you should know that wait events are a problem. You can work with the DBA to identify the wait events in the database.

Deep dive analysis tools – code profiling, database, network

Because you have core technical skillset, you also need to have a mastery of the tools/products that support you skillsets. The deep dive tools are required to support the root-cause analysis process. These are part of the overall monitoring strategy for your APM solution. If the APM monitoring tools finds a slow transaction that is spending a majority of the time in the database, then you need to do a deep dive analysis of the database.

These tools can be use across all the Knowledge Areas, the SA:SDLC Knowledge area is where you can prevent performance defects. There are tools that can help, for instance static code analysis. The performance engineer should be able to provide input in to the Static Code analysis tools, to provide rules for performance patterns. In the development phase you can help prevent performance defects from getting past Unit testing. There are even tools to help automate the peer code review process.

There are requirements tools that help track functional and non-functional requirements through the development process. When a critical business transaction have been identified with performance goals, for instance; The Check order status transaction must respond within one second for the 95th percentile under a peak user load. This requirements tool can help trace this though the SDLC.

You should have a list of deep-dive analysis tools at hand and people well trained in the use of the tools. This type of tool kits is required to successfully implement the Problem Detection and Resolution (PDR) Knowledge area. Remember, applications and systems will break, you must have a well defined approach to find them fast and fix them fast.

Test data

A large amount of test data is required to support a performance test, it must be an accurate representation of production. For a performance tests or benchmark, the database must be populated to the right size, and then be able to grow in size.

There are many types of tests, some require the ability to continuously run performance tests to profile the system, some require the ability to replay a days' worth of production transactions to test the new release.

There are several starting points for a performance testing database;

- A full copy of the production system: this is becoming less of an option with security and PCI requirements, often times the data must go through a masking process before it can be used.
- Starting with a smaller seed data, then replicating and increasing the size of the tables.
- Using programs and tools to create synthetic data to be loaded. This requires knowledge of the data distribution within the production database. How many customers (active, inactive), how many orders, how many lines per order, billing history, etc.? There are many tools out there to use from Microsoft to Computer Associates

The second category is the data required to support each test case and the Virtual Users. You need to identify what data is required for each test case and where the data will come from. If you have 500,000 customers in your test database, then how will you distribute them across the virtual users?

Code profiling tools

These tools are very helpful during the development phase, and even during performance testing when you are in diagnostics and tuning mode.

4. What are the Techniques supporting the KA's

Each of the knowledge areas requires competency and techniques to be effective. Your goal should be to capture and inventory as many techniques as possible. To improve the organization, sharing these techniques will help greatly. In large companies, where business units can decide how they do work, they often times solve the same problem differently and solve the same problem many times over.

Capturing these cases of solving the same problem and then establishing a best practices guide is beneficial to the business and the performance team. To help show the value of performance engineering you should provide a periodic update to your management on the number of Best Practices you found and categorized.

Something as core as performance testing can be recreated many times in the same company. They each have data problems to solve, they have to create the proper user profiles for testing, and record the scripts to generate the load. This is often done with a "pick-up" team. The people running the performance test may not have worked together as a team, and the LoadRunner Scripter may be in another continent as part of a Center of Excellence team. The only excellence they tend to have is writing a script and running it. If they break, they add little value.

There are techniques that help the performance testing team be more effective and efficient. We really want to capture the techniques and prevent the doing it the first time use case.

Creating test data

Triaging and Brainstorming solutions

This is a critical section of the entire software and systems universe. Brain storming and triage is typically driven by more senior performance engineers with strong leadership abilities. How do you solve problems quickly and verify them before putting them into production. What techniques do you have to navigate multiple teams and groups, then to focus them on the next steps? What techniques do you have to keep the teams on the same page?

The Knowledge Area for Problem detection and resolution (PDR) is where many of these techniques get used and tested. When solving a problem a key guideline is to make one change at a time to measure the impact (positive or negative), then evaluate and move on. A Triage event typically occurs under tight timelines.

System metrics collection

How do you collect all the metrics when you don't have the right tools or products? What techniques do you have? The options you have will vary by the environment you are in, development, integration testing, performance testing or production? You need real-time access to the system performance metrics to help correlate event and outcomes.

For instance; when collecting metrics in a Windows Server, what do you do with the Windows Performance counters? For Unix you can use VMStat, IOStat, etc. Have you created some simple scripts to help schedule the measurement tool and the intervals, then write to a log for analysis.

In a performance testing environment, how do you use the metric collected form HP Sitescope?

Summarizing performance testing results

Performance test scenarios development

Creating a workload model

Planning, coordination, information sharing and control

1. Where is the leader and point of contact

To truly harness the value and show the value that an Enterprise performance engineering practice can add, there needs to be a leader. The Sr. person who can communicate the value, help make sure that PE tasks and activities are not options, and keep the team ahead of the curve. They will help plan and manage the budget for the core PE team, and help the major Enterprise Programs plan and budget for PE in their projects.

Organizational positioning

The PE team is found in many different organizational structures. It is found in the QA group, the Operations teams, a shared service for performance testing, and in the development group. Each has its strengths and weaknesses for the team. The criteria for selecting where the PE team reports;

- Wide view of the projects
- early awareness of critical projects,
- Requires technical career path
- Included in the IT governance process
- Integrated into production and release processes

The PE team can either be integrated into the Enterprise Architecture team or it can be a peer group to the EA team.

Large fortune 1,000 companies will have a central or shared IT services, plus IT services in the business units. In this case a Federated PE group will work most efficiently. The central group will help with Enterprise coordination, best practices, return on investments approaches, and keeping ahead of the curve. This group will have three to seven Sr. Performance engineers who track the large PE projects. This group must understand what he Organizations risk profile is. They will be at the Director level and should be a direct report to VP of IT services.

In the business units or in mid-size companies, the PE team will be planning and execution on specific projects. The career planning will occur here as well. The team must be part of the Development organization

Project or program start-ups must include SPE selection criteria

The PE processes and activities must be included in the project initiation phase, the Stage Gate process and any post go-live processes. This is necessary to help determines the technical risk of a new project and new product. Here is a list of selection criteria;

- User population: Open population with new customers, or very large
- Application type: Enterprise middleware, Enterprise data warehouse, Supply chain
- Application or system technology: New technology, major new version, vendor has not demonstrated success at your business volumes
- Scale and scope of the modifications to the applications
- The Software development process: are non-functional requirements tracked?
- Production issues with the last release
- Performance test schedule: how well planned is it and are the resources available

Create the SPE feedback loop – SDLC to production and back

A key responsibility the performance engineer has is to measure the production performance and then to communicate to the development team, the Sr. IT management team and the business. Is the application meeting the service level agreements?

The Application architect and the lead developers should be informed on a regular basis of the production performance of the application. The performance testing team should be aware of the production information as well. Both teams must know if the workload or the user behavior has changed resulting in a different resource consumption profile.

Many times when a new Release is moved into production, the application consumes significantly more resources than thought. We need to get out of this predicament; the development team and the performance testing team lose credibility when this happens. The business does not understand why the system response time changes.

Every developer and architecture must know how their application behaves in production

Embrace the Dev/Ops trend. This is a critical guideline. Many times the development team is unaware how their application is working in production. The Enterprise Performance engineer must facilitate this communication. There is an emerging trend of the Dev/Ops, where the operations team works with the development team, or the development team works in the operations team. This allows both teams to gain tremendous insight into the application and production systems. This gets the developer out of developing code in a vacuum.

SPE Roles

1. Do roles cross the KA's?

Each person will start out with one KA being dominant. A test automation person will start in the Performance testing and validation KA, the performance architect will start in the SDLC KA, the capacity planner will be in the CP KA. Your personal goal should be to define your career map to cover all KA's to some degree. Your Major can be SDLC, with a couple of minors in APM and PDR.

a) Technical project manager
b) Performance architect
c) Performance engineer
d) Performance test developer
e) Performance data architect
f) Capacity planner

Establishing common job descriptions

You can start with this template and review with your Human Resources department. In many large companies, introducing the Performance Engineer role is difficult. To maximize the value, the role crosses organizational boundaries. There is a trend in these large companies to define an Engineering group, such as Database engineering, and an Operations groups. The performance testing team is somewhere else. For instance, if the performance engineer wants to select and install an APM tools in testing environment, there are several other groups who currently must approve the request, and the budget. Often times, the key question is who will pay for it?

The value of a performance engineer is enhanced with an end to end viewpoint, solid technical knowledge and great communication skills. The performance engineer connects the design and the implementation. They know how the systems behave and what is was designed for. Minimally, the role is to follow the hand-offs across the SDLC. However, in so many companies, no one is responsible for performance.

Recognizing the performance engineer

Today the role and responsibilities for a performance engineer vary by organization, for immature organization the starting point is performance testing. However they do not recognize that there are performance tasks and activities across the SDLC and Operations. The desire to improve the role of the PE could be the need to provide a better user experience on the web site. To improve your web site performance, it requires better design and coding.

You, as an aspiring PE, must be able to define the compelling need to kick-start the Enterprise Performance Engineer. Typically, the starting points align closely to the Knowledge Areas.

Ultimately, you will have a senior manager recognize the gaps; they realize they are not building well performing systems. The production issues point them there.

Regardless of the starting point, performance engineering tasks and activities must be introduced to the SDLC. Build better code.

Junior and Senior

Like many other key roles in companies, there are junior PE's and senior PE's and of course middle level. The more junior role is typically focused on project execution and delivery. Working on one project. The skillset is technically focused and starting to learn the formal techniques and methods. For the APM KA, the junior person must know the technologies being monitored, but learning how the APM tools are implemented, and how the various levels of dashboards are created.

The mid-level person must be approaching the expert level in one of the KA's and working towards another. They must also be working on their communication and presentation skills. The knowledge of the business becomes critical at this level as well. You should be starting to work on multiple projects are one time and leading small teams.

The senior-level person is focused on the Enterprise. There are two versions of this level, one becomes more focused on the best practices and managing the group, and working with the business. The other stays technical and project focused. They can help capture the Enterprise techniques.

Career paths

This is critical to define in a company for the performance engineer. Without the formal career path, then people will not stay in the role. This is multi-disciplined role that crosses quality assurance, IT risk management, architecture and development, and operations. It is the combination of design and implementation, of engineering and operations.

The career path starts in other disciplines, stating in software development, database administration or operations, and then the performance engineer starts to combine them. Often times the catalyst is a major production performance event that requires a triage team. The Enterprise performance engineering group will be the manager of the career path. The group then must either report to a VP or CIO, directly. In some companies, due to their need for highly performing systems, this happens. They group can also be part of the Enterprise Architecture team. This can be a culture change as well as many EA team don't actually implement.

Where do they come from

As you look at the next three figures, you will see a common theme. It revolves around troubleshooting and triage. Troubleshooting is where you get to assess the entire system; a must for root-cause analysis. Once you master a core skill and learn to troubleshooting you expand your capability.

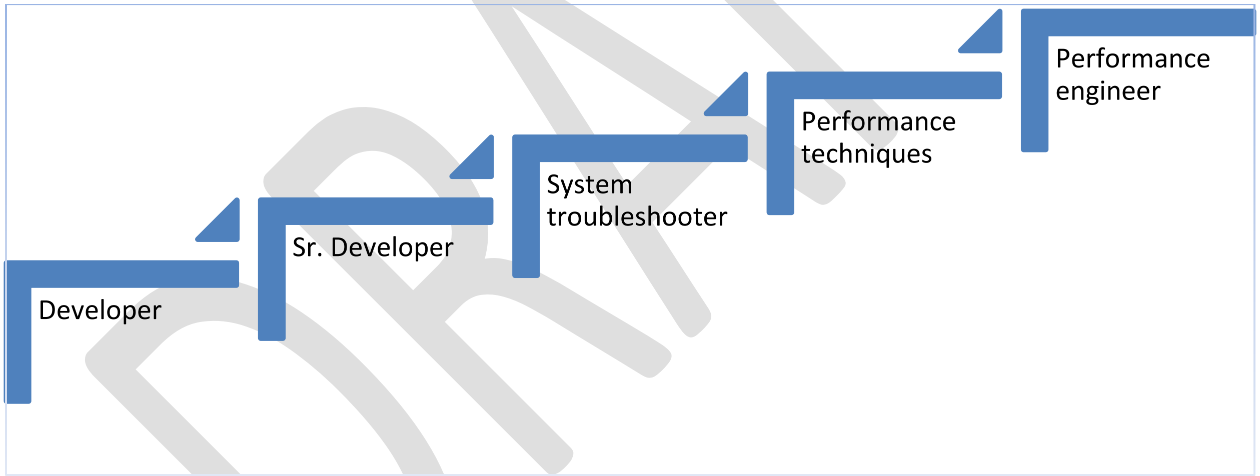


Figure 3 - Developer to PE

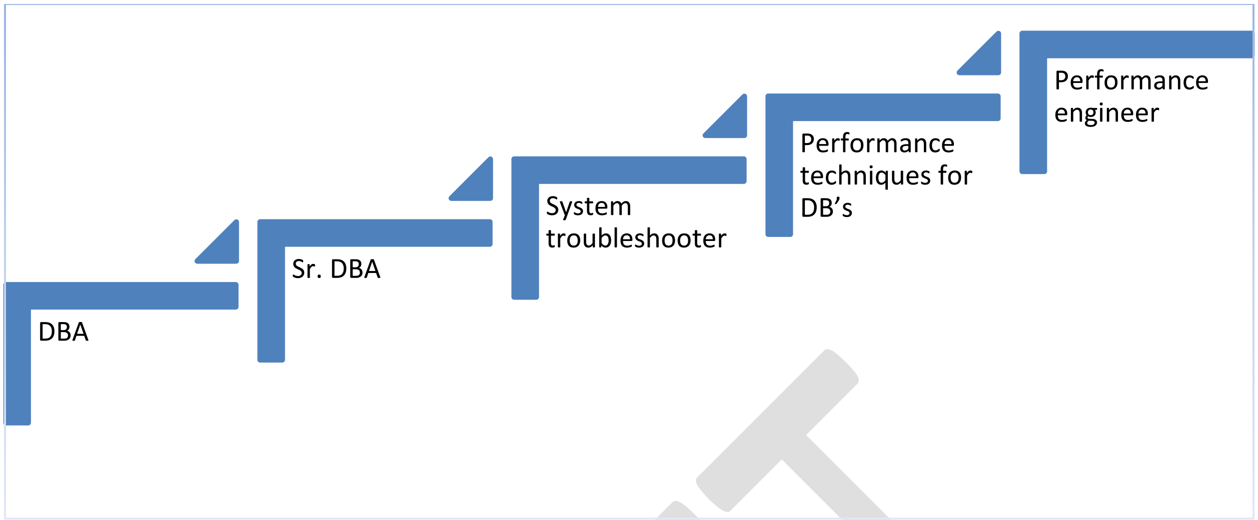


Figure 4 - Database to PE

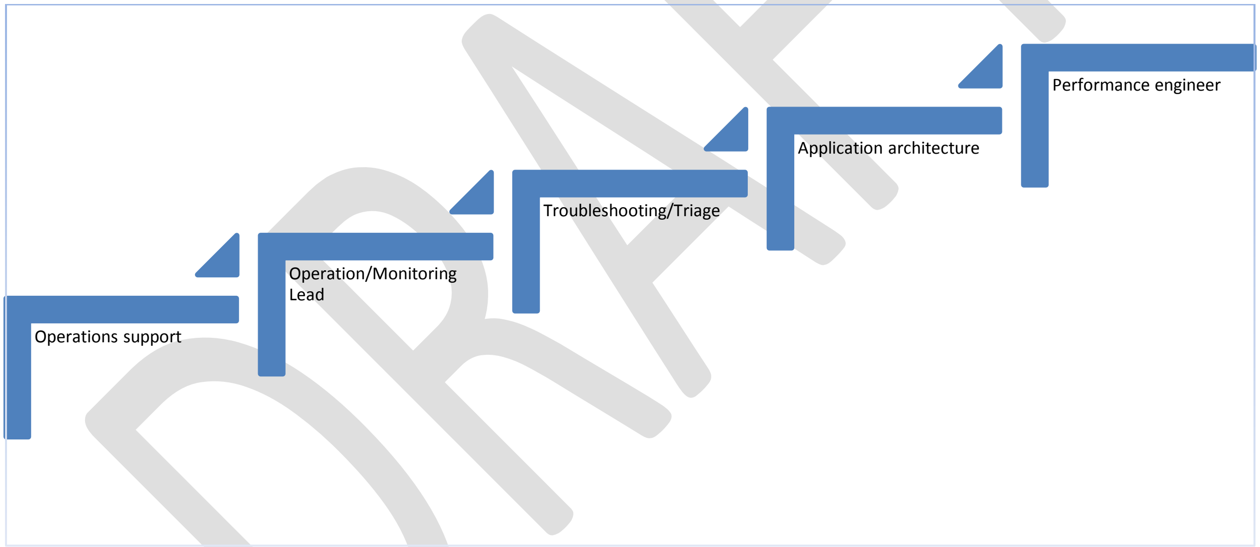


Figure 5 - Operations to PE

2. Technical Project manager

The technical project manager plays a key role for the Enterprise performance engineering team. In fortune 1,000 companies with dozens of large programs in various states of planning and execution, the technical PM works closely with PE team. This is a classic case of letting people do what they are good at. The PE person must have a proficient understanding and appreciation of project management skill. As the project increase in size, we don't want the PM tasks to dominate the PE's time. It goes without saying, the technical project manager must truly understand the technical concepts, to understand the risks involved.

In smaller projects and where the performance engineer is on-loan to a project, he/she must be able to plan and manage a small project. They must be able to plan the activities and report progress on those activities against the plan. The must be able to create the weekly and monthly status reports and provide status to Sr. management on the program.

Communication

The PM must be able to communicate progress and communicate risks, in technical terms and business terms. A key item to communicate is the unplanned event; these happen in KA:PTV performance testing, and KA:PDR problem detection and resolution. How to the unplanned events impact the schedule. This can be an area of misunderstanding for the business and miscommunication for the performance engineer.

Project profiles

3. What is a Performance architect

The prerequisite is that you must be a solid Enterprise or application architect.

Performance starts with the requirements (NFR's) and the proper design. Non-functional requirements are also known as the key quality attributes of the system. The architect is responsible for translating the business requirements in to the technical solution. There are many software architects who design systems to meet functional requirements. A few learn the hard way that applications also have non-functional requirements. In our context, we care about performance, stability, and scalability. There are many more key quality attributes; portability, security, maintainability, usability, for instance. To learn more about the key quality attributes, see the Software Engineering Institute (SEI).

Understands the business

Understand the overall business context and profile, for instance; what are the overall business goals? What is the business' tolerance for risk, do they view technology as a key differentiator, or are they a technology laggard? How do they allocate the budget for building systems?

For the specific project, they must have a complete picture of the functional and non-functional requirements. The Performance Architect must be the gatekeeper of the NFR's. Many companies are not mature enough to capture or give proper attention to gathering the NFR's in the requirements phase. As such, if the Performance Architect cannot find the NFR's, then he must escalate and verify. He must be able to identify the critical business transactions and assign performance requirements to them.

The Performance architect sets up the traceability of the non-functional requirements of the critical business transactions. The critical business transactions must be monitored during the design and build phases of the project. The architect must communicate to the developers the performance requirements of the critical business transactions.

Aligned with the SA:SDLC KA

The primary knowledge area for the performance architect is the SA:SDLC. They must be an expert in the development lifecycle and methodology. In the lifecycle they know the processes, tasks and activities that are specific to performance and scalability. They have the techniques that enable performance in the design phase and the development phase.

They are fully aware how the volume of transactions can drive out a different design, they have low volume solutions and high volume solutions. When do you use polling to communicate a state change or when do you use an event driven architecture?

An Application architect fully understands non-functional requirements

The NFR's set the goals for the technical components of the system. Here are few examples;

- The application needs to support the volume that 700,000 customers will produce.
- The Order Status online transaction must respond to the user request in two seconds or less for the 95th percentile under *peak seasonal* business volume.
- The insurance policy quoting engine must respond to a Rate Quote request in four seconds for an Auto Insurance policy request and support three rate requests per second.
- To prepare to bring the new system live, the history data conversion and validation process must be accomplished in three days, 72 hours. There are 2 Million history records.

The performance architect will takes these requirements and design the right technical solution.

Understand transaction response time, component or system throughput.

Every architecture and design decision is made in the context of the response time goals and the throughput goals. Each critical business transaction can and will have different performance goals. He is aware how the complexity of the transaction influences the response time. A simple order look-up versus a show me my orders for the past year, are different.

The throughput influences the scalability approach; 1 per second or 10 per second? He must design a system that is horizontally scalable. For our Insurance rate quote example; if one rating engine can support 2 TPS, then five engines can support 10 TPS. The Performance architect must be aware of the scalability curve of the application.

The Performance Architect is fully aware of the technical architecture. He does not simply design something and then hand it off to the Infrastructure team to let them solve for the performance goals. He knows that an application with transaction response time goals cannot be designed without consideration of the actual infrastructure it will run on. He is aware of how the JVM Memory size can impact the transaction response times

Monitoring and measurements

The application was designed and developed with performance goals, it must be measured to validate that those goals were achieved. It must be measured in the performance testing phase and in production. Many architects are not involved in the actual measurement part of the lifecycle. It is a flaw in the organization, they move the architect along to another project too soon. The architect does not get to see how the application is performing. Is he even held accountable for performance?

The Performance Architect is fully involved in the monitoring aspect of the application he designed. He is there at the end to see how the application meets its deign goals. To borrow from the Civil engineering, he is there to walk over the bridge.

Has a solid understanding of the performance testing process

The PA must be able to design and oversee performance tests of the systems they design. They should know what types of test should be run to stress test the application, to validate the scalability curve of the application and to test the stability of the application.

The PA should be part of the results analysis review to help decide if the results of a test are valid and it was a successful test. The PA will work with the performance engineer who will lead the test scenario design, automation scripting, results collection and analysis process. The maximum value is derived when the application architect or the performance architect help design the performance test and review the results.

Solid understanding of the physical database

Many architects lack the appropriate understanding of the physical database system. The PE has a strong understanding of the physical database model and design. Many performance problems arise when the architects lack the understanding of the database, during a triage event this often the first place to look because the architect did not place any SQL code reviews, and a how the application will access the data within the database.

They should know how to place tables in tablespaces based on how the tables work together, and they should know how to help design the index scheme for each table and the queries using the tables. Another area the PA must be competent at is knowing when to take advantage of a table partitioning scheme.

4. What is a Performance engineer

The question on everyone’s mind. I am reminded of an old Saturday Night Live line “It’s a dessert topping, no it’s a floor wax”.

Over the past few years, even longer, this role has been stretched and twisted into many different interpretations. Many people think that a performance tester is a performance engineer, or a person on an operations monitoring team is a performance engineer. These are roles one goes through on their way to becoming a PE. The role has also been confused by the occurrence of remote and off-shore testing centers of excellence. The CoE’s are really only test scripting centers. Generally, if they cannot run a test due to an unknown technical issue, they stop and contact the application team to solve the problem. If your performance testing team cannot solve the basic configuration issues, then don’t bother.

A few key attributes the performance engineer has are; multi-disciplined and are good at them, they communicate well verbally and in writing to technical and non-technical people, they can lead a team to triage and solve problems. They understand how to monitor an application, they understand the physical side of the technology.

The performance engineer can face of with the architect, the database administrator, or the product vendor. A performance engineer can play a key role in keeping these people honest about what is possible.

Understands the business and can create performance test workload models

The PE is able to translate the business transactions into a moderately complex spreadsheet model. The model is able to show the critical business transactions all the way through the system level transactions. It also includes the business volumes and arrival rates.

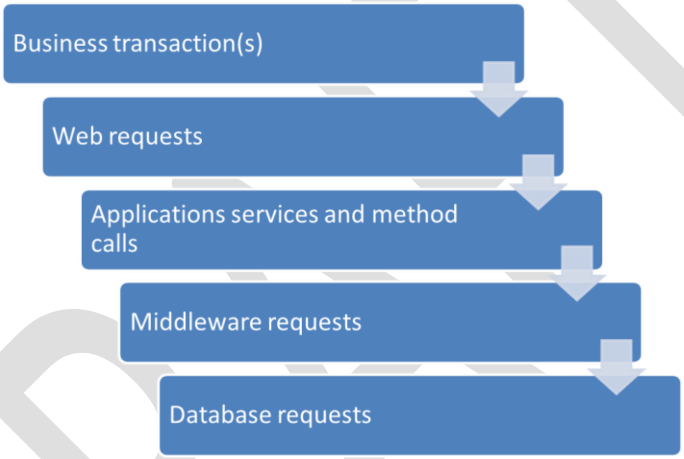


Figure 6 - Transaction decomposition

Aligned with the PTV: Performance testing and validation, or APM

The PE can design and execute performance, scalability and stability test that will break the system. This is because the fully understand the technology and have experience across different systems. They have seen where the weak points and they know how to find them.

The PE has a repository of techniques that support performance testing projects. These projects vary significantly in size and duration. For example;

- Release based performance and integration testing: The business unit has a monthly release cycle and all the applications in the release must go through performance regression test. This focuses on many applications being tested at once.
- Vendor product assessment: Before the product can be selected, it must undergo a series of performance, scalability and stability tests.

- Major program: Large companies have multi-year programs, the program itself will have a dedicated performance team for regression performance test, tuning and production support.

A quick checklist of key skills and roles.

Competencies	Comments
Fully understands the performance testing process and can lead PT project	
Solid communication skills	
Has a one or two core technical skillsets	
Hands-on engineer	
Expert in one or more performance testing tools	
Can create test results reports and dashboards for Sr. management	
Solid understanding of metrics collection from technical component	
Can design and create test data	

5. Performance test developer

This can be a great way to get a junior developer into the performance engineering career path. They must be a developer and not a QA functional tester. The QA functional people typically come from the Business Analyst path, not the technical path. To become a performance engineer, you must start with a solid technical foundation. A developer with 2-3 years of experience coupled with a veteran performance engineer, can really accelerate the experience of the junior developer.

The purpose of the performance test developer is to create the performance test scripts. The test scripts simulate the steps an actual user will execute on the application; whether a mobile client, a call center, or even remote access call.

Aligns with PTV knowledge area

This role exists in the PTV knowledge area.

There is Jr. level and a Sr. level

The junior level person starts out working for the lead performance test developer or the performance engineer. They create the scripts for the test plans and understand the data required to support the scripts. The Junior person should be able to kick-off the actual test and be able to watch the indicators to determine if the test is running normally. They can help with the mechanics of the test, but may not be able to interpret the test. During the test execution they should monitor the health of the servers and the error rate of the test harness. There is to make sure a valid test is executed. The test may fail because the system is slow, but is should not fail because no one was aware that a file system filled up.

The Lead performance test developer should be able to triage the scripts are they are executing to make sure the test is running successfully and all the system level metrics have been captured for later analysis. They are the key person watching the test. The performance engineer will supervise the performance test developers. However, the performance engineer should not be running the tests. Some tests should be routine, that do not require the performance engineer to participate in. For example, a performance regression baseline test, this is executed when new Release of the application is moved into the testing environment. You need to determine is the new Release is faster or slower. The more Sr. person should start to ask questions about the technical components such as the configuration of the JVM, and begin to understand when the database is not running normally, etc.

Can develop test case documentation

This is a critical component for proving that you ran a test and capture the results. The test case document describes why the test was selected, what the goals are, and what the expectations are. This is needed for repeatability. It is required when you have a distributed testing CoE. The performance test developer will be able to describe the data required to support the test case for the documentation.

Additional capabilities	Comments
The developer should be able to learn and apply hew testing tools.	
Develop and debug test scripts	
Execute tests and collect results	
The Sr. level can lead one or more Jr. test developer resources	
The Sr. level has an expert understanding of test data correlation	

6. What is a Performance data engineer

Aligns with CP, PTV, APM

For large complex systems, a performance data engineer is a must; because, historically, the database and the SQL statements have caused the most trouble when implemented by developers and application architects who lack a solid understanding of a database management system. Experience tells us, that when the application team refers to the database as persistent storage there is much opportunity for performance improvement.

This role requires deep understanding of the physical implementation of the database system. They must understand how the application will use the tables and the access paths for those tables. This insight is needed to define the indexing strategy. They must be able to create a physical implementation model spread out across the disk subsystem. This role must fully understand the normalization process and when you must de-normalize for performance. The goal is to minimize joins for very large tables.

Provide guidance on writing SQL statements, database packages and procedures. They provide guidelines on the of the "hint" statement, don't use it. Understanding the impact of views is very important. With database views there is a balance to strive for between too many views and too few views. In most large organization there is a mandatory one view per table, then additional views based on application requirements. Views can hide complexity, however, they can be tough to maintain for new team members.

This role requires the person to be very familiar with the particular optimizer for the database. They must be able to read SQL plans, know how to access the relevant performance tables and views, understand wait-events. What are the different types of wait events? What are the top SQL statements?

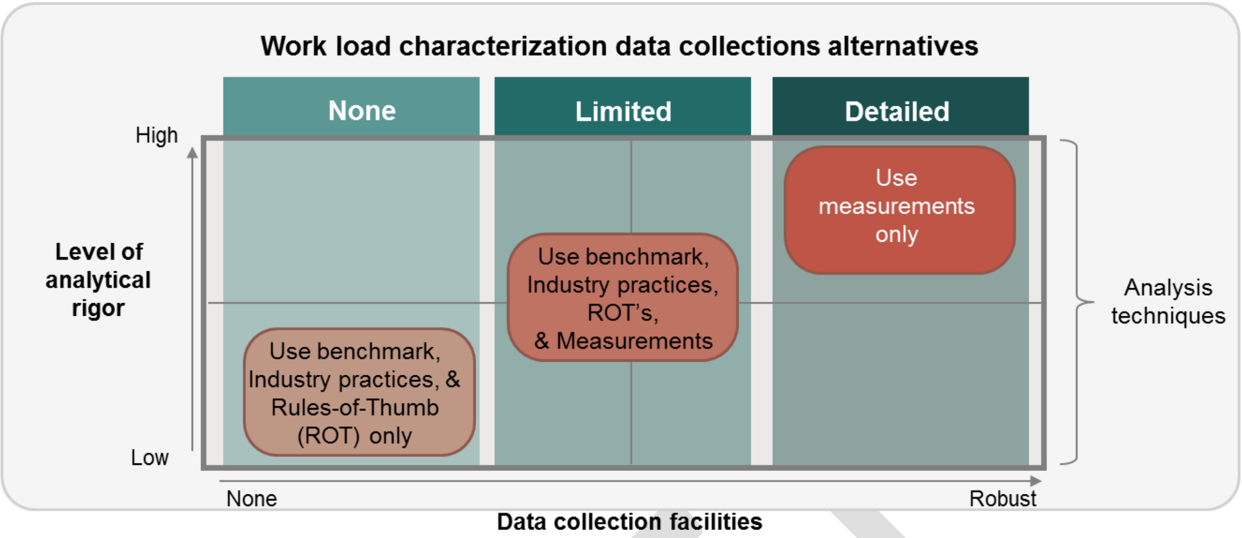
Requirements	Comments
Fully understands the process to create test data for performance testing.	Critical to a successful performance test, the data distribution and history is important.
Can read and create the ERD's	
Can define and create data generation scripts or programs	They can specify the process to use to create the data, the performance test developer may actually create the scripts and run them.
Can create and correlate test data for load testing	
Can create a forecasting model for database growth, either for a performance test, or for business growth.	

7. What is a Capacity planner

Aligns with CP and APM; can support PTV for workload models

The Capacity planner is responsible for making sure the right level of computing resources are available just before you need them. This requires knowledge of the current workload on the system and utilization of that system. There are a number of prerequisites; you need to have the proper collection and analysis processes, methods and tools to use the current data for forecasting. You need to forecast the workload and then the resource utilization. This is a tall order. The other critical responsibility is to factor in the costs.

The capacity planner used many different models to forecast, at different levels of the technical architecture. There are many modeling approaches, closed systems and open systems, analytic models (aka queuing models) and simulation models (event driven computer models).


Work load characterization data collections alternatives
None
Limited
Detailed
High
Level of analytical rigor
Low
Use measurements only
Use benchmark, Industry practices, ROT's, & Measurements
Use benchmark, Industry practices, & Rules-of-Thumb (ROT) only
Analysis techniques
None
Robust
Data collection facilities

Organizational alignment

In many organizations there is a capacity team and a performance team. Often times the two do not meet frequently. The capacity team is typically in the production operations group, and the performance team is in QA, Development, a business unit, or elsewhere in a pre-production group. The CP team is really removed from the application architect and the development team. The application team is building without any real knowledge of how much capacity is required to support the workload. The CP team is often involved late in the life cycle. When these two teams or responsibilities work closely together, the business benefits greatly.

ROI: Must factor in the cost

Capacity planning is where the dollars are factored into the application. It is a very useful exercise to create a dollars/transaction cost metric for the application. This can be helpful when solving performance issues and they keep telling you to throw hardware at it.

Understanding capacity planning; present and future needs

Analyze current capacity

You need the ability to collect a large amount of system and application usage data that will be used for analysis and forecasting. The Capacity planner must fully understand the current capacity of the production environment. In large organizations the production environment is large, supporting dozen of business critical application. Ability to create trending for the business workload on the application or system, track utilization of existing systems

Plan for the future

You cannot plan for the future until you fully understand the current utilization and capacity of the current system. You can then plan for the future for the existing system. You can start with the basic application workload forecasting, how will the users of the system grow or change behavior? Is the business changing the user/customer behavior in the near future? This is a top-down approach.

The other is an infrastructure or bottom up approach. The IT team decides to change hardware and need to understand how changing the hardware will impact the capacity of the system. The workload may not be changing; however the underlying system is changing.

Solid understanding of Service level agreements

Solid understanding of modeling techniques

Sample Techniques

Knowledge Area	Technique	description
SDLC: requirements	Gather critical business transaction response time	