

# Application Performance Management Terminology Reference

The technologies of APM continue to evolve rapidly, as does the entire IT environment. As in every industry, the terminology used to describe technologies and methodologies evolves over time, as various vendors and practitioners apply labels to their offerings and activities. This is a living document OPNET will update periodically to act as a glossary of APM terminology. Its purpose is to facilitate communication on this topic. We welcome any suggestions for enhancements or modifications, which may be sent to [acohen@opnet.com](mailto:acohen@opnet.com).

The terms are currently arranged in a logical order, so that, to the extent possible, later terms can make use of earlier definitions.

**SERVICES** are the functions of IT that are utilized by the rest of the enterprise in order to conduct its business. In most cases, services take the form of software applications used by an enterprise's lines of business. There are other services that are not typically viewed as "applications", such as the enablement of access to information (e.g., *file storage and retrieval*), telephony, and video-conferencing. Services may also be external, such as an e-commerce web site used by an enterprise's customers. Ultimately, services are IT's reason for being – the only purpose for which the enterprise funds IT.

**APPLICATIONS** are the most common embodiment of a service. Applications are implemented in software and provide users with an interface to accomplish various business tasks or obtain desired information. The interface is typically presented to the user via a screen on a desktop computer, terminal, or handheld device, and is designed to allow the user to specify the parameters of the task to be performed. The users and also the interface by which they access the application are referred to as "clients." The client commonly resides within a web browser, or a dedicated desktop program (*sometimes referred to as a "rich, fat, or thick client"*). We often think of the client as being a human user, but it may also be another application that makes use of the service.

**TRANSACTIONS** are a unit of work performed by an application. Most applications support many different transactions related to a common business area and depending on a common set of data. Each transaction is designed to accomplish a particular business purpose and exposes parameters to specify the particularities of the transaction each time it is requested by a user. This description per-

tains to a "business transaction". Some Application Performance Management (*APM vendors also refer to lower-level requests and responses as transactions, e.g., a web browser obtaining an object from a web server*).

**END-USER-EXPERIENCE (EUE)** refers to the quality of the interaction between the user and the application from the user's perspective. While this can relate to all aspects of the interaction, including ease of use, intuitiveness, etc., APM concerns itself only with performance-related aspects of the experience, such as response-time and availability.

**SERVICE ASSURANCE** is the discipline within IT that concerns itself with ensuring that services are provided with the level of reliability and performance required to satisfy the enterprise, which is the "customer" of IT. From a management perspective, the major activities within Service Assurance are Application Performance Management (APM) and Network Performance Management (NPM). Because most services are business applications, and because virtually all business applications depend heavily on the network, NPM is a critical activity within APM.

**APM** stands for Application Performance Management or according to some definitions, Application Performance Monitoring. "Management" is a superset of "Monitoring". Monitoring includes real time acquisition, analysis, and alerting with respect to performance and availability indicators, as well as long-term storage of such metrics for trending purposes. Management includes all of these capabilities, and additionally: gathering of forensic data for the purposes of analysis and troubleshooting; and analytics to automate the process of root cause isolation and problem prevention.

**NPM** stands for Network Performance Management and covers the capabilities of monitoring, alerting, trending, and problem resolution and prevention, all as specifically relates to the network infrastructure that supports IT services. Because the network infrastructure is fundamental to the operation of virtually all IT services, NPM is an essential part of a successful APM approach.

**MANAGEMENT DOMAINS** are the various components of the infrastructure that support an application. Management domains typically include desktop computers, the wide area network, local area networks, web servers, application servers, databases, and storage systems. For each domain, there are typically specific management tools for monitoring and troubleshooting. These tools are oftentimes used by distinct teams within the IT organization. When management domains are treated in an entirely separate manner for the purposes of APM, they are referred to as management "silos".

**END-TO-END APM** refers to the notion of a cohesive approach to application performance management that operates across management domains, in contrast to treating these domains as silos. An end-to-end approach is based on the following capabilities:

- Easily transitioning from one management domain to another, while properly maintaining the scope of information for the workflow of interest. This is commonly referred to as a "contextual drill-down" from one domain to another.
- Generating unified views and reports combining information from multiple domains;
- Performing correlation and combined analysis across multiple domains to facilitate root-cause isolation of performance issues.

# Application Performance Management Terminology Reference

**PERFORMANCE DATA** consists of measurements that are captured and recorded over time, and usually aggregated in various ways to provide information that is statistically representative of system performance. Aggregations are typically across dimensions such as:

- Time intervals (*e.g., minutes, hours, days, weeks, or months*).
- User communities (*e.g., business departments, geographic locations*)
- Transaction families (*e.g., logins, on-line purchases, URLs with a certain prefix, etc.*)

**PERFORMANCE DATA** is usually archived to provide historical views and trends, and also analyzed and presented in real-time for the purposes of monitoring, alerting, and troubleshooting. Performance data can pertain to the end-user experience, as well as to the health and status of the underlying components of the application.

**FORENSIC DATA** goes beyond quantitative metrics, and is obtained by recording information during a particular event of interest, such as a transaction of a certain type, a performance alert, or a program exception. The information is often of a detailed nature since one of its common purposes is to serve as evidence or to provide clues (*hence the term "forensics"*) in the retroactive analysis of the event. Forensic information is generally obtained from lower-level techniques, such as code level tracing, detailed resource tracking (*e.g., memory consumption*), and deep packet inspection of network traffic. The data is used to "drill-down" deep into the behavior of the application, determine how components are interacting with each other, and break down all activity related to an individual transaction or other events of interests. Rich forensic information is fundamental to providing root cause isolation capabilities.

**HIGH DEFINITION** refers to a management approach that operates on the basis of sufficiently complete information to significantly automate the process of resolving and preventing performance problems. To provide a consistent advantage across a broad spectrum of performance issues, an APM solution must be sufficiently broad and deep. Breadth implies that the management tools span multiple management domains. Depth refers to the ability to obtain and process extensive performance and forensic data within each management domain. This data must be of the appropriate nature to support the troubleshooting process. The term "high definition" also conveys the requirement that the data be of sufficient quality, which has implications for aspects such as the time-resolution of the data, the specialized nature of performance metrics, and the level of detail retained from forensic tracing.

**HIGH DEFINITION APM** may be contrasted with approaches that merely provide a coarse indication of "where" the source of a problem might reside. Worse yet, such solutions are routinely used to exonerate individual components of the application infrastructure based on highly diluted performance data. Such, "low definition" approaches may, in some cases, be useful for initial triage among potential problem sources, but also often mislead administrators in their search for a root cause. This occurs because the quality of the information is insufficient to reflect the complex interactions that are inherent in modern application architectures.

**ANALYTICS** refers to a set of automated techniques provided by an APM system in order to process and leverage performance and forensic data for service management purposes. These purposes are typically to monitor, alert, and troubleshoot all aspects of an application's performance and behavior in a production environment. Once data is gathered, analytics operate on the data to provide functions such as:

- Intelligent storage of information conducive to rapid retrieval during problem investigation or performance analysis.
- Derivation of new information to answer specific questions. Some, examples include: long term trends, peak operating periods, top consumers of resources, memory leak identification, and allocation of delay among application components.
- Adaptive baselining to establish "normal" behavior, and thus also uncover periods of "abnormal" behavior to support dynamic thresholding and alerting.
- Correlation across management domains in order to identify dependencies among system components, and uncover "cause-and-effect" relationships.
- Advanced visualizations synchronizing data across management domains in order to provide intuitive reporting, and promote sharing of information among service management teams.