

Solution Brief

Application Analysis

Introduction

Application Performance Analysis is the practice of assessing and profiling an application's efficiency, effectiveness and overall usability. Its purpose is to identify those areas that can impact an application's performance and end-user experience and to establish a baseline for future monitoring, SLA development and/or application reengineering efforts. Application Performance Analysis is not a final step, but rather the first step toward a continuous improvement process.

Background

Over the past several years, companies have invested heavily in network based applications as a means for driving revenue, lowering costs and minimizing risk to their organization. As a result, companies have significantly increased the dependency that they have on their application environment to perform even the most basic of business functions. An outage or even degradation is no longer just an inconvenience; it can literally stop the business in its tracks.

Today's enterprise applications are complex systems made up of multiple components that must each operate at optimal performance for the service to be effective and usable by an end-user. If any component within this environment including technology infrastructure components, application infrastructure components or application logic components are not performing as required, the service delivered to the end-user can be significantly degraded or often unusable.

Application Performance Analysis provides organizations with a means for measuring the effectiveness and usability of an application while assessing the performance of the individual components that make up that application. It should be considered an essential element of any application development, application monitoring or application re-engineering process as it provides both a reference baseline of performance as well as insight to the discrete areas that can influence that performance.

Analysis

An application's performance can be impacted by several factors, both internal and external. Understanding these factors and how they can impact an application's performance is the basis of Application Performance Analysis.

- **Utilization** - provides a detailed profile of what, who, when and how and application or service is being used. This measurement provides a baseline for further analysis as well as establishing an operating benchmark for any re-engineering efforts.

- **Responsiveness** - is the most commonly understood, but least measured area of application performance. It is the measurement of time it takes for a server based application to respond to a client request, often denoted as one-way or round-trip-delay. Many tools use the connection set-up sequence to establish this measurement. While informative, this approach can be misleading as it only provides a point in time view. To be most effective, responsiveness should be measured through the life of the application connection at a transaction level.
- **Connection Health** - measures the effectiveness and reliability of the Transport Layer (Layer 4 of the OSI model) services that provides the client/server control for higher level applications. Poor connection health can cause significant application degradation and is often misinterpreted as an application level problem. Often times poor connection health can be caused by poorly configured systems or intermittent network problems.
- **Consistency** - of responsiveness is extremely important to the overall experience felt by the end-user of an application. It is a key metric in understanding usability and extends the basis of responsiveness analysis. Denoted as jitter (either message or stream), it provides detail about how consistent the response from a server application is to a client request.

Summary

Application Performance Analysis can be a very powerful tool for ensuring that an application is operating at peak efficiency and effectiveness. It can and should be applied at all phases of an application's life-cycle from development through production.