Chapter 2
Performance Engineering and Management Method — A Holistic Approach to Performance Engineering

Dave Jewell

Abstract Experience has shown that there is no one “silver bullet” for achieving acceptable performance in IT solutions. Early performance models help us ask the right questions but may not be as accurate as we would like in predicting future performance and capacity utilization. Performance testing of the solution once it is built gives us more accurate information, but may occur too late in the life cycle to permit fixing persistent performance problems in a timely manner. The Performance Engineering and Management Method (PEMM), first proposed by IBM in 1998, integrates these and other techniques into the Information Technology (IT) solution development life cycle, yielding a more comprehensive approach to addressing the risks related to IT performance, capacity and scalability. This paper provides an overview of the major themes of PEMM, including examples of its application and potential synergy to be gained by combining PEMM with other disciplines such as Information Technology Infrastructure Library (ITIL®) Capacity Management.

2.1 Background

The Performance Engineering and Management Method (PEMM) was first formally outlined in March 1998, in the form of a reference document for IBM system architects. PEMM was based on the experience of practitioners in IBM’s United Kingdom organization. Many of the life cycle principles embodied in PEMM had already been used successfully during the 1980s, with associated technical papers presented at forums such as IBM’s Information Management System (IMS) Guide and Computer Measurement Group (CMG), and had been included in client education between 1988 and 1994.

Dave Jewell
Performance Engineering Practice, Systems Engineering, Architecture & Test (SEA&T), IBM Global Business Services, e-mail: jewell@us.ibm.com
The developers of PEMM had come to believe that the best way to address system performance was to proactively manage it throughout the solution development life cycle, from the system’s inception through its deployment and maintenance in a production setting. It has been IBM’s experience that throughout the last twenty five years; the principles within the PEMM have been long lasting and continue to deliver high value.

In an information technology (IT) context, performance generally refers to the speed with which a system accomplishes the tasks it was designed to do, and is commonly expressed using measurements such as response time and throughput. Performance is closely related to and often dependent upon capacity (the measured ability of a system to perform work) and scalability (the ability of a system to accommodate workload growth). The dilemma faced by the IT industry with respect to performance and capacity is at least two-fold:

1. While systems, the technology on which they are based and the demands placed upon them are leading to increased complexity and therefore greater risk of poor performance, software engineering as a whole has traditionally been focused on ensuring the functional correctness of those systems rather than addressing performance, capacity and scalability concerns.
2. Even when so-called best practices for performance are followed during a system’s development, it is difficult to grasp the full extent of inherent performance issues until all the components of the system can be integrated and tested to see how they perform together.

This has led to the situation where the industry has been working backwards over time to get a better handle on system performance. Poor performance in production led to the recognition of the value performance testing prior to deployment to identify and address performance issues. This in turn has led to focus on performance best practices, the notion of designing for performance, estimating and modeling in attempt to predict future performance, and creating requirements which unambiguously describe the users performance requirements and clearly link them to the needs of the business. The intent of the PEMM is to organize these activities in a cohesive manner, and to successfully manage performance requirements and mitigate risks throughout the software development life cycle.

---

1 Throughout this paper, system refers to an interrelated group of IT components working together; solution refers to the application of IT systems, products, etc., to solve a customers business problem and/or address a customers business opportunity.