Payoff

Benchmarking can uncover ways in which IT can add value to a business. Six key components for baselining the IT organization are discussed. The article's worksheets will help IT management create a baseline/benchmark snapshot of their organizations.

Problems Addressed

Benchmarking is a technique that companies use to compare themselves against other companies and identify best practices. In IT organizations, the use of benchmarking as a proactive tool for performance assessment and goal setting is rare but on the rise.

There are two fundamental reasons why this technique has not been widely applied in IT. First, benchmarking requires measurement, and few IT organizations have mature measurement programs. Second, there is a limited understanding of the key aspects of IT performance to be benchmarked.

One source of the problem is that measures in the world of IT are thought to be overhead and are not evaluated in terms of the value they add. In addition, those who have attempted to measure usually get caught up in trying to find the best measure instead of a suitable one. Finally, IT is under tremendous pressure to reduce expenses while producing more work. Most IT executives believe that to establish a baseline they have to impede progress before moving on again. If they slow down, they are unemployed. In short, baselining and benchmarking in IT are generally resisted for all of the wrong reasons.

Benchmarking as a Target-Setting Strategy

There are some documented cases of IT organizations that have invested the time and energy to focus their benchmark efforts by identifying key areas of performance, measuring them, and then comparing themselves against others using the same rigorous measurement criteria. Here are two examples that illustrate the results of benchmarking IT when properly applied.

Case 1. A bank's IT organization seeking to lower costs while strengthening the business alignment of IT.

Bank X focused its internal measurement efforts on characterizing the yield of the support costs of its work prioritization process application, the support process cycle time, the application quality, organizational structure, and new development productivity. By applying the same measures to a broad-based group of organizations with exemplary IT workflow management and support, the following findings were evident:

- Approximately 60% of the application enhancement requests being approved in the process had no business value.
- It was taking longer to approve work to be done than to actually do it.
- The organization had a hierarchical structure with seven levels; peer companies only had three layers.
Support costs were 40% higher than would be expected and applications were almost twice as defect-prone compared to equivalents at other companies.

Almost three times more staff was allocated to support the applications portfolio than was the average in the benchmark group.

Using the insights and observations obtained from the benchmarking project as a basis for setting IT performance targets, Bank X redeployed approximately 55% of its IT staff from support to new development while decreasing support cycle time by approximately 40%. The time frame for this radical change: 10 months.

**Case 2. A multinational chemical company wanting to become more responsive to environmental regulations while improving the quality of its applications portfolio.**

Company Y focused on its applications maintenance process complexity, productivity, and cycle time. Simultaneously, it assessed its entire portfolio from the vantage points of Technical Quality and Functional Quality. It also determined its current return-on-investment for its extensive suite of support and quality assurance tools. By doing a benchmark analysis with peer companies and vendors of packages in its industry, Company Y discovered that:

- Its maintenance processes were more serial in nature and had more bottlenecks than those in the comparison group.

- Technical quality of its applications was about 22% better than others, but functional quality was 35% worse. Customers were not benefiting from these technically high-quality systems.

- Tool penetration was low. The high technical quality of the applications could be attained with roughly 67% of the current level of effort if the toolsets were used as intended.

Within three months, Company Y had a program in place that reduced the cycle time of enacting a regulatory change from an average of four months to three weeks. In addition, specific functional quality targets were being set for each application. Skills upgrading and retraining of the staff was under way, with a target of increasing support productivity by 25%.

These two examples illustrate how benchmarking can be effective when the results are used for target setting and action. Benchmarking is not a passive activity, but one in which quantifiable findings must be turned into measurable gains. Benchmarking is also a continuous process. The two previous examples are simply snapshots taken as part of a program of continuous improvement.

**Benchmarking and the Link to Business Alignment**

Perhaps the biggest challenge facing IT organizations is to establish the link between their work efforts and business value. The starting point for making this happen is obtaining a total picture of the IT organization and relating it to the production of value for the business.

The key element in making the connection is the creation of an IT baseline - a quantitative view of where the IT organization is today - focusing on six basic factors that are the drivers of IT business value:
The applications and project inventory.

The IT organizational structure and human resources profile.

The management practices portfolio.

The delivery process methodology.

The technology infrastructure.

The business-IT interface.

Once complete, the baseline provides a framework for comparing an organization against competitive benchmarks and a context for improvement and innovation.

Among the criticisms of baselining is that it is costly, time-consuming, and involves too much introspection. Resistance to baselining, benchmarking, and measurement is quite widespread. Less than 10% of software-producing organizations worldwide has any kind of ongoing measurements program. Only one out of six organizations that start measurement programs is successful.

To be a success, a measurement program must be perceived by all to add value to the organization's processes and products, must supply information used for decision making and organizational learning (e.g., continuous process improvement), and must serve as a common basis for communication both within the IT organization and throughout the business itself. This definition of success fits with analyses undertaken by the Software Engineering Institute (SEI) of Carnegie Mellon University in Pittsburgh.

The SEI methodology classifies an organization's software process maturity into one of 5 levels. Level 1 is the Initial level, where there is no formalization of the software process; Level 5, by contrast, is the Optimizing level, where methods, procedures, and metrics are in place with a focus on continuous improvement. According to SEI studies, 80% to 86% of organizations in the US are at Level 1.

**Getting Ready to Benchmark**

The ability and capability to measure is a prerequisite for benchmarking. The first step is to carefully assess the IT organization's measurement readiness. Answers to the following questions can be a basis for setting the direction for the IT organization's measurement strategy; each answer is scaled from 0 to 5:

- How intense is the organization's desire to improve its performance? 0-No desire; 5-Intense desire.

- Is the enterprise willing to invest time and money to improve systems performance with measurement? 0-No; 5-The organization already allocates funds and people.

- What is the level of the systems skill inventory for using metrics? 0-None; 5-Already in widespread use.

- To what extent are measurement concepts known and understood by the systems staff? 0-No staff members have this knowledge; 5-Staff is 100% trained.
· Is the systems culture opposed to using measurements at the organizational and individual level? 0-100% opposed; 5-Eager to implement.

· To what extent is a support structure in place to foster measurement practices and perform metrics technology transfer? 0-No infrastructure; 5-An in-place team exists.

· Are tools and repositories for acquiring and analyzing metrics data in place? 0-No; 5-A full suite and warehouse are available.

· Does the systems organization understand its role in business processes? 0-No; 5-Yes, and it is documented and tracked through metrics.

Readers can chart their answers to these questions on Exhibit 1.

Measurement Readiness Profile

Actions for Beginners

If the answer to these questions is at the low end of the scale, the IT organization's measurement readiness is quite low. A good starting point is to contact professional societies so that experiences in measurement can be shared and exchanged. Readers can make contacts through the International Function Point User Group, the Institute of Electrical and Electronics Engineers (IEEE) Computer Society, the Quality Assurance Institute, the Gartner Group, and at seminars offered by Digital Consulting and Software Quality Engineering.

Actions for Improving Readiness

If the IT organization's readiness is somewhat higher according to the readiness profile in Exhibit 1, several actions should be taken in parallel.

First, IS management should reinforce the measurement infrastructure by acquiring automated tools to help with collection and analysis, then build a rapid baseline using the structure outlined in this article and produce a measurement program design. Next, it should embark on a 30-day mission to collect and analyze information using the 80/20 rule as a guide. The metrics findings for the baseline should be consolidated into a single page and should paint a picture of the six driver areas.

The baseline then has to be related to published benchmarks, translating the opportunities uncovered into business terms that will provide value from the efforts. This step brings focus to all the work and provides a platform for transformation and the ongoing use of measurement.

How to Build a Complete Information Technology Baseline

IT productivity and quality measures do not in themselves provide a complete picture of IT performance. Establishing a baseline as a prelude to benchmarking goes beyond just productivity and quality. A complete baseline involves assessing an IT organization's current portfolio of projects and applications, its human resources and organizational structure, its management practices and processes, the technology infrastructure, and most important, business factors that drive the computing function.
The typical time it should take to construct such a baseline is approximately 30 to 60 days with a dedicated team of no more than three individuals (including consultants). The goal is to create a workable and useful organization profile rather than accomplish 100% complete data acquisition the 80/20 rule applies here.

### Six Baseline Components

A divide-and-conquer approach is needed for assessing the baseline performance of an organization; this means viewing the baseline as containing six key segments that can be combined into a single comprehensive picture.

#### Applications and Project Portfolio Baseline.

The work performed by the software side of a typical IT organization takes the form of creating new software applications or modifications to existing applications. This baseline segment creates an inventory of the applications and current projects as they exist today as is. Key descriptive information and metrics for each existing system and project underway includes:

- Demographics (e.g., age, language, implementation date, technology platform, and tools and techniques used).
- Financial history (e.g., cost to build, cost to maintain, cost to use, and cost to operate).
- Size (e.g., lines of code or function points counts).
- Support information (e.g., number of people on staff, number of requests, and average request size).
- Quality attributes, such as:
  - Rating of Functional Quality by the user (i.e., the ability to support user requirements in terms of functionality, accuracy, reliability, and data quality).
  - Rating of Technical Quality by systems staff (i.e., design strength, complexity, architecture, maintainability, portability, and interoperability).
- Problem history.
- Defects found per line of code or function point.
- Productivity attributes, such as:
  - Support ratios (e.g., lines of code or function points per support staff member).
  - Original delivery rate (e.g., lines of code or functions points per team member per month).
**Systems Organization and Human Resources Baseline.**

This baseline segment provides a profile of the people side of the IT equation and the current organizational structure, including:

- Organizational chart (functional).
- Average managerial span of control.
- Human resources profile (e.g., skills inventory, educational inventory, training history, team and individual profiles such as Meyers-Briggs).
- Work distribution (i.e., percentage of people and dollars expended on development versus support).

Ultimately, this effort attempts to answer one question: Does the IT organization have the right resources to support the business today and into the future?

The production of an “organizational readiness” footprint, to determine the ability to assimilate new technology, is a major baseline output. This parallels the measurement readiness footprint used earlier (and shown in Exhibit 1) but concentrates on software technology instead of measurement. Assessment questions include:

- How intense is the organization's desire to improve its performance? 0-No desire; 5-Intense desire.
- How much is the organization willing to invest to improve its performance? 0-No investment; 5-Up to $100,000/professional.
- What is the current level of the systems skills inventory in software engineering? 0-Abstractions and models not used at all; 5-Formalization and models used by all.
- To what extent are basic software engineering concepts known and understood by the systems staff? 0-No staff members have been exposed to software engineering principles; 5-Staff is 100% trained.
- Is the systems culture adverse to using new tools, techniques, or innovations? 0-100% opposed; 5-Eager to implement.
- To what extent is a support structure in place to foster measurement software engineering technology transfer? 0-Not in place; 5-An in-place team of critical mass exists.
- What is the current software engineering platform? 0-Dumb terminals; 5-Client/server workstations.
- What is the development and support split? 0-0% versus 100%; 5-100% versus 0%.

The results should be plotted on a circular scale similar to the measurement assessment. This time, however, it is necessary to plot where the IT organization should be either in regard to a particular technology (e.g., client/server) or the overall software
process (e.g., Software Engineering Institute Level 1 through 5). The gaps that become apparent are those that need to be filled to transform an organization to where it should be.

**Management Practices Baseline.**

The focus of this baseline segment is on how the existing resources perform work. It means gathering and summarizing answers to basic questions about management practices:

- How are planning and prioritizing done?
- How does the organization translate requests into systems?
- How well defined is the systems development life cycle?
- What is in the organization's tool inventory and what is actually used (i.e., tool penetration)?
- What is the organization's current software process maturity level? (This entails performing a formal or informal SEI assessment.)

This information is typically gathered through interviews and workshops conducted to assess the rigor, actual end use, and effectiveness of these practices themselves.

**Delivery Process Baseline.**

This is the baseline segment in which specific representative development and maintenance projects are examined in detail to assess schedule and effort productivity and quality. However, it is often necessary to go beyond these issues and assess other factors likely to affect productivity and quality.

By determining delivery and support rates at the project level and comparing them to external benchmarks, an organization can create a framework to quantitatively assess the impact of potential changes and identify opportunities. Furthermore, the framework provides a clear basis for understanding the impact that tools potentially have on the overall delivery rate and product/process quality.

Typical metrics collected at the project level include:

- Lines of code or function points per professional by project size.
- Lines of code or function points per work month by project size.
- Defects per line of code or function discovered before and after implementation.
- Percentage of defects originating from each life cycle phase.
- Point in life cycle where defects were found.
- Percentage of defects removed by phase.
**Technology Infrastructure Baseline.**

This segment of the baseline identifies current and proposed delivery and production environments. If a company is, for example, wrestling with the possibility of shifting from mainframe to workstation-based development to reduce cost while increasing productivity, the practical steps that must be taken to accomplish this transition are evaluated.

**Business Factors Baseline.**

Perhaps the most important component of any baseline is mapping the link between the software engineering function and the business's performance itself. Executive interviews with major systems customers have to be undertaken to develop an understanding of the mix of internal and external factors that may be changing the business. Understanding the volatility of the customers' environment sets the stage for examining how the systems organization is aligned to support the business. In addition, interviews with key end users should be used to identify those projects that are not currently being worked on that could provide a measurable difference in the way the business is run.

The essence of this baseline segment is to discover how business value is created by the systems area from the vantage point of the business. This step also sets the stage for defining business-value-level metrics.

The results of this baseline segment should be used to create a table that separates all the people interviewed into peer group audiences. For each audience the cross-reference table shows which performance assessment areas are essential. Another table can then be constructed to link each performance area to the measurements that support it. The resulting set of tables is essentially the IT organization's measurement program design document.

**Making the Connection to Business Value**

The baseline process illustrates a metrics dashboard. With proper instrumentation, it can be used to monitor and manage organizational performance as well as clearly identify the system's contribution to business value.

The exact business measures can be derived by extending the business factors baseline segment. If external business customers and internal business customers are included in addition to the IT audiences, a complete dashboard framework will be the result, showing what performance improvement looks like to each constituency and what the suitable indicators (metrics) are. Using this as a framework, two types of dashboards should be constructed one containing the navigation gauges, the other containing the destination gauges used to declare success.

**Using External Benchmarks: A Step-By-Step Approach**

If the IT baseline is constructed following the guidelines given, it then becomes possible to assess IT performance against best-in-class benchmarks. Exhibits 2 to 11 are tools to help readers create a baseline/benchmark snapshot. The steps to construct a snapshot of an IT organization are detailed next.

**Applications Portfolio**
Using Exhibit 2a (lines of code) or 2b (function points), mark a point for each application for which a support ratio is computed per support professional. Using Exhibit 3, fill in the percentage of the total portfolio that is in each of the following categories:

- Low Functional Quality and low Technical Quality.
- High functional quality and low technical quality.
- Low functional quality and high technical quality.
- High functional quality and high technical quality.

Rating Development and Support Productivity

Application Portfolio Characteristics

Technology Infrastructure
Using the inventory of tools and techniques in Exhibit 4, indicate what percentage of the target audience is properly employing each tool and technique in the intended manner at least 80% of the time.

Tool/Technique Inventory US Data

Delivery Process
Mark a point on Exhibit 5a (lines of code) or 5b (function points) for each project that has been assessed in terms of delivered lines of code or function points per person month. Then, mark a point on Exhibit 6 that represents either the average number of postimplementation defects detected per function points or lines of code.

Rating Effort Productivity

Defect Density

Management Practices
Using Exhibit 7, place an X in the segment that most clearly relates the IT organization's Software Engineering Institute process maturity rating.

Software Process Maturity (US)

Organization and Human Resources
On Exhibit 8, mark the point that shows the average span of control, then answer the organizational readiness assessment questions and plot the results on Exhibit 9. Using Exhibit 10, indicate the percentage of resources allocated to maintenance and development.
At a more detailed level use Exhibit 11 to categorize the work as corrective, adaptive, and perfective maintenance.

Span of Control

Organizational Readiness Profits

Development Versus Maintenance

Maintenance Distribution

When used together, Exhibits 2 through 11 provide a snapshot of conditions in the IT department. Though this picture does not take into account industry specifics, IT managers can still use the worksheets to get value from their baseline through comparisons with the benchmarks provided.

Recommended Course of Action

Benchmarking clearly provides a context for assessing IT performance, business contribution, and competitive positioning. The real issue for IT organizations is how to transform performance and produce business value through computing.

In the context of benchmarking, metrics are a core competency that an organization must develop to promote learning and continuous improvement. The road to a learning organization is a difficult and complex one. It starts with the basics knowing where the business is, where it is going, and how it is going to get there. These issues are the essence of the benchmarking process.

IT organizations must transform their performance as business transforms. The benchmarking steps and worksheets in this article can be a tool and catalyst for making this change happen in the IT department.

Author Biographies

Howard A. Rubin

Howard A. Rubin is a professor at Hunter College of City University of New York (CUNY) and a Cap Gemini and Gemini Consulting faculty affiliate.
a. Using Lines of Code—Max/Min Observed Against the Average

b. Using Function Points—Max/Min Observed Against the Average
a. Productivity in Thousands Lines of Code (KLOC) per Staff Month
Note: With the exception of the Level 5 point, each square represents 10 companies.