IT Key Metrics Data 2023: Applications Measures — Framework Definitions

Published 8 December 2022 - ID G00779741 - 21 min read

By Analyst(s): Eric Stegman, Jamie Guevara, Nick Michelogiannakis, Shaivya Kaushal

Initiatives: Technology Finance, Risk and Value Management

This research outlines the Gartner Applications consensus model and framework definitions for IT cost management.

Key Findings

This document elaborates on the framework used by Gartner to compile the IT Key Metrics Data End-User Services Measures — Applications and the data structure a user should align to in order to accurately benchmark using either the published research or the self-service IT Key Metrics Data Comparison Tools.

- Ensure you understand the metrics before referring to the published research or using any of the available self-service comparison tools.
- Evaluate your organization by leveraging the available published content or receive a report tailored to your organization by completing the End-User Services & Enterprise Application Portfolio Budget & Efficiency self-service comparison tool.
- Follow the Practitioners Guide to prepare your data for comparison.
- Schedule an inquiry with a Gartner Expert to address alignment questions or to review your results and gain valuable insight based on your submission.

Scope

The scope of Applications is the provisioning and management of all business applications within an enterprise.

Application spending includes personnel spending as well as annual capital plus operational (cash out) spending for maintenance, installation and taxes, as appropriate for all non-personnel spending (i.e., application software licenses, software maintenance agreements, and development and support tools). This also includes third party or outsourced spending for application development and support activity and SaaS/PaaS based solutions.

It covers building new capabilities as well as supporting existing capabilities.

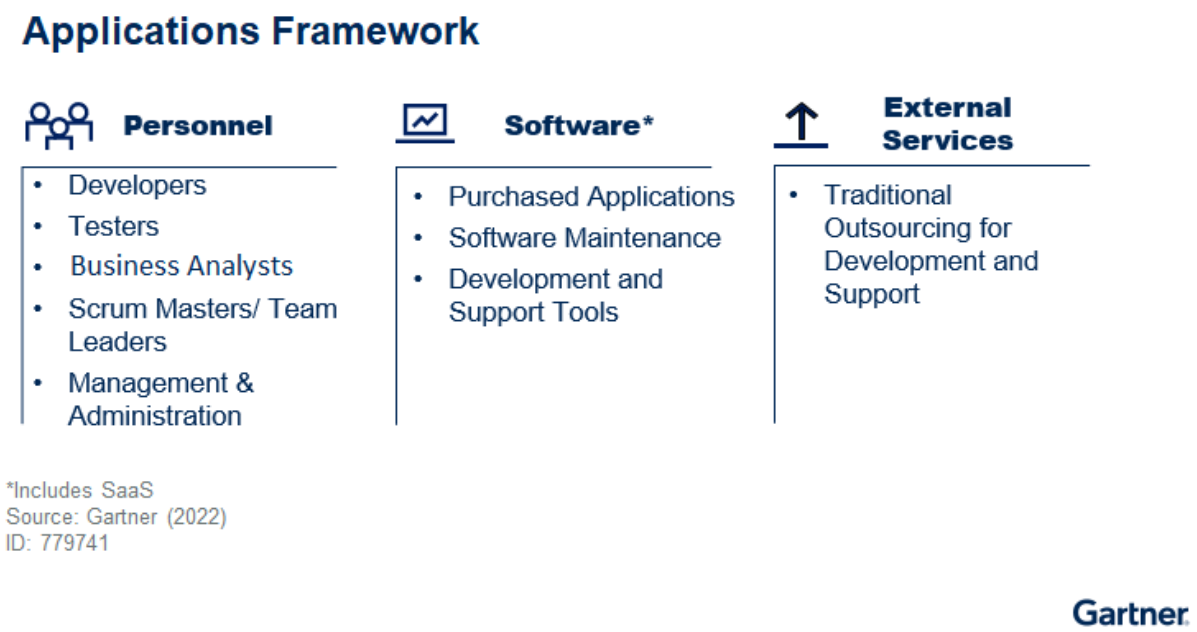
This model excludes costs for applications that are:

- Sold to external clients
- Providing external clients with third party digital content and/or analytics e.g., market data feeds, internet search, music streaming etc., social media platforms
- Embedded in other products such as automobiles

The model excludes purchased software licenses and software maintenance costs for database management systems, middleware, and applications that are:

- Associated with the management of IT Infrastructure and Operations e.g., Operating Systems, utilities, communications, IT service desk software, security software etc.
- Associated with workplace services, email, messaging, personal productivity, end-user collaboration, or end user content services.

Figure 1: Applications Framework



General Information

Business Metrics

The enterprise Revenue and Operating Expenses associated with the business units supported by the IT organization This section should be answered based on the latest 12-month actuals available.

Company Employees

The count of employees (i.e. headcount excluding contractors or consultants) on a full-time equivalent basis, regardless of whether these employees are frequent users of the technology supported by the IT organization. This includes full-time and part-time employees, or as reported in the public record.

Definitions for Total IT Spending and Staffing, Revenue, Operating Expenses, and Company Employees can be found in IT Key Metrics Data 2023: Industry Measures — Framework Definitions.

Applications Staff

Staff should be reported as full-time equivalents (FTEs). FTEs should be measured in calendar time. For example, an individual who works full time on an assignment for one full year would be reported as 1.0 FTE while an individual who was employed for six months of the study period would be reported as 0.5 FTE. Do not subtract such activities as vacation time, sick days and administration time. Do not count any one physical person as more than one FTE (for example, due to overtime). FTEs are assigned to services based on the functional definitions provided. If an individual or group performs more than one function, FTEs may be prorated between services and/or functions based on client estimates of time spent in each area.

Insourced FTEs are defined as FTEs who are employed by the IT organization (excluding contractors).

Contractor IT FTEs are defined as FTEs who are supplemental to your staff and are “operationally” managed by the in-house staff.

Outsourced Application FTEs are employed and operationally managed by an external organization. These are typically seen in application outsourcing agreements where the customer is billed on a time and materials basis. If this information is not available it can be excluded.

Service Definitions

Personnel

Annual spending for Applications internal staff and contractors includes salary, overtime pay, benefits and “other” employee costs such as job-related travel and IT training. The “benefit load” should include costs for bonuses, paid holidays, vacations, medical/dental coverage, life and accident insurance, retirement plans, stock plans, disability, Social Security, unemployment compensation, dependent care, tuition reimbursements and employee assistance programs (for example, physical exams, exercise programs and similar).

For contractors and consultants, include all compensation that was paid directly to the individual or agency.

Do not include the spending related to human resource department staff allocations, early retirement incentive bonuses and internal “cross-charges” for corporate overhead such as for the chairperson’s salary.

Roles and Activities

Personnel roles reflect the types of skills individuals bring to the organization. Personnel activities define the allocation of their time supporting various activities. For example a Developer or Tester may be involved in New Functionality Delivery as well as Business Unit Support. A Scrum Master or Manager may allocate their time to the work their team is doing. An Architect might be building a process to improve application maintenance.

Role Distribution

- **Developer:** This includes personnel devoted to developing new applications, and enhancing, maintaining, repairing, and providing business unit support for currently operational applications. These individuals may work on all phases of development including design, coding, implementation, deployment and the creation of user and system training and documentation materials.
- **Tester:** This includes personnel responsible for developing and executing the test scenarios and test cases in various types of testing including the identification of security flaws in application design, development, deployment, upgrade, or maintenance. Testers may use manual or automated techniques such as black box analysis, static analysis, scanning, etc. They are involved in test planning, writing test cases/scripts, test case automation and test execution. They also document all problems and assist in their resolution.

- **Business Analyst:** Business Analysts translate business ideas to implementable requirements. This function participates in strategic activities such as conducting research, assessing viable innovation opportunities, collecting necessary information for prioritization decisions, managing business expectations during execution, and measuring business benefit after deployment.

- **Scrum Master/Team Leader:** Scrum Masters are coaches (not managers) who are the primary facilitators for agile development teams. They are general troubleshooters and change agents who can perform tasks such as helping the business adapt to agile principles, and improving processes to maximize productivity. They are responsible for removing any impediments from the scrum team, and ensuring efficient communication between the Scrum Team and the Product Owner. They facilitate various scrum events (e.g., Daily stand up meetings, sprint planning meetings, sprint review meetings, sprint retrospective meetings).

Scrum Masters are a specific type of team leader. Any team leader or agile coach who performs these types of functions but doesn't have management responsibilities e.g., hiring, firing, performance reviews etc. can be included here.

- **Management & Administration:** Management personnel spend time on supervisory, departmental administration, or strategy related tasks. These tasks include but are not limited to setting strategic direction, communications activities, hiring and firing of staff, personnel performance reviews, expense management, approving relevant documents, planning day to day personnel workload etc. Time spent by managerial personnel on non-supervisory or departmental administration tasks (for example a supervisor who spends half his time engaged in projects) should be represented in the relevant category. It is not necessary to include any management time for any associate if the total time contribution related to the scope of the analysis of the individual (including non-management activities) represents less than 30% of the individual's total hours.

Administration provides direct administrative and clerical support to all organizations related to the service being studied. Typical positions include secretary, receptionist and administrative assistant. These individuals often work for executives at a high level in the organization.

Activity Distribution

- **New Functionality Delivery:** New Functionality Delivery includes the creation of new applications, and functional enhancements to current applications that take longer than your organizations functional enhancement threshold. If your organization doesn't have a threshold use more than 2 person-weeks as the dividing line between new functionality and minor enhancements.

Activities cover the full systems life cycle including analysis, design, coding, testing, communications, documentation, defect removal, quality management, and implementation and deployment of application software. Staff may make recommendations toward the development of new code or reuse of existing code. Responsibilities may also include participation in component and data architecture design, product evaluation and buy versus build recommendations.

- **Minor Enhancements:** The modification for an existing application that provides new functionality and takes less time than the organization's functional enhancement threshold. Includes design, coding, testing, communications, documentation, defect removal, quality management, and implementation and deployment of the enhancement.
- **Maintenance:** Maintenance encompasses upgrades and revisions to applications that are required by infrastructure, operating systems, vendors, etc. It covers associated analysis, design, coding, testing, communications, documentation, defect removal, quality management, and implementation and deployment of the upgrade or revision. This activity also includes routine maintenance of business rules, hard-coded data or tables embedded within programs, and minor upgrades/patches to vendor package/COTS software.

It does not include upgrades where the primary purpose of the work is due to new or changed business functionality, This would typically happen with a major release and the effort would be captured in new functionality development.

There isn't necessarily a limit on the size of an activity that can be considered maintenance, but typically larger projects will involve new functionality and be classified as new functionality development.

- **Break/Fix:** This includes defect repair for anything that causes the system not to function as designed. It covers associated analysis, design, coding, testing, communications, documentation, defect removal, quality management, and implementation and deployment of the defect repair.

- **Business Unit Support:** This includes assistance and ad-hoc consulting to users and business units regarding application functionality from the support team. Can include responding to requests for generating reports or sets of data, advice on changing or reconfiguring systems to improve performance/functionality, training of users, and helping users understand data calculations.

If these activities are handled through a service desk via a voice or chat channel, they would not be included in applications, but would go into the service desk category.

Software

Software includes Traditional and Software as a Service (SaaS) licenses, and Software Maintenance Agreements. The types of software covered includes Internal Business and Vertical Specific Applications, External and Digital applications and software that is used to provide business process outsourcing for external customers. It also includes development and support tools. Any software maintenance agreements around these applications included as well.

- **Purchased Applications — Traditional Software Licenses:** Perpetual, term, subscription, or open source licenses for application software. This does not include Software as a Service.
- **Purchased Applications — Software as a Service:** Application software that is owned, delivered and managed remotely by one or more providers. The provider delivers software based on one set of common code and data definitions that is consumed in a one-to-many model by all contracted customers at any time on a pay-for-use basis or as a subscription based on use metrics.
- **Software Maintenance Agreements:** These are contracts sold by vendors for support related to their software. The contract is usually calculated as an annual fee based on some percentage of the total software cost. It covers updating software, adding new functions, fixing bugs, and solving problems. Support may include telephone assistance time as well.
- **Development and Support Tools:** This includes software required by the application development and support staff members to do their jobs. It may include the following:

Languages/Compilers/Databases: This includes language products, compilers, debuggers, language libraries, language editors, PC/workstation languages used for development, fourth-generation languages with integrated databases, database products, database languages and PC/workstation databases used for development.

Development/Testing Tools: This includes application generators, prototyping tools, design aids, screen generators, process managers, integrated project managers, testing tools and repository structures.

IT Management Software: This includes any desktop software that is used by the development and support staff that is not included in languages or development/testing tools as defined previously. Some examples might include project-estimating software and stand-alone project-estimating tools/schedulers.

External Services

Fees incurred for any situation in which the full operational responsibility for IT services is completely handed over to an external service provider.

- **Application Development Outsourcing** Services for new functionality development that come from an external service provider.
- **Application Support Outsourcing** Services for functional enhancements, break/fix, maintenance, and business unit support that come from an external services provider.

There are several different models used to purchase external services.

- **Time and Materials** (Development and Support) In a time and materials contract, (T&M) the service recipient is billed for the costs incurred by personnel to complete perform services at a predetermined hourly rate. Any other incidental costs may also be passed on from the service provider.
- **Unit Pricing** (Development Only) These contracts may be based on understood workload factors such cost per function point where some industry standards exist. They can also be based on story points where the service recipient and service provider agree on how that is defined.
- **Fixed Price per Project** (Development Only) In a fixed price contract a fee is developed based on a defined statement of work.

- **Price per Sprint** (Development Only) Here each sprint as a mini-fixed-price project. Payment is made when working software is delivered, and it is subject to a strict "definition of done". The charges can be adjusted between sprints, if necessary, using a labor rate card to calculate the cost of increasing or decreasing the size of the service provider's team for the next sprint.
- **Business Outcome Pricing** (Development Only) Service providers are paid only when the client achieves the business value that the new system was intended to deliver. Service providers from the digital business and agile worlds are particularly likely to support such an arrangement. While this practice is highly desirable it is often difficult to implement, and many clients revert to fixed pricing.
- **FIxed Monthly Price** (Support Only) A single price is developed for a portfolio or applications
- **Price per Ticket** (Support Only) Pricing is based on a menu of different types of problems that need to be resolved.
- **Price per Application per Month** (Support Only) Applications may be classified (S/M/L) and with a price per month for each size. Some transparency in how charges change when applications are added or removed from the scope of the agreement.

Grouping Application Assets into Development and Support

The spending and staff components within the model can also be arranged into Applications Development and Applications Support. The detailed grouping methodology can be found in the figures below.

Figure 2: Applications Spending Grouped by Sub-Function

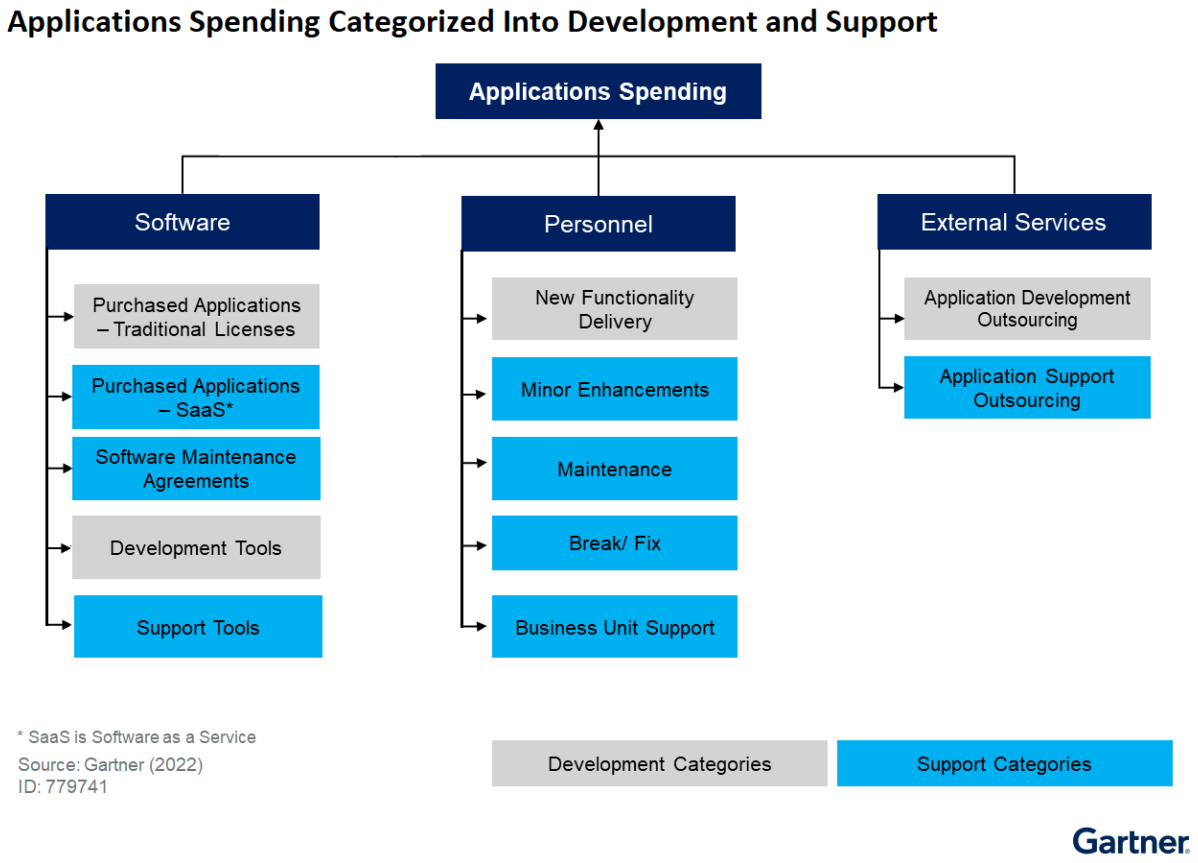
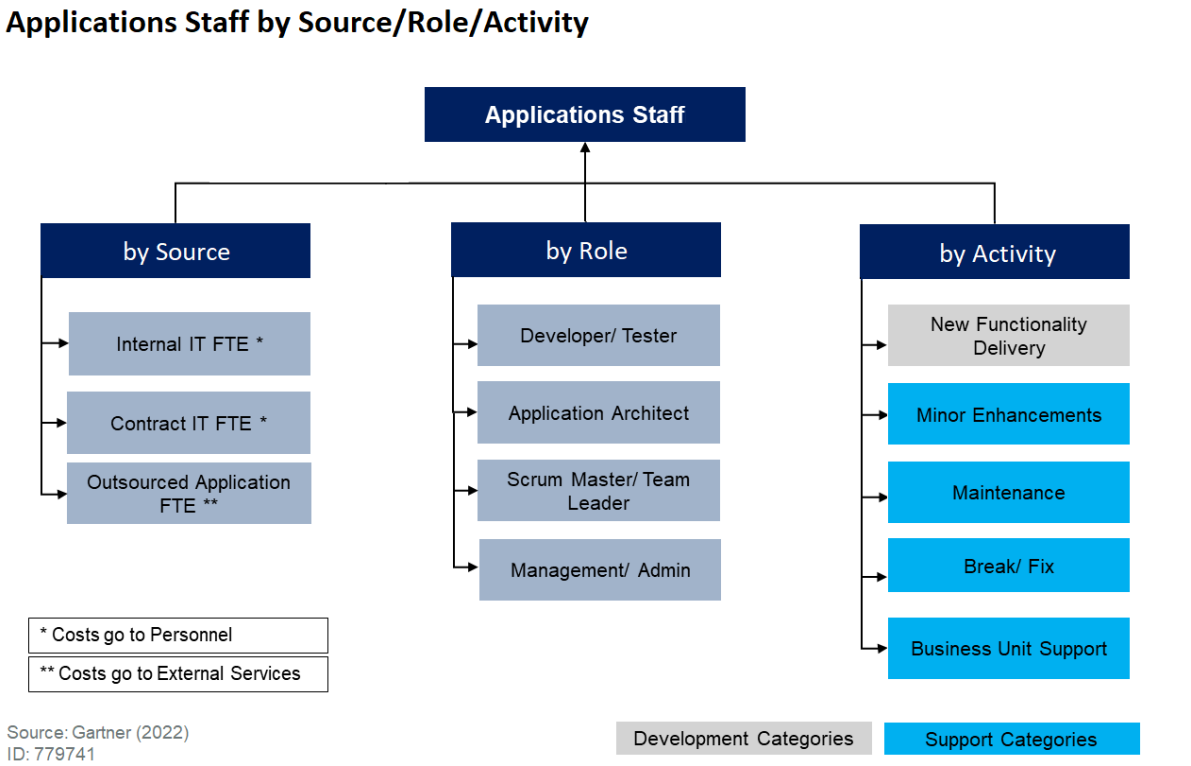


Figure 3: Applications Staff Distributions by Source/Role/Activity



Business Application Distribution

The business application distribution represents the types of investment the enterprise makes in building and maintaining internally developed or purchased applications.

- **Internal Business and Vertical Applications:**

These are applications designed specifically to support business processes or operations. They can include but are not limited to Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), Supply Chain Management (SCM), and other business applications that may be tied to a specific industry or business. Examples of other business applications include applications for insurance claims processing, electronic health records, power flow analysis for utilities, or clinical research applications.

It also covers business specific tools such as computer aided design tools to create blueprints manufactured parts or semiconductors. Additionally this category can include analytics applications that support derivation of insights, decisions and actions from data and algorithms that can automate discovery and action (such as Artificial Intelligence, Rule based systems, Machine Learning, Deep Learning, Big Data technologies)

- **Digital and externally focused applications**
  - Digital Applications allow for the operation of a business whose product set and business model are only made possible through information and digital technology. A digital business leverages a technology platform in order to generate net new revenue for the company that couldn't be possible in absence of the technology. Digital applications allow the business to create digitized offerings that complement traditional, non-connected offerings in the enterprise's product mix.
  - One example is an application that supports connected products such as appliances or large infrastructure items. Another example is a pay-as-you-use business. These businesses involve buyers getting the benefits of an asset without sinking resources into owning or managing it. Cases include electric light as a service for a large airport or pay as you drive auto insurance.
  - Externally focused applications. These applications allow customers to determine the correct configuration or customization of a product they are buying (e.g., size of a shirt, design on a greeting card, amount of insurance coverage, term of a loan, amount of inventory a business needs to meet demand). They can also provide customers with guidance in the proper use of a product or service. One example is an agriculture supply company application that helps farmers determine when to plant crops. Another example is an investment advisory company application that helps a client calculate their net worth by gathering data from their financial institutions.
  - This category does not include applications sold to external clients It also excludes applications that provide external clients with third party digital content and/or analytics e.g., market data feeds, internet search, music streaming etc., social media platforms. Additionally, applications that are embedded in other products such as automobiles are excluded.

Workplace Applications (Customization and Support): This category covers applications that are deployed primarily to improve employee productivity and engagement. It can include activities related to collaboration tools such as Sharepoint, and Slack, or management tools such as Service Now. This category only includes costs related to customizing, and supporting the customization of the applications (these are typically personnel and outsourcing costs). The costs for purchased software and software maintenance agreements for workplace applications are excluded from application spending as they are included in other areas such as digital workplace.

Business Process Outsourcing (BPO) Provider and Other Applications: Business Process Outsourcing Applications help BPO companies provide services to external clients but aren't sold externally. One Example is a case where a BPO provider uses contact center software to perform customer service activities for a client. Another example is a BPO provider using HR software to manage employee benefits for a client. Other Applications include applications that do not fit in any of the other categories.

Practices

When analyzing application spending we try to understand some of the very basic practices in place around building and supporting the portfolio. This isn't meant to be an extensive maturity exercise, but instead a way of highlighting possible opportunities for optimization.

Methodologies and approaches:

Waterfall versus other methodologies

Percent of Application Development FTEs workload involving applications using waterfall versus other methodologies

- **Waterfall Development:** In waterfall development each phase of development is discretely defined with clear gating criteria into the next phase, i.e., Analysis, Design, Development, Integration, System Test and Implementation.
- **Other Development Methodologies:** These typically include,
  - Iterative Development: In iterative development an application is developed in small sections called iterations. Each iteration is reviewed and critiqued by the software team and potential end-users; insights gained from the critique of an iteration are used to determine the next step in development.
  - Agile Development: This is not a single method; rather, it is a term used for a set of methods or best practices typified by XP, Scrum and DSDM. These methods have common principles that are summarized in the Agile Manifesto:
    - Individuals and interactions over processes and tools
    - Working software over comprehensive documentation
    - Customer collaboration over contract negotiation
    - Responding to change over following a plan

Product versus Project Approach

Percent of Application Development FTEs workload involving applications using a Product versus Project Approach

Project approaches to application development treat the work as a one-time effort with the aim of creating a product or service. It has a start and end date, as well as a defined result.

Product approaches to application development differ from a project approach as they don't look at it as a one-time effort. In a product approach the application evolves and adapts to business needs to prove its worth. Other characteristics are:

- Products are aggregations of business capabilities that are consumed by the business organization.
- The perception, from a business perspective, is at the user interface (UI) or groups of UIs level, which is linked to capability.
- The perception, from the IT perspective, is based on aggregates of functions used by a consumer, not on business capability.
- The delivery method (build, buy, cloud, software as a service [SaaS], etc.) of a product is inconsequential to the business, but important to IT.
- A product can be process-centric or information-centric.
- The business cares enough about the product to associate a budget for its purchase, enhancement and maintenance over a given life cycle.
- The product has a life cycle, from conception to retirement.

Contracting (Staff Augmentation) Practices

Using staff augmentation to handle peaks in workload or for access to hard to find skills quickly can be cost-effective. Using staff augmentation because of difficulties in hiring permanent staff can be costly and incur risk.

Citizen Development

A citizen developer is an employee who creates application capabilities for consumption by themselves or others, using tools that are not actively forbidden by IT or business units. A citizen developer is a persona, not a title or targeted role. They report to a business unit or function other than IT. Companies can harness the work of citizen developers to accelerate digital business while saving on IT costs.

Open-Source Applications

Open source describes software that comes with permission to use, copy and distribute, either as is or with modifications, and that may be offered either free or with a charge. The source code must be made available.Application leaders may sometimes replace existing applications, tools and platforms with self-supported OSS, as a cost-cutting strategy. This means relying on your own resources for technical and legal support, and turning to an OSS project community for additional support.

Remote Applications Workforce

Hiring talent irrespective of location can expand the talent pool Sourcing talent from less competitive regions, where high supply and low demand can lower the cost for hard to find skills.

Diversity Equity and Inclusion

Making IT an equitable place for all to work, they will have a hand in enhancing their company’s brand especially when it comes to attracting, recruiting, and retaining much needed talent. Remote working makes it possible to hire highly skilled people who cannot commute to an office. Examples include people who tend to family members and people with disabilities who bring untapped potential to bridge the skills gap.

Insight Into the Applications Portfolio

In order to optimize resources, enterprises must be able to understand the nature of the assets in place. The items below represent some key characteristics of the application portfolio. For this set of questions we aren’t asking for actual numbers, but if you can or do track them. For each area we have three options to describe this.

Little or no insight - This means while it may be possible to get this data it would be a very manual and time consuming exercise.

We have the ability to gain insight with some effort, but we don't track formally - This means that while it isn't formally tracked the basic artifacts or systems needed to develop the data are available and could be leveraged with a reasonable amount of effort.

Tracked Formally This means that the organization has this data and uses it to analyze and make decisions around their portfolio.

Portfolio Characteristics

Age of Applications

The age of an application is defined as the difference in time between the date the application was first launched and the current date. The average age is the sum of the age of each application divided by the total number of applications.

Redundant Applications

Redundant applications is the situation where more than one application can be mapped to the same business capability.

Technical Limitations

Technical limitations occur when the technical stack or platform underlying an application is either not fit or at risk of being not fit to support it. This can include the associated hardware, operating system, database and the development languages and techniques used to create it.

Business Limitations

Business limitations occur when an application is unable or at risk of being unable to support business capabilities or processes that users need around:

- Support: How well does it improve productivity of employees or customer users?
- Agility: How well does it support changes and anticipated process or capability requirements?
- Completeness: How well does it generate the desired results without the need for overrides or manual intervention?

Service Level/Business Requirements mismatches

Service Level/Business Requirements mismatches occur when an internal or external service level agreement or actual service levels delivered is different than what is actually needed. An example of this is 24/365 support on a system that only has use Monday to Friday.

Software licensing levels mismatched with usage

When software licensing levels don't match usage Application leaders either risk the unbudgeted costs that can come from a vendor software audit, or incur wasted spending for assets that aren't used.

Gartner Recommended Reading

Some documents may not be available as part of your current Gartner subscription.

"Toolkit: The Application Leader's First 100 Days"

"Cost Optimization and Productivity Strategies in Application Services for Short- and Long-Term Returns"

"How to Choose a Sizing Framework to Measure Productivity"

"How to Prioritize Application Inventory and Rationalization — By the Numbers"

"How to Assess the Fitness of Your Application Portfolio"

Evidence

This research contains relevant industry standard consensus model and IT performance measurement framework as defined by Gartner Benchmark Analytics. To learn more about Gartner Benchmark Analytics contact your account executive or email us.

Document Revision History

IT Key Metrics Data 2022: Applications Measures — Framework Definitions - 16 December 2021

IT Key Metrics Data 2021: Applications Measures — Framework Definitions - 18 December 2020