Gartner Research

Publication Date: 29 July 2009 ID Number: G00168947

The PMO IT 'Control Tower': Filing a Flight Plan

Matt Light

The project management office's (PMO's) role in guiding IT projects from inception to completion is much like that of a control tower. An effective PMO administers the project-chartering process on behalf of a more senior IT governance board (ITGB) with prioritization authority. It also has a role in selecting the "flight path" for projects, in collaboration with the architectural team and project manager.

Key Findings

- PMO responsibilities may overlap with others' responsibilities, so effective collaboration calls for communicating responsibilities and authority to rationalize agendas and eliminate friction.
- IT projects differ in degree, but also in kind, so that PMO control towers provide from five to 10 main paths that their software (and other) projects can follow.

Recommendations

- The project-chartering process should be administered by PMOs in support of a more senior ITGB with prioritization authority.
- A PMO should be engaged with an architectural review board to drive toward future-state architectures and apply appropriate project methodologies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

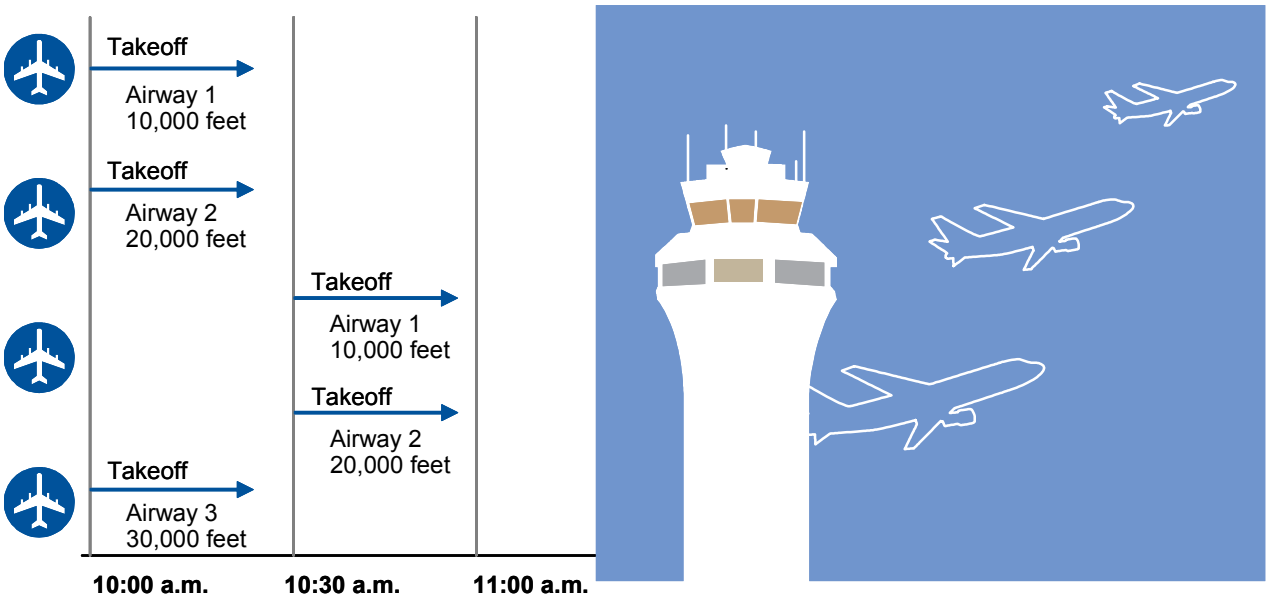
1.0 The PMO's Charter: Safe Takeoffs and Landings

IT organizations have many ongoing projects "in the air" at once, and have many project-management-related issues that are beyond the scope of any single project. In some ways, an IT PMO supporting projects has much in common with an air traffic control tower (see "Project Management Office: The IT Control Tower").

Like an air traffic control tower, the IT PMO guides individual projects through project chartering, coordinating each project's "takeoff" and "landing" times and deliverables. The PMO also monitors to proactively avoid project disasters. Such "radar warnings" provide project assistance to help project managers reliably meet quality requirements, deliver when expected and hit budget targets — all while also monitoring other ongoing projects simultaneously.

Administering the project chartering process is a key activity of a Level 3 PMO that seeks to improve project and program selection and, thus, IT alignment and performance (see "The 'Pretty Good' PMO at Maturity Level 3"). The PMO's project charter template should structure the information for prioritization — such as business goals, customer impact, competitive drivers, initial cost-benefit analysis and other data.

Figure 1. The PMO Control Tower: Different Flight Paths for Different Projects



Source: Gartner (July 2009)

1.1 Project Chartering

Coordinating the project-chartering process is typically accompanied by various degrees of program oversight and project control. An inquiry we often get has to do with project cost estimation, which is a key input to prioritization that is based on cost-benefit (and risk) analysis.

Approval of project ideas for data-gathering purposes should not be taken as carte blanche approval of "project takeoff" — that is, launching a project that will occupy perhaps dozens of staff

for many months. Approval of project takeoff depends not just on the preliminary idea's hoped-for returns, but also on the more considered review of those returns' likelihood of being realized (risk) vs. any necessary capital outlays and staff-months to perform the project tasks needed to achieve success.

This detail about risks and costs is generally not part of the exploratory project proposal (although terminology such as "charter," "proposal" and so on differs). It is typically gathered into a project charter, after the initial proposal's approval, for data-gathering purposes. Then, prior to takeoff, the project charter should be reviewed and approved to authorize capital outlay and significant staff allocation.

A project charter is a contract between senior management sponsors of a project and those who will manage its delivery. It is a "living document" and subject to change; however, significant changes must be approved by key stakeholders.

1.2 Elements of a Living Charter

A project charter issued by senior management formally authorizes the initiation of a project and gives the project manager authority to apply organizational resources to project activities and tasks. It includes a description of business needs and how the project deliverable will address those needs. Although limited to a high-level overview in its initial form, the charter should provide sufficient detail (or refer to a previously approved business case containing such detail) to perform these actions. It should:

- Show alignment to enterprise strategy, goals and priorities.
- Illustrate how the project deliverable will meet specific business requirements.
- Establish clear success criteria based on measures of client satisfaction.
- Identify funding sources and high-level costs.
- Identify qualitative benefits and translate to quantitative measures.
- Establish appropriate intervals for updated cost estimates.
- Examine assumptions (for example, system performance or skills needed).
- Identify risks and prepare mitigations.
- Factor risks against cost-benefits.
- Select the project's guiding principles and methodologies.
- Communicate control mechanisms to stakeholders.

1.3 The PMO and Decision Making

If governance decision-making structures are ill-defined or otherwise dysfunctional, then the prioritization process for "IT-heavy" projects tends to migrate into the IT organization, and even into the PMO per se. This is little better than no governance process at all, and is unlikely to optimally decide among business trade-offs.

No matter how thorough the project-chartering process, or how insightful a PMO's understanding of project proposals, the decision to outlay significant capital expense or to allocate perhaps dozens of staff for many months is generally "above its pay grade." In most enterprises,

prioritization decisions among the many competing demands on IT are decisions for more senior business management.

The PMO should be involved with planning and controlling strategic projects, but it is principally charged with tactical execution (not strategizing — including prioritization).

1.4 Coordinating the PMO With the Governance Board

For newly proposed projects, the PMO may prepare an analysis of alignment and trade-offs, but for these higher-level, strategic decisions, there is the ITGB.

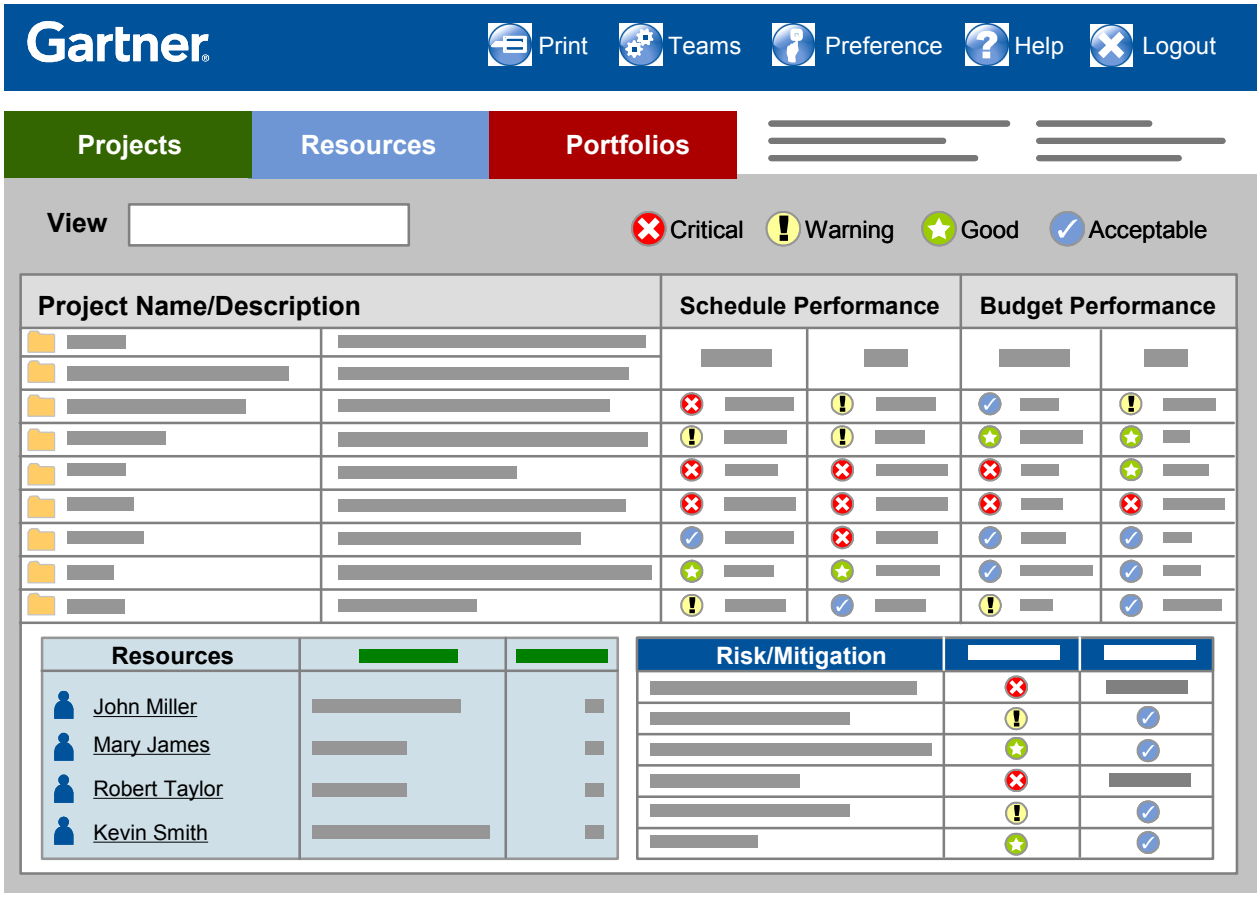
A properly constituted ITGB is composed of members primarily at the executive level who meet regularly (monthly or bimonthly) to make resource decisions, track progress toward project objectives, resolve conflicts and, when necessary, terminate failing projects. An essential function of a governance board is to direct and review the PMO, which in turn provides analysis of project portfolio issues with prioritization, remediation and other recommendations.

Making such decisions requires good, consistent information. However, different business units and functional areas may have different formats and metrics for monitoring their work. Prior to board meetings, the PMO prepares a project portfolio "dashboard" summarizing such project data as updated schedules, progress reports, cost variances and quality — or other issues, such as bottlenecks and trends, as well as delivery assessments for the upcoming month and quarter. (see "Toolkit: Project Dashboard Template").

However, in working with multiple business units (and even subunits of IT), different units often have different types and forms of status information. One functional area might track costs against budget, but have little sense of schedule progress. Others may track a project's completion percentage by asking the project manager for a rough estimate, while others may track completed tasks against the number of planned tasks. Still others may only consider milestones on the critical path.

The goal of the dashboard is to provide a realistic, overall picture of the entire project portfolio so the governance board can make fast, accurate decisions. If failures in key projects threaten an initiative, or if resources are clearly insufficient for the task, then this should surface early in the dashboard.

Figure 2. Sample PMO Dashboard



Source: Gartner (July 2009)

If nonstandard data in inconsistent formats can contribute to a realistic, overall picture, then the dashboard can certainly summarize some of this range of heterogeneous data, with backup available in multiple formats. However, the PMO should also strive to include key metrics in consistent, standardized form and should, over time, drive toward a set of common, comparable project metrics.

The PMO plays an important role in governance: It acts as a conduit for project proposals and as an "early warning system" for projects in trouble. When the governance board makes decisions (approving a project scope reduction, the addition of resources to another project or cancellation of a project altogether), the PMO communicates this information back to the project managers and other affected stakeholders.

In addition, when significant new problems or opportunities arise, the PMO should provide recommendations — complete with dissenting opinions — on the next steps. (Less-significant problems or opportunities, as well as actions that the PMO has taken since the last governance board meeting, may also be listed, with additional analysis or backup provided on request.)

The PMO normally also supports the governance board by handling the logistics of meeting scheduling, circulating reports and contributing agenda items.

1.5 Coordinating the PMO With the Architectural Review Board

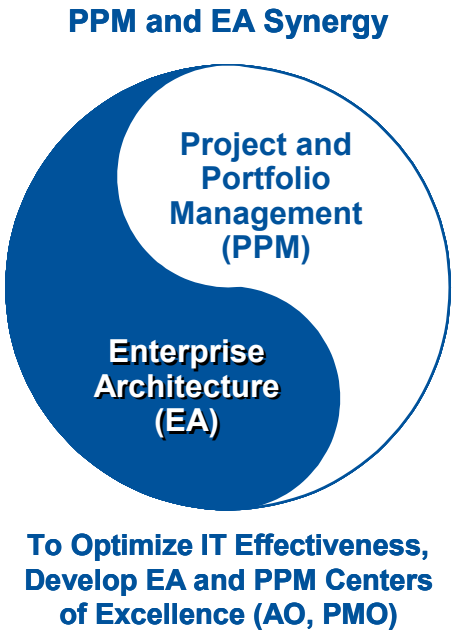
Project managers often view the notion of enterprise architecture (EA) with suspicion — as something that will constrain project choices and delay delivery. However, a well-developed architecture provides a framework to enable faster delivery of more integrated and interoperable, standardized, and longer-lived solutions that are also easier to support full life cycle. And, because the majority of projects usually don't report to the enterprise architecture group, many project decisions are made without regard to architectural goals.

Progress toward architectural goals occurs in two ways. There are, of course, projects to rearchitect or replace systems that do not fit the desired "future state" architecture (or may actually impede progress toward it). These projects have architectural progress as a main goal and can be characterized as top-down architecture projects. However, there are also bottom-up architectural efforts, where architecture is a consideration of projects with other primary goals.

Incremental steps taken, one at a time, as projects make many of their own architectural choices, generally contributing more to the future state than would be the case with top-down EA efforts (which often are challenged to provide short-term cost-benefit justification).

Figure 3. PPM and EA Synergy

- EA and PPM are needed.
- EA provides a leverageable foundation for PPM and the PMO:
  - Both are driven from enterprise strategy.
- PPM is focused on managing change and project delivery:
  - PMO provides project oversight.
- PPM links abstract EA to real projects:
  - Directly integrate EA activity into the project methodology (or software development life cycle process).
  - EA “closing the gap” must generate new project ideas for PPM to manage.
  - Real EA value is driven only from real project success.



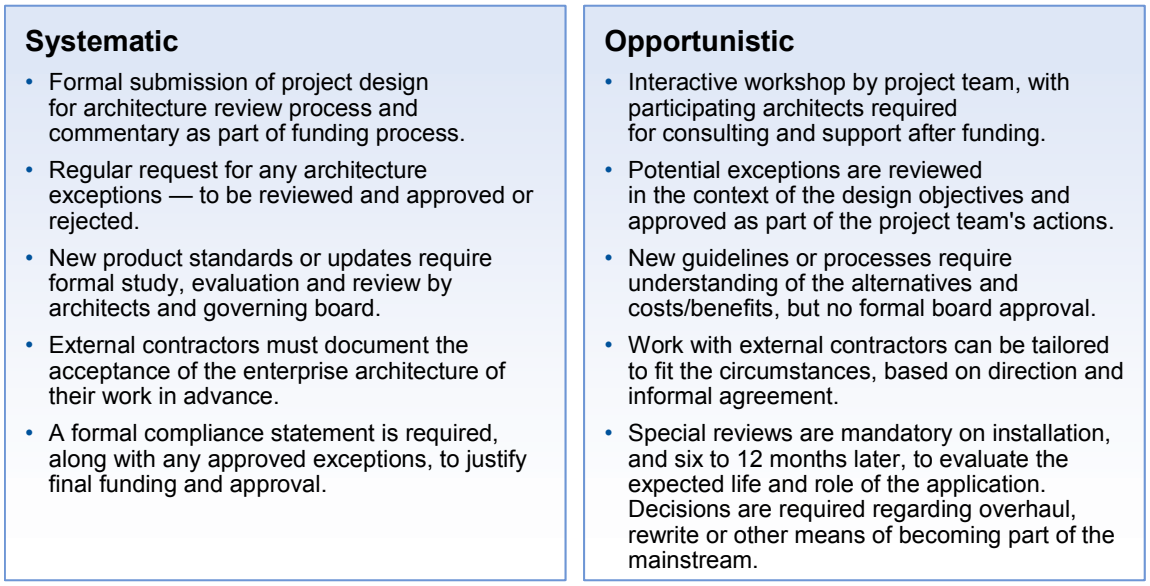
Source: Gartner (July 2009)

Because the PMO is in a central position collecting the project charters for review, it is effectively positioned to coordinate reviews of proposed projects' architectural components by the architecture review board (ARB) as necessary. The review will help the PMO select the type and level of methodology suited to the project, identify areas of possible nonintegration and their consequences — if any — and generate opportunities for reuse.

These reviews are needed mainly when the deliverables are expected to have a long life and to interface significantly with other systems as part of the enterprise's overall "software ecosystem." These "systematic" projects are subject to a different level of architectural review and rigor than

more stand-alone or "opportunistic" projects, with the ARB getting answers as to how the project will address the EA.

Figure 4. Architectural Review Requirements: Systematic vs. Opportunistic



Source: Gartner (July 2009)

The ARB is responsible for the EA waiver approval process, and the PMO director often is a member, working with the architectural team to ensure that PPM and EA leaders are aware of each other's work. EA responsibilities should not, however, be handed off to the PMO.

2.0 Identifying the Right Flight Paths for IT Projects

Quite apart from before-and-after activities, and keeping the dashboard current, the PMO has a role in ensuring projects meet or exceed client expectations. To deliver a good project, a prerequisite is to understand what "good" means — namely, what is "quality" for a given project:

- Is it immediate delivery of a few key functions (with future versions to improve and extend)?
- Is it delivery of a range of several well-integrated and 99.44% bug-free functions?
- Is it delivery of as much "good enough" function as possible within a tight budget?

Answering these questions with business users helps organizations understand what "quality" their quality management methodology should pursue (see "The First Key to Project Success Is Collaborative Requirements Definition and Management").

The PMO has a key role in this. While administering the chartering process and coordinating with the ITGB, it also helps determine the delivery methodology by which projects will proceed — establishing a project's "flight plan," to extend our previous analogy. This is done in close cooperation with the ARB, and with the appointed project manager (whether that project manager is from the PMO, from another part of the IT organization or from a business unit).

The method is distinctly different from the control framework — perhaps based on Project Management Body of Knowledge (PMBOK) or Projects in Controlled Environments (PRINCE2) —

that is reflected in the dashboard. The method will be mainly conducted by the project manager (not the PMO) and should be flexible to the given project at hand. *Through 2013, PMOs tying a flexible range of rich project methodology options to the project-chartering phase will have twice the survival rate within two years of their introduction, versus those with single or simplistic process architectures.*

A PMO — because of its early involvement in the chartering process, and its role as a repository of best practices — is well-positioned to provide guidance on selection and tailoring of a project's flight plan. An airport's control tower must address such questions as: Should the airway be at 5,000 feet, 20,000 feet or 50,000 feet? Should we use a northern airway to avoid thunderstorms, or should we brave turbulence?

A PMO acting in a similar capacity must address such questions as: Should the project plan weekly tasks for individual team members, to enable progress tracking in granular detail (for example, a single percentile at a time) — or should planning be on a team activity basis? Should the project avoid scope volatility via structured analysis and requirements specification, with formal management signoffs to any change, or should it accept the turbulence of such agile approaches as scrum? Through communication and collaboration with application delivery management and the EA group, the PMO can address these questions at the right point in the project life cycle — at takeoff.

2.1 Staying on Course

Staying on course along the flight path, which involves the mechanics of steering the plane, is the pilot's and not the control tower's responsibility — and conducting an IT project, whether according to a structured "waterfall" methodology, or an iterative "spiral" methodology, or an agile "NeoRAD" methodology, is up to the project manager. However, making the initial determination of the method has implications beyond the project itself. Therefore, other stakeholders also take part in that determination, with the PMO as an arbiter and advisor.

In defining and elaborating different methodological "airways," the methodology experts, senior IT project managers, PMO and other stakeholders should keep the method's utilitarian nature in mind: A method is mainly meant to be useful during the project execution itself (not before or after for justification or review), and of primary use to the project manager and the project team, not just something at the beginning or end.

Because it is primarily for the project managers, definition of the organization's core methodology's different routes — if they have not been defined — should include participation and even approval of the key influencers among the organization's experienced project managers — whether they are part of the PMO (or another process group). The acceptance or buy-in of these influential stakeholders will be key to successful adoption and ongoing use of the methods.

If the core routes are largely in place, then these same key influencers should be involved in the ongoing kaizen, or continuous improvement, of the method and its different routes. The team should ensure that the organization's current good practices are reflected in the methods, and that new practices (for example, service-oriented architecture testing) are captured and incorporated in the methods.

2.2 Best-Fit Methodology

Gartner endorses the common-sense notion of a flexible and configurable approach to process and methodology. Unfortunately, as the saying goes, "common sense isn't so common." Most IT organizations have some practitioners that are reflexively antimethod, to whom "flexibility" and "configurability" mean the organization's methodologies are optional, are used randomly or get whittled down to nearly nothing.

Our recommendation of flexibility and configurability should not be abused by the method-averse organizations that permit carelessness in their projects. Nor should the need for flexibility and configurability be ignored by those who view a method as a foolproof series of steps that always lead to success (they don't). Training on intent, proper use and potential misuse of methods is essential.

Many IT organizations have essentially a single methodology, based on a structured analysis or "waterfall" approach to project delivery, which they then attempt to scale up or down to fit different-size projects — perhaps adding a few design or test checkpoints. In our air traffic analogy, it is as if every flight from an airport followed the same path in a different-size plane.

IT projects are not just different from one another in degree; they are also different in kind. Thus, IT organizations generally benefit from having between five and 10 main flight paths that their software (and other) projects can follow. These should change over time.

Often, this need results in ad hoc adaptation of the single, structured methodology to fit different project realities. More-capable organizations define several different core methodologies, and the PMO control tower helps identify which method a project should follow in the chartering phase, which the PMO helps administer.

2.3 Application Project Methodologies

Only careful analysis of the main attributes of a project will reveal what kind of method to use. Some of these attributes may be identified by answering such questions as the following:

- Can the requirements of a proposed project be reasonably expected to be stable, clear and bounded?
    - Can the requirements be met with a commercial off-the-shelf (COTS) system?
    - Will such a project be done in-house, or will it be handed off to an external service provider if internal resources are insufficient or not appropriately skilled?

    By contrast, does initial analysis of the proposed project suggest somewhat less clarity and "boundedness," although still finding that fewer than 25% to 30% of the functional requirements have yet to be discovered, or are "fuzzy" and likely to shift?
- Are requirements very open-ended, with perhaps more than 35% to 40% that cannot be well-understood until further into the project?
    - With requirements open-ended and unclear, are key dependencies apparent, with major functional areas representing substantial changes to a business process or system?
    - Are there no key dependencies, and do the projects' features represent mainly incremental improvements to a business process or system?
    - Is high quality an immediate requirement, or is quick-and-dirty functionality sufficient, with improvement later?

Waterfall vs. COTS. When requirements are stable, clear and bounded, the PMO might reasonably recommend the use of a traditional "waterfall" methodology (structured in five sequential phases, from requirements to design, implementation, verification and maintenance) — although conventional execution of the method has often been poor. However, when requirements are that stable and clear, the software industry — over the course of the past 20

years and more — has increasingly provided COTS alternatives, which are usually more cost-effective alternatives for commodity (or near-commodity, customizable) applications.

Package implementation. COTS software requires a different methodology for evaluation, acquisition, implementation (see "Using Application Selection Guidelines to Reduce Risks") and, perhaps, customization.

Do not, however, underestimate the effort, or trade-offs, of implementing a COTS "80% solution"; the Pareto "80/20" principle often pertains — so that the last 20% of the functionality actually takes 80% of the effort to implement. (Working with a system integrator familiar with the package may or may not cost less.) For noncommodity systems, if you cannot provide that last 20% of function (for example, in a software-as-a-service solution), then you may be trading off 80% of the value.

Earned value management (EVM). If commercial alternatives are unavailable, then the solution might still be externally sourced if the internal development group lacks capacity or suitable skills. A senior project manager out of a PMO should often be charged with collaborating with the contractor's project manager in regularly verifying progress against the project's schedule in keeping with budgeted effort and any payment outlays (such as EVM, which can be implemented with varying levels of rigor).

In such cases, a waterfall-style methodology shared with the contractor might still be preferred, particularly if budget baselines must be solidified early in the life cycle (that is, after an analysis project). If requirements are stable, clear and bounded, then they form a good basis for fixed-bid contracts on projects to be developed mainly by external service provides. Too often, however, such contract projects run into difficulty because of vague or shifting requirements. Waterfall methodologies have also been shown to be decreasingly effective as project effort, duration and complexity increase.

Iterative (spiral). When only about 70% to 75% of business requirements (as distinct from technical specifications) seem readily definable, clear and stable, then a more iterative (or "spiral") methodology will likely succeed where a "waterfall" method would fail.

Many IT organizations do not have or use a distinctly iterative method for their larger projects — instead labeling as "iterative" their (sometimes interminable) series of mini-waterfall projects, although they often still attempt to specify nearly all requirements before that is feasible.

Often, when a large deliverable of complex and integrated functionality is likely to exceed, for example, 10 person-years of effort, many of the requirements are not readily definable early on (depending on other complexity factors, such as the number of user types and use cases). Often, a prerequisite of such medium-to-large projects is to address certain key dependencies in an early iteration, perhaps the first iteration. Security, scalability, architectural, performance or other requirements may loom — system dependencies that may not address users' immediate functional priorities but, if neglected, could cost much time and effort in refactoring later in the project.

Truly distinct iterative methods typically involve creating a prototype that addresses such dependencies and most of the known requirements in the first, approximate, iteration. Subsequent iterations address strengths, weakness and risks of the first, and further extend the solution's functional reach. Mini-waterfall series are little informed by this approach.

Increasingly, such projects are also able to leverage design patterns and frameworks and architected RAD (ARAD) tools for efficiency (see "Trends in Model-Driven Development, 2H08 to 1H09"). IBM's Rational Unified Process is a good example of an iterative or spiral method.

Agile (NeoRAD). Similar in concept to the original rapid application development (RAD) approach popularized in the 1990s, agile methods go further in accelerating feature deliveries — sometimes called "sprints" — from the three- to six-month time frames typical of RAD, to monthly or even weekly delivery. In both, a small, collaborative team dedicated wholly (or almost so) to the project works together, in one location, on the project.

By 2011, more than a third of large AD organizations will succeed in blending a NeoRAD approach with such other approaches as waterfall, model-driven and iterative in a flexible software process architecture to yield continuously high quality and customer satisfaction ratings (see "Pairing Agility With Quality: Gartner's 10 Principles of NeoRAD").

Agile methods are sometimes characterized as "voyages of discovery," in that they constantly adjust and fine-tune the project as they proceed, rather than sticking to a plan — that is, they are "empirical" rather than predictive. Requirements are defined as "features" or "user stories." Agile projects cycle the delivery of functionality to users monthly or faster — each cycle's objective being immediate added value via new or improved features, which are put into production as soon as possible after each cycle, both to provide value and to collect empirical information about their usefulness and quality.

Agile methods fit best when business and functional requirements are relatively open-ended, with perhaps more than 35% to 40% that cannot be well-understood until further into the project — and when features represent mainly incremental improvements to a business process or system (not requiring frequent, major change). Typically, agile projects do not immediately face critical, non-feature-driven dependencies, or can afford the refactoring to address them later, after the delivery of key, priority features. They do not seek to address several integrated and complex requirements in a first, lengthy iteration.

Agile projects are characterized by highly collaborative, adaptable teams that include users throughout. Requirements are captured in user terms. In one approach, for example, an "action-result-object" format is used — for example, "validate account user" would be described as: "action = validate; result = valid or invalid; object = user." The dynamic system development method is a good example of an agile method (see "Agile Essence: Dynamic System Development Method").

The small teams and short iterations of agile, however, may not be able to deliver the full range of functionality quickly enough to satisfy customers and, because agile methods stress colocation and face-to-face communication (including with users), an agile approach is less well-suited for distributed development, and coordination of large teams on concurrent efforts generally requires more design (to ensure concurrent deliverables converge) and more formal communication than is typical of agile. In addition, agile methods tend to eschew written documentation ("the code is its own documentation"), so that significant effort catching up on documentation as the agile effort winds down may be needed to support future training and support needs.

Agile maintenance vs. release management. In some cases, systems that were not initially delivered via an agile process have been successfully switched over to agile maintenance methods. When many important features remain to be rapidly delivered or improved, but problems prevented the use of agile on the system's early versions, a colocated agile team of developers and users can be an effective way of rolling out those features.

This nonproject work may not be within the PMO's immediate purview, but it is still a concern of the PMO because of its impact on resource availability for high-value-add projects. As the priority of new features declines, however, more-traditional release management methods pertain (see "Managing the Unrelenting Demand for Application Work").

2.4 Infrastructure Project Methodologies

For infrastructure and operations (I&O) work, much falls beneath the "project" threshold and would not appear on the PMO's "radar." The Information Technology Infrastructure Library (ITIL) provides I&O groups a framework for handling such work.

However, some I&O efforts certainly qualify as projects in terms of their duration, their complexity (sufficient to require significant scheduling, monitoring and control), and the need to budget and justify them separate from daily work processes. Some of these I&O projects are cross-technology in nature. They may be recursive technical infrastructure projects, or enterprise implementations requiring advanced system integration.

Infrastructure projects also can be routed along alternative paths, whether very defined or very incremental. For example, a small, departmental upgrade might have only local impact, and would require only immediate team oversight, which the line manager would track and monitor. By contrast, a major system upgrade, perhaps of servers and even the operating system, might be subject to PMO chartering, detailed impact planning in communication with affected business units, and management reports against major phase gates or even lesser milestones. Between these extremes, a moderate level of methodology might be used on projects with less-widespread impact. For example, an e-mail upgrade might be planned and tracked within the operations group, with updates for centralized reporting.

2.5 Sources of Methodology Content and Tools

For organizations with little or no defined methodology in place, or that require significant enhancement or replacement of what they have in place, various external service providers and application providers may have suitable offerings. Vendors of packaged methodologies often offer more than 10 methods from which to choose, and sell these methods as a package or individually.

These will likely require adaptation, and again, key influencers and stakeholders should be involved in their selection and adaptation. Some are general-purpose project management methodology offerings from a consortium or standards body — such as PRINCE2, developed by the British Office of Government Commerce, and the PMBOK from the Project Management Institute. Some professional service providers offer project management frameworks that are similar to, or based on, such standards, such as those from ESI International, International Institute for Learning, PM Solutions and others. The content of these is typically purchased along with training and other consulting services.

Other providers may offer similar project management frameworks but go further with more-specific IT or application delivery methodologies. Sources include some large consultancies and system integrators, such as CSC and Fujitsu Consulting, that offer AD contracting and have their own methodologies (for example, CSC Catalyst), which have rich methodology content (that they may offer commercially to clients as part of an implementation engagement), as well as various focused consultancies providing methodology consulting along with methodology content and tools.

A variety of technology providers also focus on IT/AD methodology, providing products with various levels of tool enablement, along with their own or third-party content, as well as consulting services focused on their products' methodologies. Some sources of these more-specific IT/AD methodology tools and services include:

- 6D Tech (Project Catalyst)
- BOT International (Processes on Demand)

- BrightWork (pmPoint)
- CSC (Catalyst)
- Fujitsu (Macroscope, Productivity Centre)
- Gantthead.com (Processes)
- Headstrong (dPACE)
- IBM Rational (Method Composer, Rational Unified Process)
- Microsoft (Solution Frameworks via Visual Studio Team System)
- Osellus (Process Author)
- Planview (Prisms)

Academic and public-sector sources of methodology can also provide viable alternatives (although without service or support, and largely without ongoing product development). For example, the AD organizations of state governments and public universities sometimes share methodologies with each other publicly, because the methodologies have often been developed with taxpayers' money and so are in the public domain.

For agile development, some tool providers (for example, Rally Software) offer a range of requirements management, project management, quality management and workflow features built to implement agile methodologies. In addition, boutique consultancies can be sources of agile methodologies and services. Some examples include:

- Services
  - Advanced Development Methods (Scrum)
  - ThoughtWorks (Enterprise Agile, Distributed Agile)
  - Three Rivers Institute (Test-Driven Development)
- Tools
  - Rally Software Development (Rally Community, Rally Enterprise)
  - ThoughtWorks (Mingle, Cruise, Twist)
  - VersionOne (V1: Agile Enterprise; V1: Agile Team)

3.0 Improving Project Prioritization via Post-Project Review

In financial portfolio management, investors can readily identify those investments that are yielding better or worse returns than expected. For PPM in IT, post-project reviews are equally fundamental to determining whether previous prioritization decisions were accurate, and to driving business cases and project charters toward accurate, or at least realistic, assessments of likely returns.

Many Gartner clients we speak with do little by way of reviewing project returns, and often, they voice concern or regret at not verifying that projects' benefits are in fact realized; however, there are many obstacles to doing so, explaining why benefits review so seldom occurs. Key among these obstacles are:

- Avoidance of benefits review as a defensive measure on the part of project sponsors whose business case rationale was inflated, or for which the risk assessment was unrealistically optimistic
- The considerable difficulty, time and effort that must be expended to attempt to understand, describe and quantify a project deliverable's impact — especially given the multivariable business world where, for example, a project's impact may be obscured by other factors, internal (such as merger and acquisition activity) and external (such as recession)
- Insufficient influence, authority or "clout" on the part of individual project managers attempting best-practice project postmortems, or on the part of a marginalized PMO

An ITGB — also known by such other terms as an investment decision council or IT steering committee — is critical to successfully addressing the last obstacle. An ITGB should be made up of management that is sufficiently senior that it is not competing for the IT resource, so that a PMO gathering information on project returns for its review can leverage its influence and authority.

PMOs can also address the other obstacles of "inflation" in exaggerated business cases, and of the difficulty in assessing a project's returns. Emphasizing during the project-chartering process that ITGB review of returns will occur — is already occurring — improves performance via a Hawthorne effect (as long as the behavior continues to be observed), resulting in more attention to accuracy of forecast benefits.

3.1 Likert Scale of Qualitative Benefits

As to the difficulty of assessing benefits, remember the goal: It is better prioritization for a better portfolio as a whole.

Qualitative review of benefits, as described verbally in the business case early on, is generally neglected in favor of the pursuit of quantitative measures (derived from qualitative statements), which are typically of dubious accuracy. However, qualitative review (used judiciously, along with any "low hanging" quantitative data on returns) can just as effectively drive behavioral change to improve portfolio prioritization. Use of a Likert scale (see Table 1) is a well-established method of gathering such data — see "Toolkit: Five Perspectives Beyond ROI (A Process for Scoring and Prioritizing Projects and Programs)."

Table 1. Sample Likert Scale of Qualitative Benefits

	Strongly Agree	Agree	Somewhat Agree	Somewhat Disagree	Disagree	Strongly Disagree
Did the project deliverable improve workforce efficiency?						
Did it reduce errors or rework?						
Did it reduce ongoing support costs?						
Did it enable premium pricing?						

Source: Gartner (July 2009)

In addition, qualitative Likert data will generally provide a more consistent data format for comparative purposes and trend analysis over time.

4.0 Conclusion and Recommendations

To collaborate effectively, the PMO and other stakeholders should agree on the responsibilities and on authority to better rationalize agendas and eliminate friction. They should provide five to 10 main flight paths for software and other projects to follow, recognizing that IT projects differ not only in degree, but also in kind.

An ITGB, senior to the PMO, should exercise prioritization authority, with advice and recommendations derived from a project-chartering process administered by the PMO. The PMO should also engage with an architectural review board to drive toward future-state architectures, and apply appropriate project methodologies.

RECOMMENDED READING

"Toolkit: Project Dashboard Template"

"The 'Pretty Good' PMO at Maturity Level 3"

"Q&A: PMO Services Should Be Based on Organizational Size and Maturity"

REGIONAL HEADQUARTERS

Corporate Headquarters
56 Top Gallant Road
Stamford, CT 06902-7700
U.S.A.
+1 203 964 0096

European Headquarters
Tamesis
The Glanty
Egham
Surrey, TW20 9AW
UNITED KINGDOM
+44 1784 431611

Asia/Pacific Headquarters
Gartner Australasia Pty. Ltd.
Level 9, 141 Walker Street
North Sydney
New South Wales 2060
AUSTRALIA
+61 2 9459 4600

Japan Headquarters
Gartner Japan Ltd.
Aobadai Hills, 6F
7-7, Aobadai, 4-chome
Meguro-ku, Tokyo 153-0042
JAPAN
+81 3 3481 3670

Latin America Headquarters
Gartner do Brazil
Av. das Nações Unidas, 12551
9° andar—World Trade Center
04578-903—São Paulo SP
BRAZIL
+55 11 3443 1509

Gartner