Enterprise Architecture and Projects: Maximizing EA Value for the Business

Gartner Symposium/ITxpo

November 4-8, 2007
Palais Des Festivals
Cannes, France

Philip Allega

EA and Projects — Oxymoron?

EA Is not Project Work

- EA is supposed to focus beyond any given project, between them, across them.
- EA staff should be protected from getting dragged into project specific work; otherwise, no EA will get done.

All True, but …

- EA is aligned to projects, since projects implement change — EA is planning; projects execute those plans.

Thus …

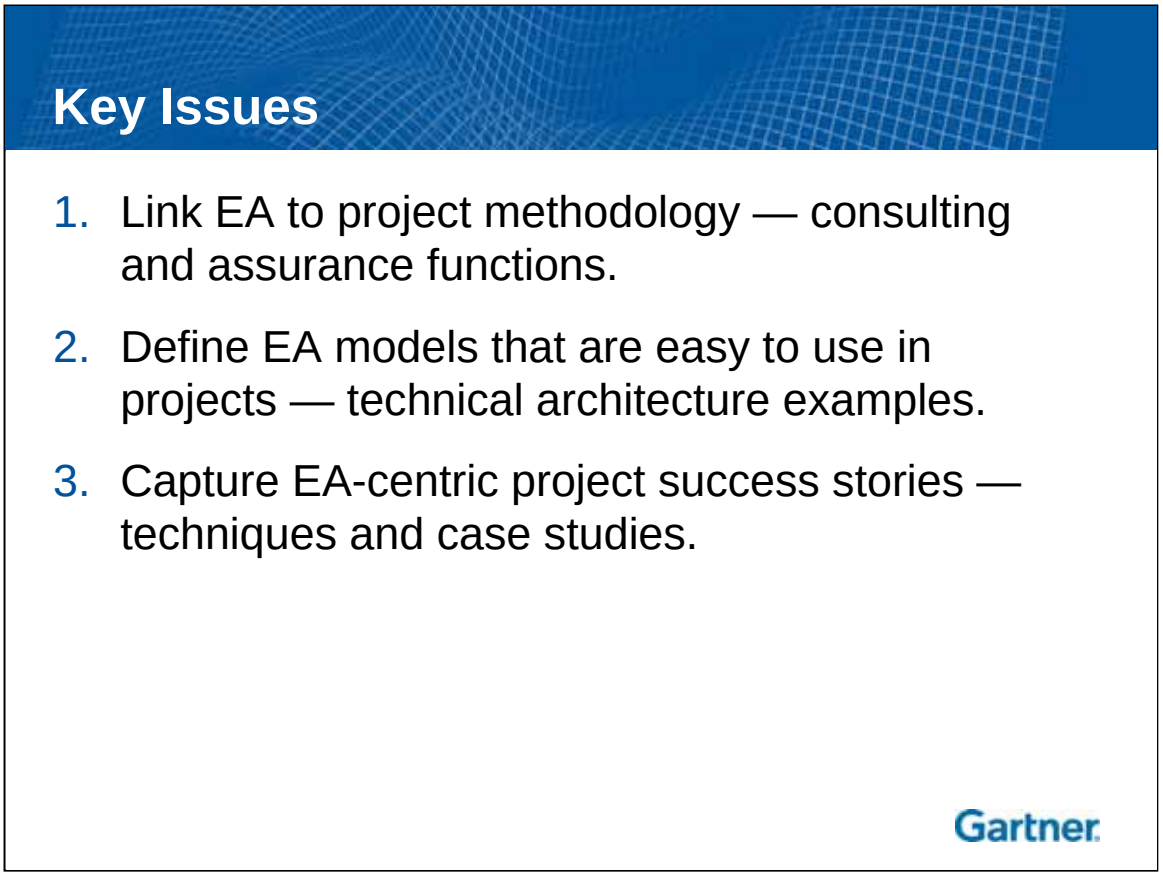
- EA must impact projects.

Gartner

Many EA teams push back from project support or all project-centric work. Why? History teaches that getting mired in project work is the easiest way to stop the EA perspective — a perspective that should be across many or all projects, not focused on just one project at a time. Time management of EA resources necessitates using EA staff for nonproject work, and chief architects often look for ways to control the level of work they allocate to projects. From the project team's perspective, adding EA is seen as slowing projects or increasing their cost.

Even though trying to stop spending all the time on projects is an important goal, the converse is also true: You can't spend no time on projects with EA. This, too, leads to difficulty. EA seems withdrawn from day-to-day activity, EA standards and guidelines get less support, projects initially spearheaded by EA wither with little EA involvement and so on.

Neither too much nor too little project-centric EA will work. Enterprises need a balanced approach. So, how is balance defined? How should project-centric work be structured in ways that don't overallocate or underallocate EA resources? How does the enterprise make EA support projects, rather than hinder them? How can EA contribute to project-centric value, the most concrete way to see any IT value from the business perspective?



In EA and EA projects, it's important to maximize EA value for the business.

Too often, EA is not used in projects or is deemed to slow them down or increase cost. However, EA can be leveraged by projects in repeatable consistent and valuable ways. This requires process change, culture change and new, project-centric architecture models.

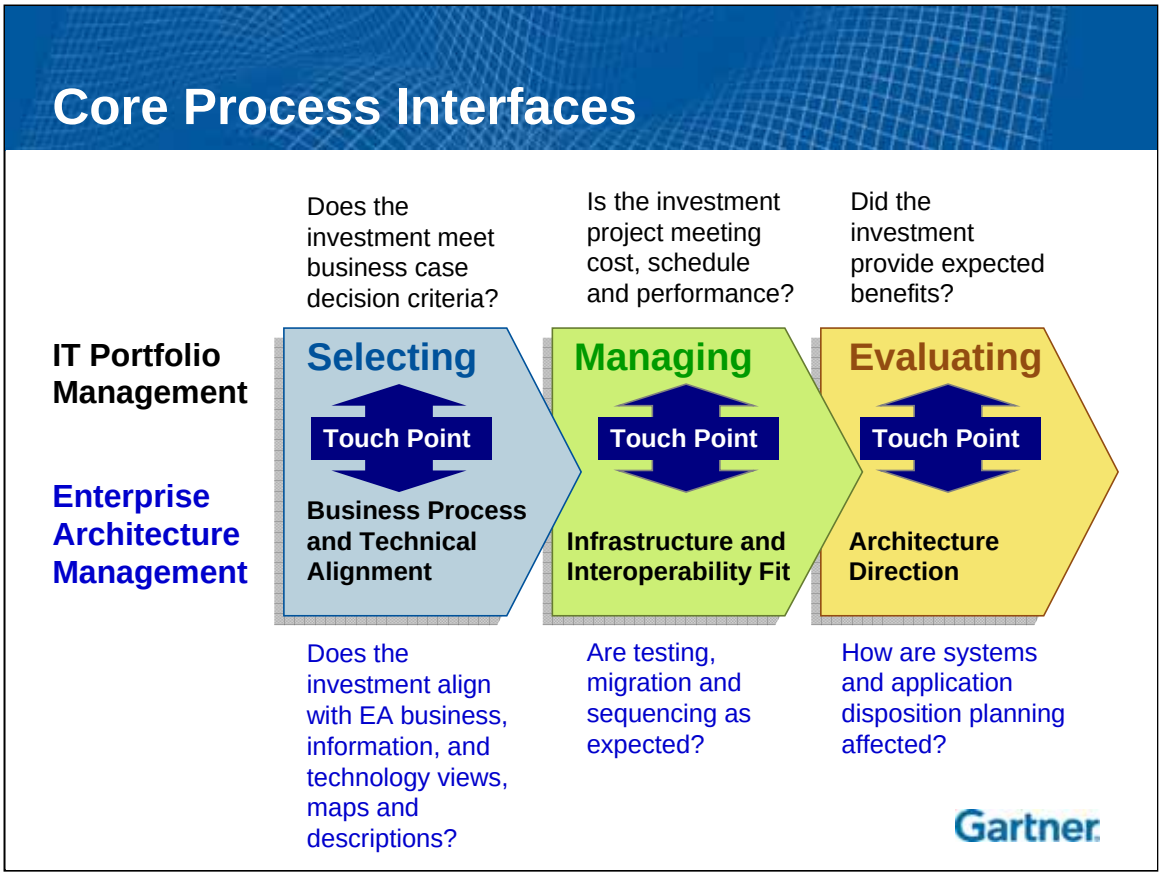
These are the key issues we'll discuss:

- Linking EA to project methodology — consulting and assurance functions
- Defining EA models that are easy to use in projects — technical architecture examples
- Capturing EA-centric project success stories — techniques and case studies

It is also important to differentiate between EA and ETA, and their respective projects. The specific examples presented will be ETA-centric (from a technology viewpoint), but they illustrate the general issues and techniques that would be true for EBA (business viewpoint) or EIA (information viewpoint) perspectives as well.

Key Issue: Link EA to project methodology — consulting and assurance functions.

Tactical Guideline: IT investment choices should be evaluated throughout their life cycles using critical questions drawn from portfolio management and EA assessment approaches.



The highest level of EA to project links should be defined at the portfolio or enterprise program management (EPM) level. EA should be supplying information to help judge all projects simultaneously, at the portfolio level, rather than only one project at a time. IT portfolio management is about selecting and managing an investment project and evaluating the expected benefit realization to the actual results. At each phase, key EA input and assessment are essential. In the selection criteria for IT investments, business, performance and technical reference models can be used to assess alignment with business and technical strategies, goals and objectives. As specific projects go through their life cycles and are evaluated regularly for cost, schedule and performance progress, testing for architecture fit, sequencing impact and project interdependencies can be included. Finally, in post-investment reviews used in comprehensive IT portfolio management approaches, actual project outcomes can be used to determine potential technical work and changes affecting the desired targeted business and technical architectures (redevelopment, re-engineering, earlier phase-out and so on). Any area of the EA framework can provide context for assessments either of single projects or whole portfolios. Most organizations start with the technology architecture links, as the examples here do, but the greater value may be in leveraging the business and information architecture as well as the EA requirements and principles that assure greater business alignment. As any individual project does a business case, this EA content can be leveraged.

Tactical Guideline: Find EA and portfolio management process synergies that can be used to strengthen analyses and decisions in both management domains.

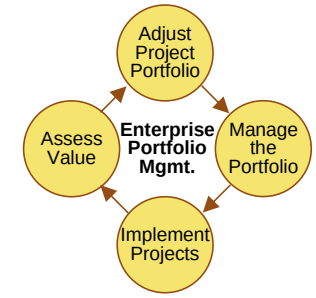
Specific Process Touchpoints

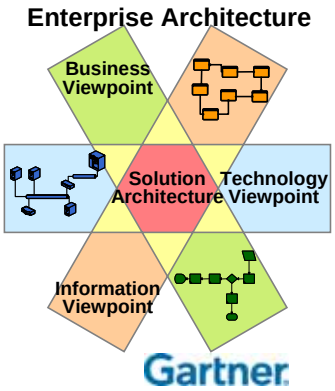
Portfolio Management Process

- EA compliance/waiver incorporated into project approval process
- EA provides foundation for as-is asset inventory expected by portfolio management
- Project selection and timing are evaluated against EA sequencing and project interdependencies affecting success
- EA weighs heavily into succession or disposition management
- EA impact factored into post-investment reviews
- EA-specific investment proposals evaluated

EA Process

- ITPM should assess progress against EA plans
- ITPM should validate inventory of active projects/programs
- Executive and IT investment committees approve EA (endorsed as a change management and transformation tool)
- Accepted projects in portfolio should comply with business, information, technical components of architecture through life cycle and should show progress toward target architecture
- EA investment work should demonstrate business impact
- Non-EA-aligned projects flagged
- Note reuse and shared application opportunities



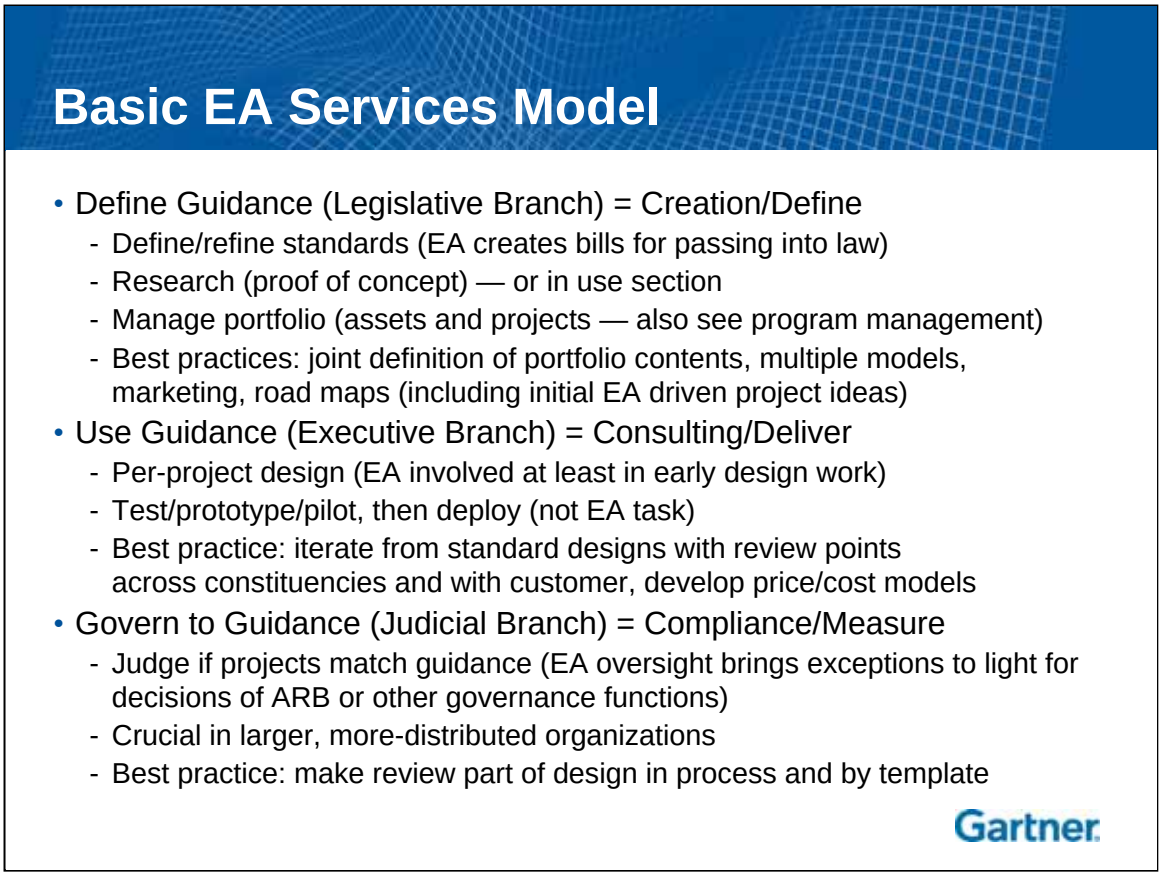


Gartner

Key Issue: Link EA to project methodology — consulting and assurance functions.

More generally, EA and IT portfolio management should not be completely separate activities, yet few organizations have taken steps to integrate these processes with any significant degree of maturity. Enterprise architects and staff seem to struggle constantly with getting nanotechnology executives and business process owners to "live the plan" envisioned in the future-state architecture. Yet, with key process steps integrated, EA and portfolio management can become strong allies and can strengthen IT governance weaknesses. But the interface must be real and must bring with it mutual benefits to architects and portfolio managers. For example, portfolio selection and review meetings should include enterprise architects for the architecture compatibility component and to ensure that the future EA state is being applied in projects. Although the EA team may have engaged business/program executives in designing the business architecture and reference models, the portfolio management process is where more-constant, regular management activity occurs on IT projects. Thus, the touchpoints between EA and portfolio management must be real and add value.

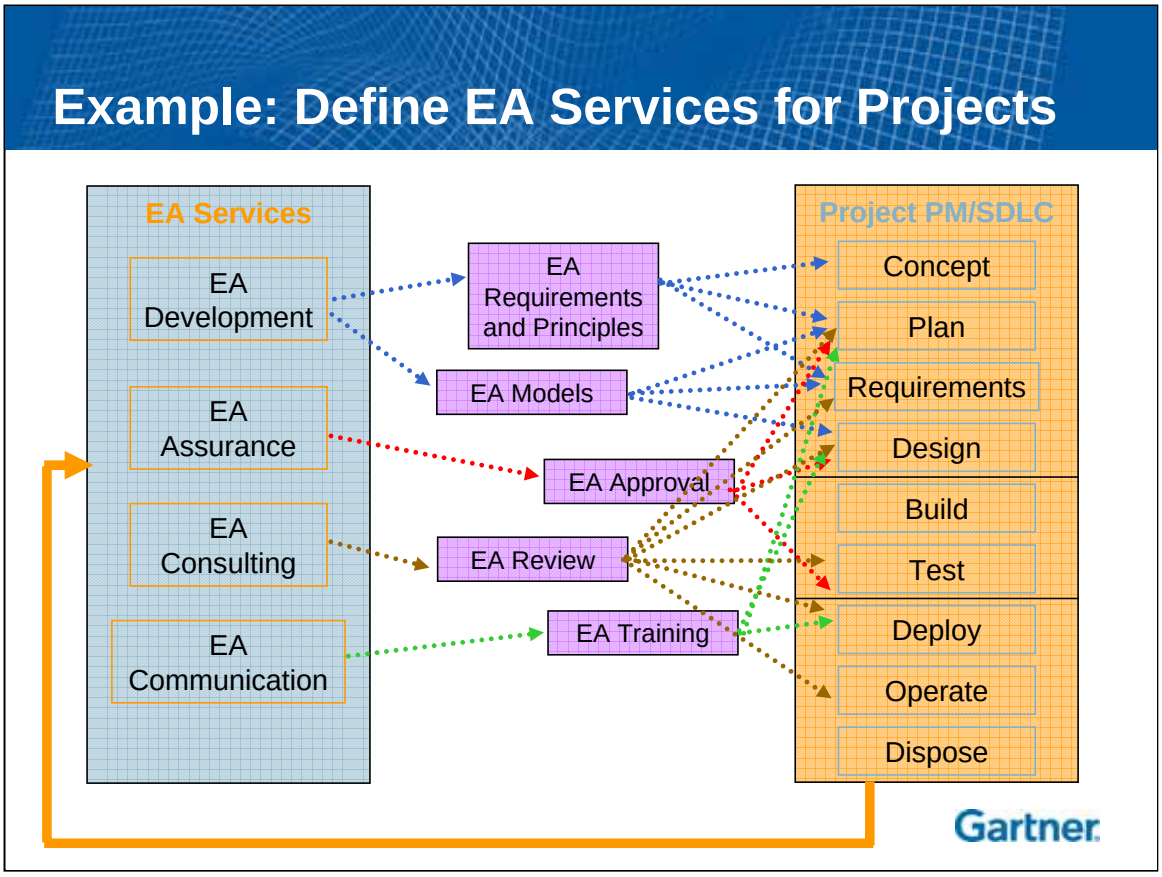
Key Issue: Define EA models that are easy to use in projects — technical architecture examples.



It is important to define an EA services model. As the slide suggests, a U.S. democratic model may illustrate the required capabilities or services. However, other models also work.

- Define one that includes creation, compliance, consulting, communication and research functions.
- Separate the job of governing entities (leadership of IT, of the business) from EA staff functions — EA doesn't govern, leaders do.
- Control the level of consultative or compliance or other project-centric work by defining separate staff if you have the resources. But, do circulate your staff between functions — don't leave certain people always only on projects. If you have a smaller enterprise, then limit the modeling work by time. For example, one client allocated Monday and Friday for EA creation activities, and left Tuesday, Wednesday and Thursday for project-centric activity.

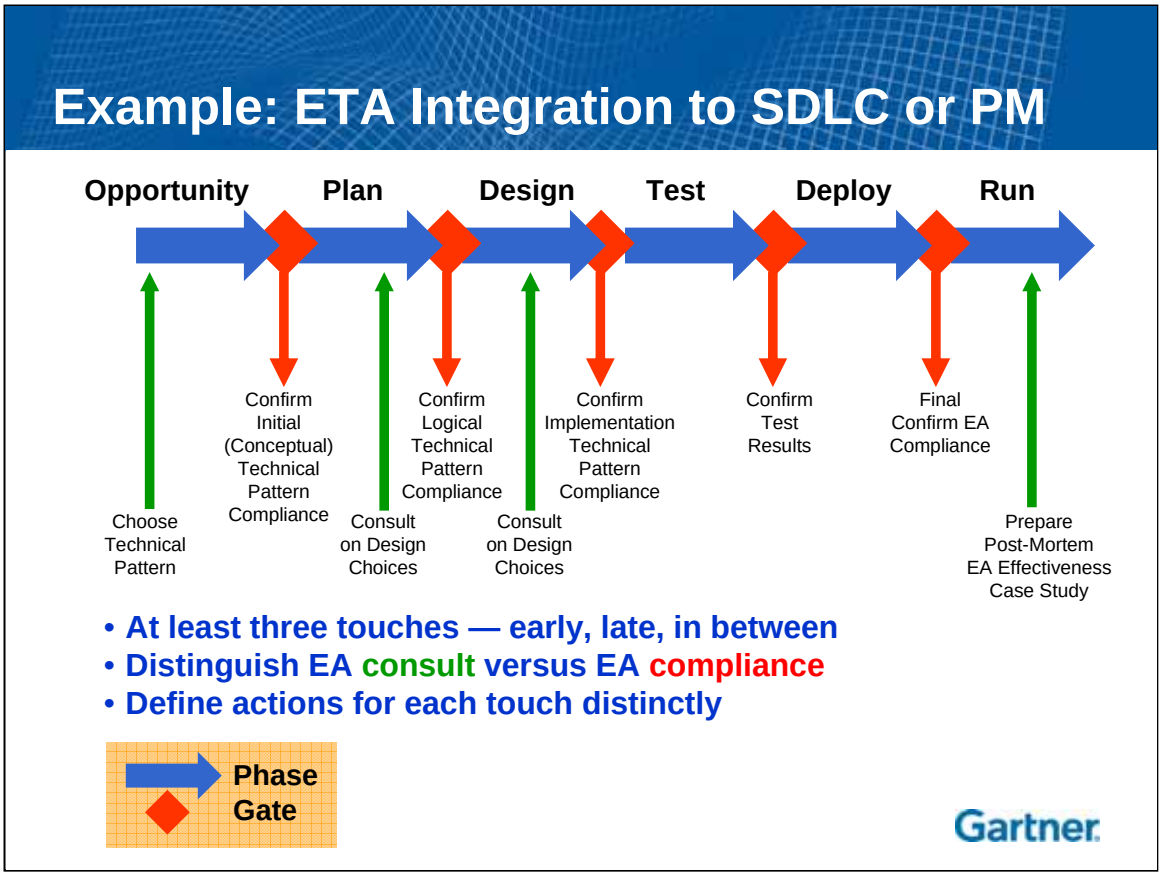
Action Item: Once you have an EA service model, communicate it well, and cement it with communicated results of each kind of activity.



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

Defining the links can be rather generic and high-level to start, but as you begin drawing the relationships, things become complex, as this example illustrates. Nonetheless, start by defining basic EA capabilities or activities or services, then get a simple view of the SDLC or PM methodology. Draw lines between these, with labels associated with more-specific activities. As it becomes more complex, start with separate views.

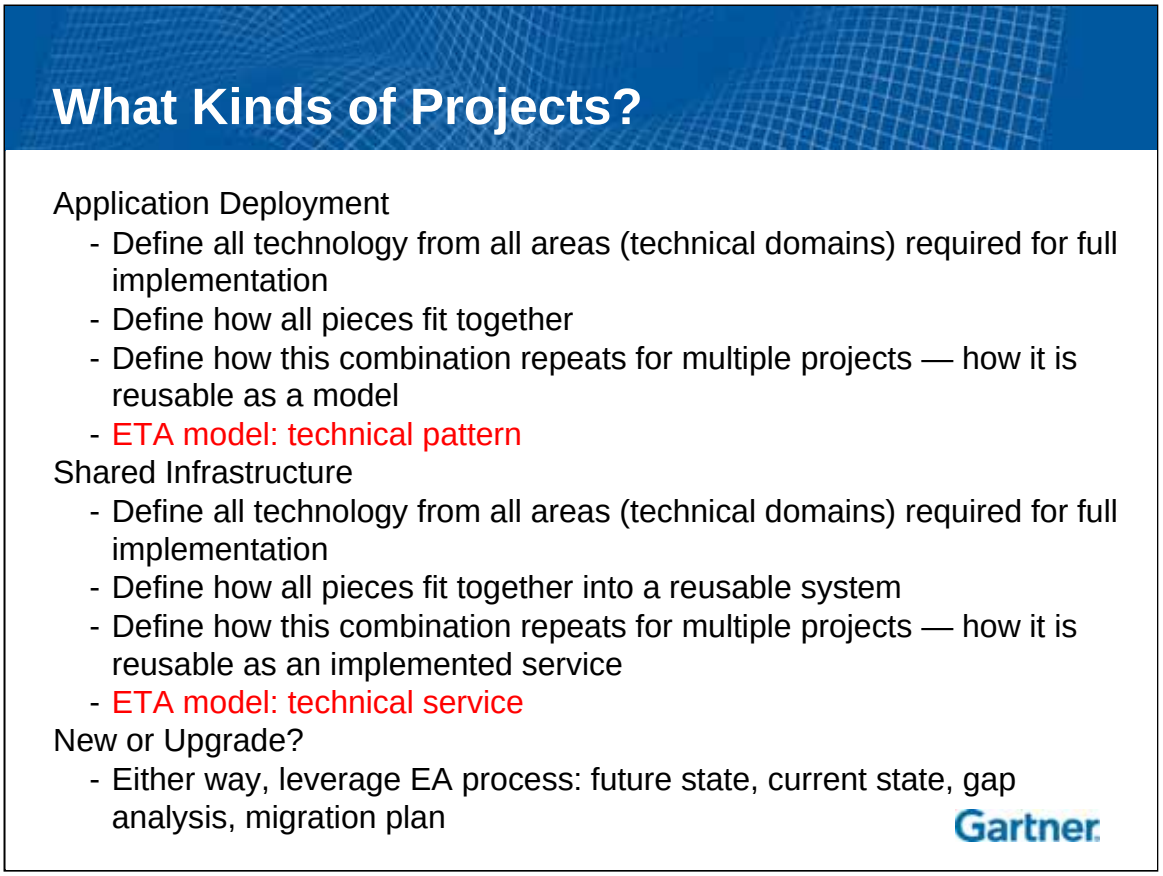
This is, of course, the business and information architecture of EA solutions, ones with very little technology — unless you consider DOC, PPT and XLS files as technology. But these links eventually will be better supported with EA tools, at least at the repository level (and providing easy project lookup of EA content, with increasingly project-centric formatted views, such as the models defined in the next section), if not also at the process management level (workflow systems and so on).



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

No matter what the solution delivery life cycle (SDLC) or project management (PM) methodology, EA can and should be mapped into specific steps to provide a variety of services. These services should be defined, and they can be applied to various SDLCs, if needed in more federated organizations with different BU PMs. There are at least two basic services to define — outgoing consulting and incoming assurance, as illustrated above.

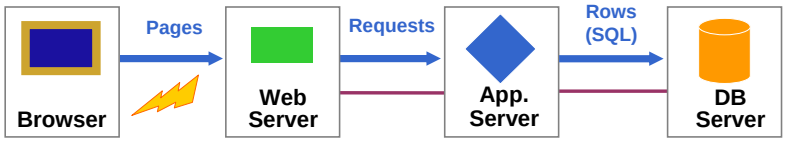
Beyond the simple idea that these basic processes should be linked, most organizations develop a much more explicit definition of exactly how particular EA activities should be delivered during stages or at stage gates. This way, the project manager (and the PMO/EPM function, more generally) can be planning for these activities to occur. EA should have some examples of how these things should be done. We consistently recommend doing "forensics" on an older project's documentation to show how new projects should use the EA guidance more directly, such as exactly how certain forms would be filled out.



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

There are different kinds of projects to which EA guidance should be applied. Some can be application or solution rollouts or updates (which implement business process and information change); others are infrastructure changes — these are increasingly about creating or updating shared infrastructure. EA models must define DIRECTLY how EA guidance should be used in each kind of project. Don't stop with defining a large list of possibilities (many products and technologies that could be useful and are approved by EA) — this makes projects do more work and allows them to combine parts in combinations contrary to EA guidelines. Moreover, this EA parts standards approach does nothing to improve the reuse of shared services, particularly infrastructure. Where is the list of shared infrastructure to reuse, as major systems, rather than just approved parts? Can you tell what's available as a shared service? And, since the project will no longer be defining the shared service (it's already built), what does the project need to know about that service? Certainly it will be less about how that service is built — the service provider view — and more about what the interface is to the service, as well as the service levels that can be expected from re-using it. To better leverage EA in projects, model EA differently than traditional domain approaches; model for leverage directly in projects of both kinds.

Example Technical Pattern: N-Tier Transaction Pattern



Name
- 3/N-Tier Transact pattern

Owner
- Bob Smith, architect

Description
- Presentation, data access, and business logic are all partitioned

Use Case
- Who — anyone
- Where — anywhere
- What — transactional applications

Service-Level Matches
- Scalability (for example, more than 500 users)
- Changing presentation logic
- Integrating new sources/consumers into application
- Speed of initial deployment

Cost/Pricing
- Derive after first few projects

Examples
- SAP R/3, PeopleSoft v.8

Principles
- Stateless farm design (except DBMS)
- Multi-POI (Web and IVR)

Component and service manifest
- API: Package, designed, standard (EJB/J2EE)
- Presentation: Apache on Solaris
- Application server: BEA WebLogic
- Integration: Leverage EAI service
- Database: Oracle v8i
- Server HW/OS: Sun Solaris on SPARC …
- Storage: EMC SAN
- Network: Leverage WAN service (need QOS for service-level assurance)
- Security: Leverage ID service
- Management: Leverage operation services

Gartner

Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

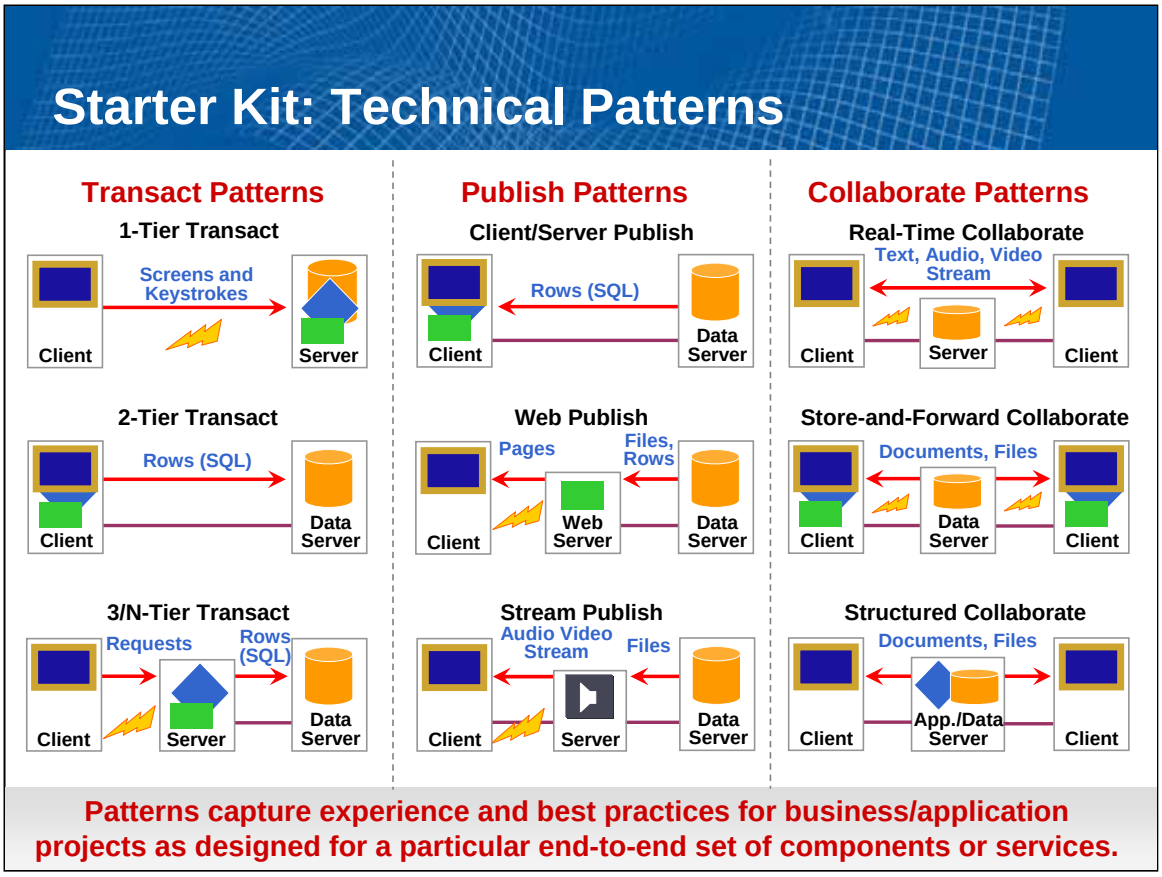
What if you could walk into every application delivery project and budget planning meeting with a simple, yet complete set of full standards that is well-documented, easy to use and specific to the project itself?

This is intended to represent only the basic content of a pattern standard. This is a conceptual-level design — a very short but reasonably evocative description of what standards apply to a common application set.

What's missing? What would you include in your standards?

Thumbnail Diagram Legend

Icons	Examples	Logical Processing
	DBMS	Resource Logic
	App. Server	Business Logic
	Presentation (Web) Server	Presentation Logic (Generation)
	Client, Desktop	Presentation Logic (Rendering)
	LAN	
	WAN	Physical Device



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

"Starter" patterns are based on the knowledge and experience of users that have implemented similar patterns, threads or blueprints to simplify application and infrastructure delivery. The categories are based on a key characteristic — how data is read, written and shared. A pattern links a set of applications to the assets that depend on or support that set of applications. Certainly, an actual inventory linkage is necessary, but also classifications by type (domain, pattern, service) will help create new understandings of the value of certain infrastructure investments by showing the repetition and reuse of certain assets.

Thumbnail Diagram Legend

Icons	Examples	Logical Processing
	DBMS	Resource Logic
	App. Server	Business Logic
	Presentation (Web) Server	Presentation Logic (Generation)
	Client, Desktop	Presentation Logic (Rendering)
	LAN	
	WAN	Physical Device

Example Technical Service: Identity Technical Service

Name
- Identity infrastructure service

Owner
- Sally Jones, architect

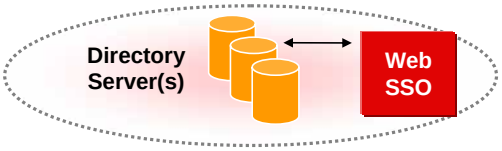
Description
- Providing user identity information (attributes), including authentication credentials and related SSO services; also offers Web URL permissions

Use Case
- Direct use by application (LDAP)
- Indirect use via Web server (with attribute passing in headers)
- Direct use by application (security APIs)

Service-Level Matches
- Scalability (for example, more than 500 users)
- Scale incrementally using replicas
- Direct application support

Pricing
- Costs included must be TCO
- "Included" in e-business costs

Maturity
- Installed since 2001 with all customer names
- Used by X, Y, Z applications now



Examples
- MSFT Active Directory (NOS file and print)
- MSFT Passport online service

Principles
- Simple authentication is usually enough
- Replication to scale (mostly read only)

Component and Service Manifest
- API: LDAP, Web server exits, proprietary
- Presentation: N/A
- Application server: N/A (see Web SSO)
- Integration: Metadirectory utilities
- Database: iPlanet Directory Server
- Server HW/OS: Sun Solaris on SPARC …
- Storage: EMC SAN
- Network: NA
- Security: Netegrity SiteMinder Web SSO
- Management: Delegated admin., …

Gartner

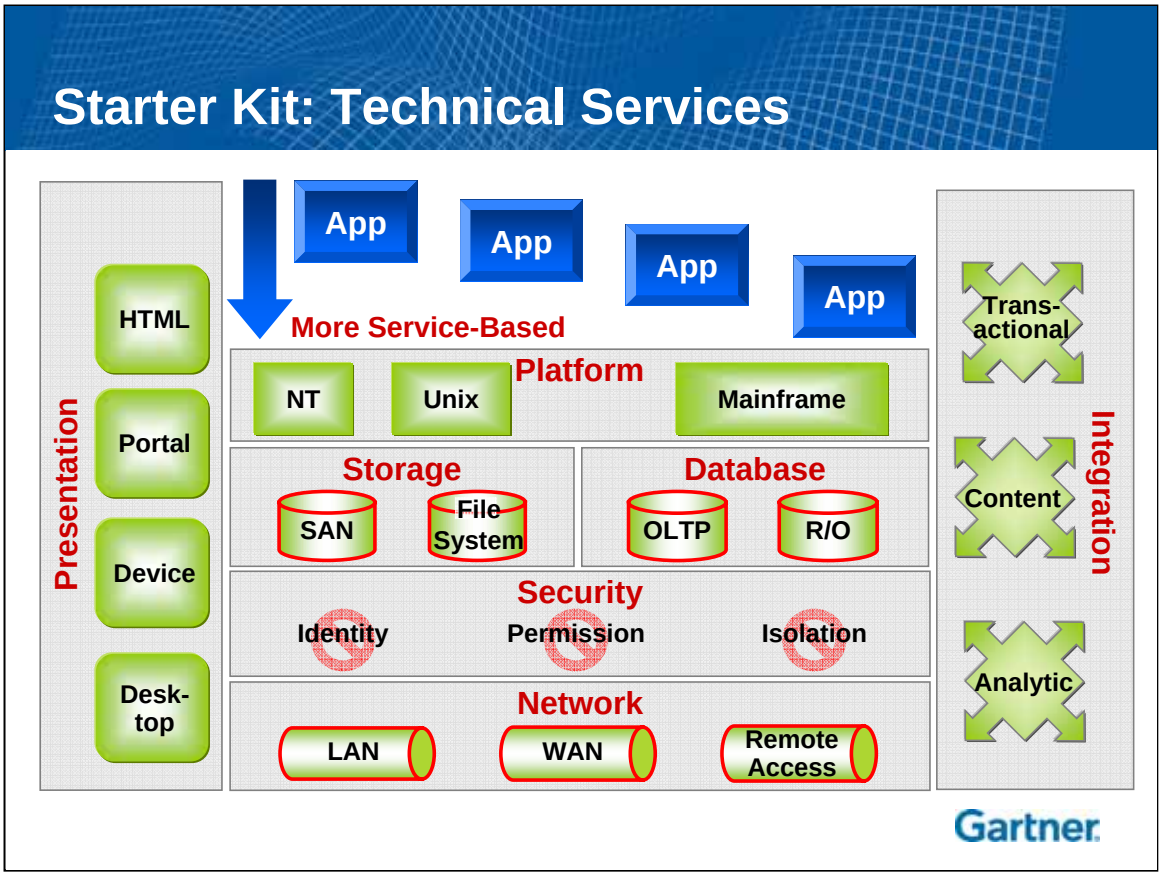
Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

What if you could walk into every project and budget planning meeting with a simple, yet complete vision of your shared infrastructure that is well-documented and easy to use?

This is intended to represent only the basic content of a technical service standard. This is a conceptual-level design — a very short but reasonably evocative description of what comprises a particular service and what it can do.

A number of actual implementations of the same conceptual technical service could be made — in this case, one leveraging Active Directory and another based on an LDAP directory.

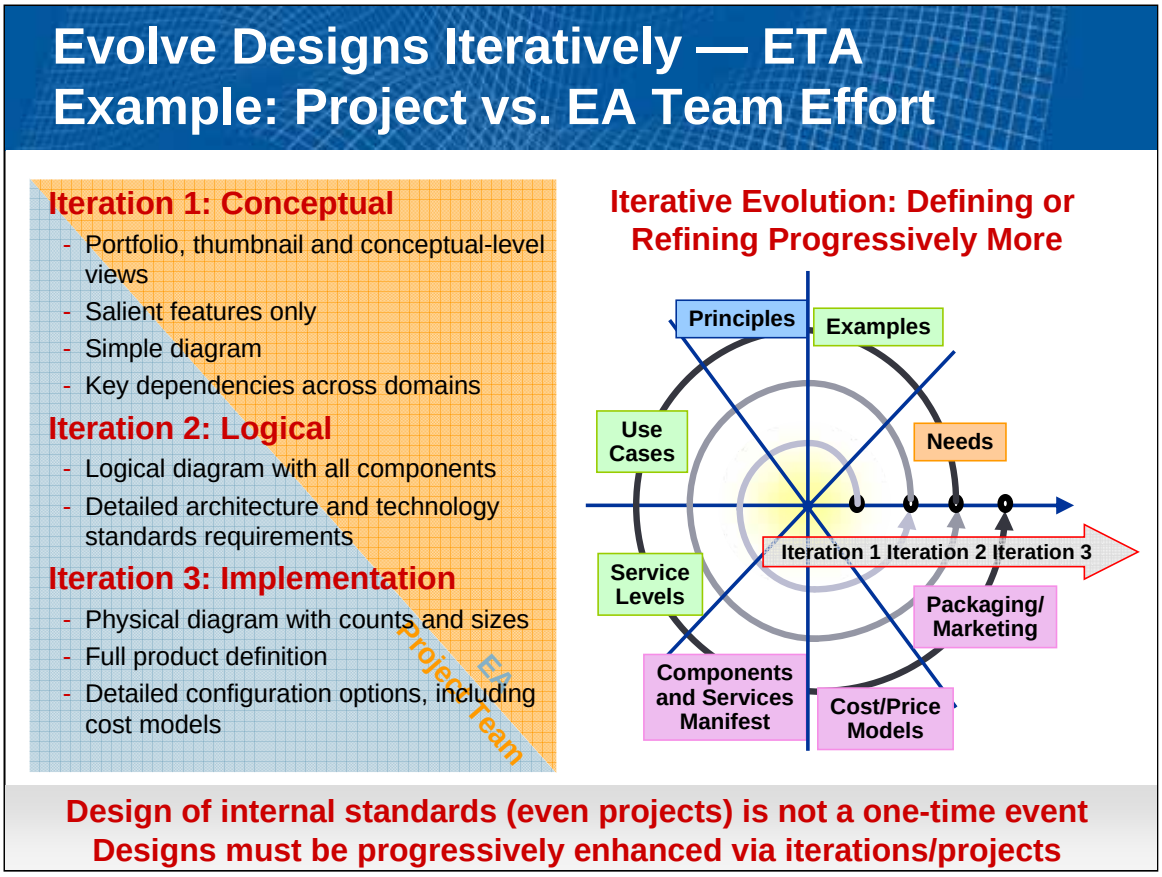
What's missing? What would you include in your standards?



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

Many models exist or can be designed. This is a starter kit, but one that has resonated with many clients.

- Communication: The Internet, a worldwide, IP-based internetwork, is the universal standard. Many organizations split this into LAN, WAN, remote access and other service options planned or paid for differently. Includes functions such as IP, DNS and DHCP. Voice is another service, not depicted.
- Presentation: HTML combines a standard Web browser, Web server and caches treated as a service. The application generates HTML, and the HTML service renders it plus additional Web edge devices and services to improve service levels, including HTTP caching, CDN services, ISP link bandwidth, various network load balancers, on-the-fly device-specific compression, encryption services. Portal adds profile, personalization and portlet services to HTML service. Device is presentation services for nonstandard browsers, mobile devices. Desktop has many locations; the standards for infrastructure technology and help desk and applications are bundled into a service IT offerings.
- Security: Identity: Authentication, user data (includes SSO, directory service). Isolation: DMZ, including firewall, NAT. Permission: Access control, authorization.
- Integration: Transactional: EAI, IEI. Analytic: ODS, DW, enterprise reporting. Content: WebDAV, ICE.
- Storage: SAN, File System: NAS, NOS (file and print).
- Database: Shared database server software on a shared platform or service (there can still be separate database tables used by different applications).



Key Issue: Define EA models that are easy to use in projects — technical architecture examples.

These iterations (the names, the details of what is in which) can be altered, but whatever the list, it should be consistent across multiple standards. Moreover, these three levels of work should match the levels of design work required in each project. Thus, if the standards already are modeled in project deliverable style, using them in projects will be easier. Project and enterprise architects will be thinking the same way.

The architectural process must add detail over time. Business-strategy-driven, trend-aware, principle-based, joint planning processes between the business and IT cultivate consistent decision making up, down and across the enterprise in forms that are iterative, spiral, evolutionary, just-enough, just-in-time, time-based and event-driven. Modeling differs by area. Scope for ETA is mostly infrastructure. The purpose is to communicate to a constituency. For level of detail, iterate to get more detail. A given model may need more than one packaging by audience (customer vs. provider). Modeling increases these same characteristics, and models should be maintained in a repository at least by description, owner and electronic link.

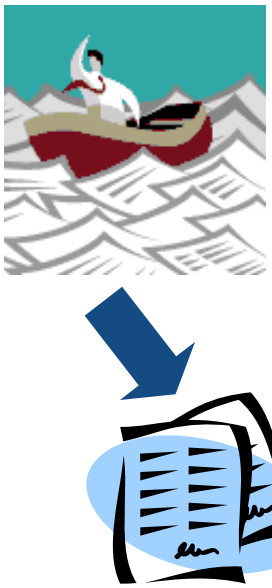
Action Item: Be sure to add detail, but don't jump into a never-ending detailed modeling exercise (analysis paralysis).

Key Issue: Capture EA-centric project success stories — techniques and case studies.

Best Practice 1: Easy EA Leverage

Make EA Content Easy to Leverage

- Use ETA technical pattern and technical service models in addition to domains and components (bricks)
- Provide automated, pattern-matching function online to ask a few questions to determine the right starter pattern (or pattern set)
- Link to other project documentation that has done similar things (for deeper content)

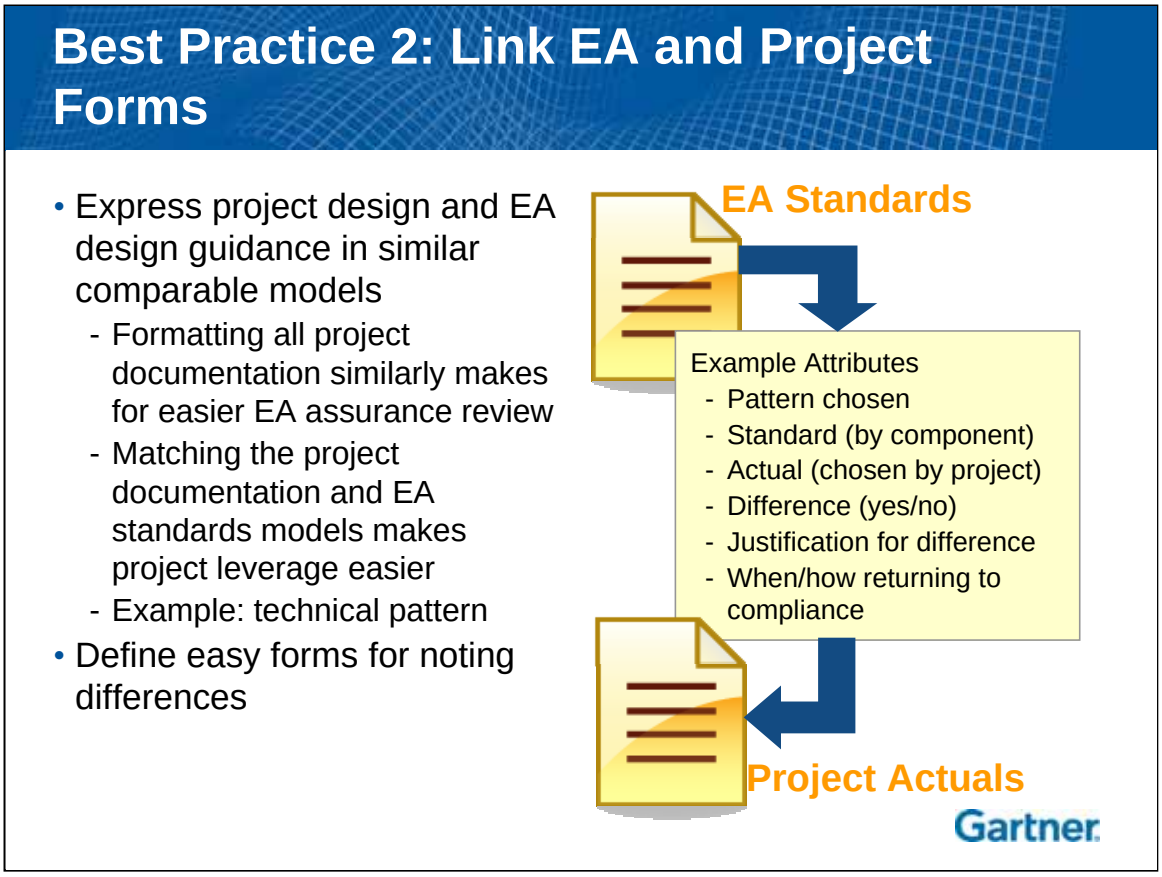


Gartner

Clients report that reducing the amount of paperwork is a critical goal — without this, projects complain that EA slows things down. So, create models that align with the way projects view the world, not differently from them — make these project-centric or applied EA models be part of the EA.

The examples in the previous section defined an ETA-specific set of models for reuse, but this point applies generally to all EA areas, no matter what framework you use. Still, use the concepts of conceptual, logical and implementation-level detail (or abstraction) to separate the level of EA input into designs, particularly for project delivery. EA should participate only at an early stage and most directly at a conceptual level but should be leveraged later for some review and/or assurance activities.

Action Items: Providing examples of how these new, project documentation approaches work, even forensically, for a first cut, can help get project teams oriented toward new structures for the project documentation. Highlight how much simpler and faster documentation and approvals can be when the new approaches are used.



Key Issue: Capture EA-centric project success stories — techniques and case studies.

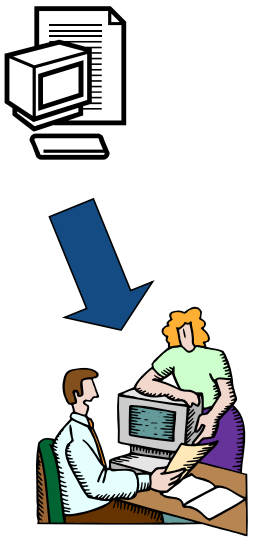
One technique that specifically works to support the load of EA assurance review is to leverage the similar structure of each project to the standards in a way that makes review simpler. The example attributes define, for example, first the standard, then the actual answer. Furthermore, there is room to note visually if these are the same — it's easier to skim through and see what specific standard choices have been changed. Of course, not all documentation can be this concise, but some levels of simplicity and easy review are certainly appropriate. Continue to refine this, and conduct meetings where project staff can give feedback and make improvement suggestions to EA for documentation issues and to the actual standards.

Another question that comes up often is about EA project questionnaires. For this, we advise separating the questions that need answers at first-conceptual-level documentation, then asking more questions later. This way, you can define and debate key issues early, without details cluttering up the discussion.

Action Items: EA and project documentation should match on first-level documentation. Separate the question list up into chunks by PM stage — they all can't be answered at the beginning anyway.

Best Practice 3: People, Not Just Paper

- Ensure that someone will support the project customers
- Dedicate some proportion of EA staff to project support — from simple, "how to leverage EA content" to detailed "design services" or "research" options
- Ensure that others are not as dedicated, and change roles regularly to keep everyone happy and up to date



Gartner

Key Issue: Capture EA-centric project success stories — techniques and case studies.

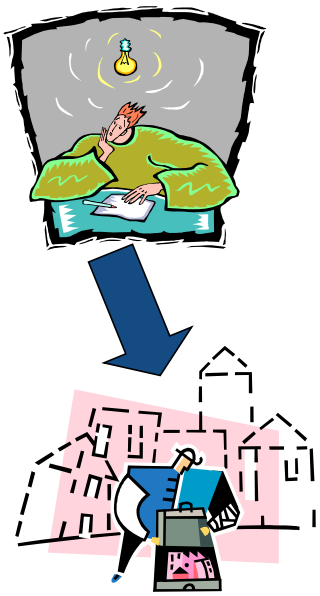
One mistake clients make is in keeping EA people too separated from projects. Smaller firms typically don't have this problem, but larger ones do. The project people are very distinct from the EA team, and the two don't seem to interact or talk much. The level of EA assurance review is provided as a process where, in some cases, the people don't talk to each other but provide only written input and feedback. Although written feedback is important, some level of human interaction generally is advised, even if it is more as a courtesy, or to explain the written content.

One way to ensure personal contact is to dedicate staff (or staff hours) to the effort. Clients with separate solution architects often take this approach. The selected individuals may be design consultants, and this is a separate EA service to support with the human touch.

Action Item: The EA assurance function could use some human touch, so allocate roles accordingly.

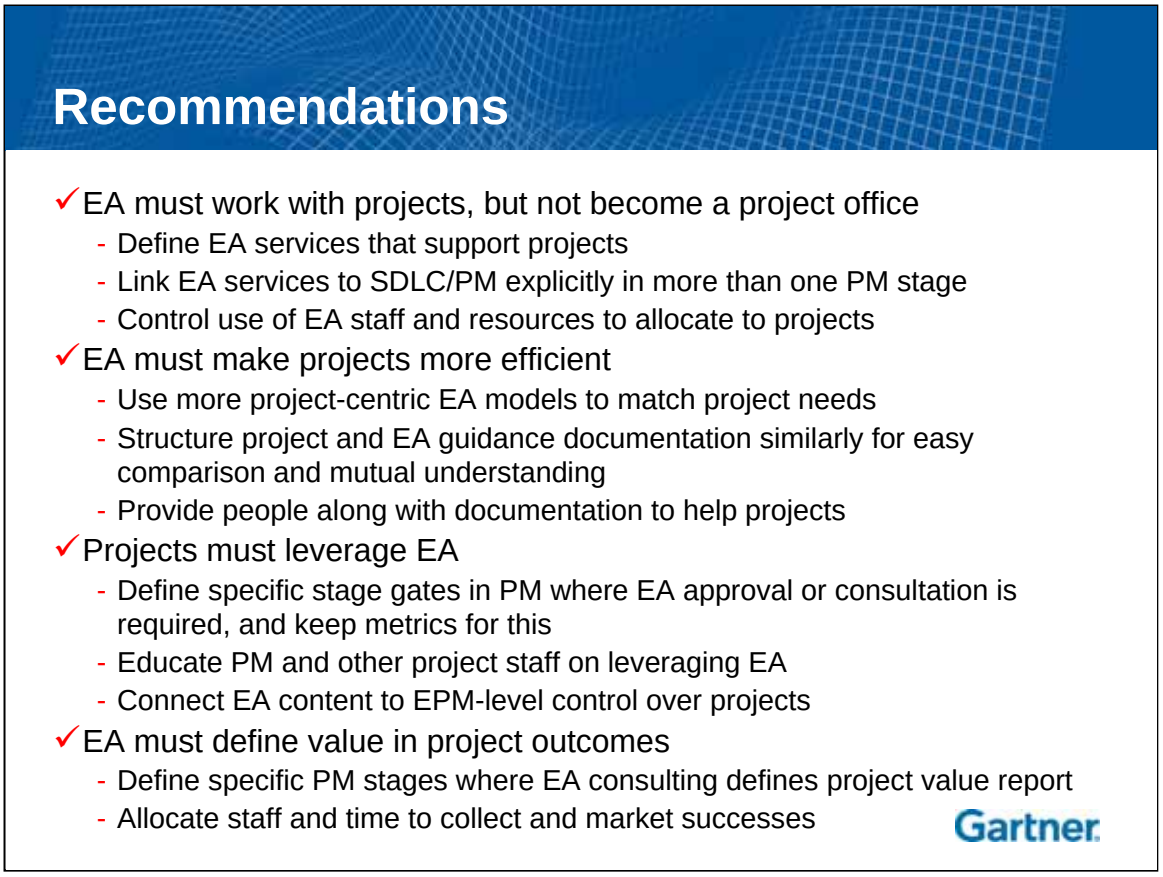
Best Practice 4: Measure Business-Level Project Outcomes

- Allocate staff and time to EA-centric project post-mortems
  - PMs will not collect everything for you
  - Prove EA was helpful to project
  - Gather project-centric metrics associated with all provided EA services
- Do early (immediately post-project) but also later (when some key EA-level benefits are measurable)



Gartner

Business Enterprise Architecture (BEA) is a very specialized case, given its size and scope. Other case studies are increasingly available on the Web. However, you must create your own case studies of situations in which your own EA has paid off in project success. You must create and collect your own metrics, then communicate your successes. No one will do it for you. Also, some responses given to you as official approvals aren't as helpful as you'd hoped. The Federal Enterprise Architecture (FEA) is a good example. With OMB oversight and EA approvals via Exhibit 300 submissions, individual U.S. federal civilian organizations cleared a budget approval hurdle, but there wasn't much that clients reported as helpful for their missions beyond this "mere" budget approval. Thus, you can't just say, "We got a green light from OMB, so our EA is good." FEA's new focus on "segment architectures" should help. Many internal stakeholders won't see or haven't seen value in this. Just because OMB/GAO finally "passed" BEA v.3 (with caveats; see www.gao.gov/new.items/d06658.pdf), EA wasn't really successful (better, yes, but not successful in many stakeholders' eyes) until projects had been completed. Similarly, most stakeholders won't see value from internal EA statistics or maturity assessments. They want results (projects), not just to hear that planning was successful.

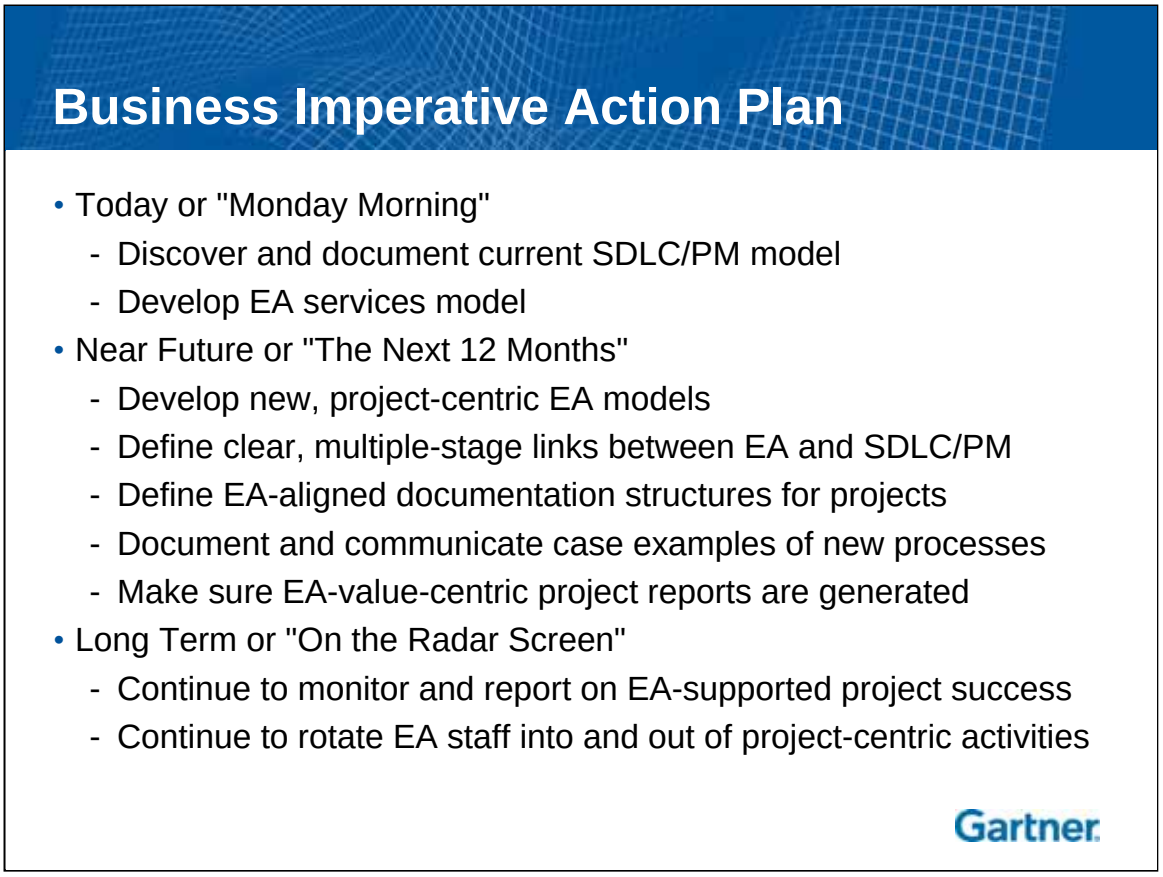


Key Issue: Capture EA-centric project success stories — techniques and case studies.

By following these best practices and approaches, EA teams can make projects leverage EA more effectively and efficiently — for their own project's success as well as for the success of the EA. Ensuring that projects don't trip over EA's involvement in them is key.

However, for the business, none of this matters unless the projects are successful on their own individual merits and for the overall value to the enterprise. EA linkages to business drivers can help ensure greater business strategic alignment.

Action Items: Leverage common requirements vision content for this at a project portfolio level, but also introduce this leverage in each individual project's business case work. Capture short-term and long-term project results to ensure that change is EA-driven and valuable to the enterprise overall.



These EA best practices enable projects to be more EA-centric and EA to be more project-centric. For the business, this should lead to more business-strategy-aligned project outcomes, with specific business case results per project, as well as overall for the project portfolio. If EA can show its contribution, then it can be valued.

EA must capture these benefits on a per-project level, with specific efforts to measure, report and communicate results. With this effort, EA's value can be shown. Without project success, EA is just planning with no successful execution — thus, not very helpful. With project success, EA is valuable — as long as the projects are good from an EA-centric and (thru the EA) a business-strategy-centric change view. This is the ultimate reward for EA and project integration for the business — successful business project outcomes.