

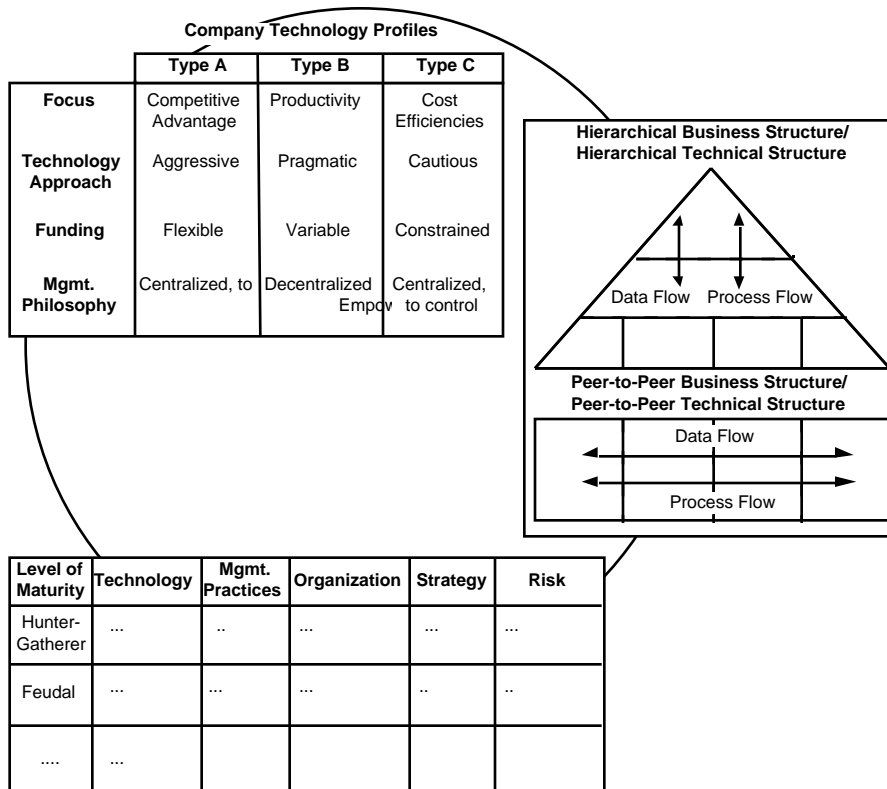
1. **How can users best align their transition strategies with the enterprise's culture and objectives?**
2. **What is a systems architecture, and how can it help users transition more smoothly?**
3. **What are the options for extending legacy applications and legacy data into new technology models?**
4. **What is the emerging model for "enterprise IS"? How will responsibilities among central IS, decentralized IS and end users be divided in the future?**
5. **How will transitions change the economics of computing? How can costs and investments be optimized at each stage of transition?**

In this era of increasing architectural chaos, it becomes tempting either to follow dominant vendors, or to set off on a unique course — in essence becoming a systems integrator, linking products acquired in mix-and-match shopping sprees. With the demise of dominant vendors, the first strategy holds little water. The second encourages the purchase of products from "boutiques" — products that may satisfy specific requirements, but do little to work within an architecture. Hence the double bind. The challenge becomes: 1) choosing and managing appropriate strategic platforms, given the balancing act among often-competing risk/reward interests, legacy applications and evolving new technologies; 2) developing and applying life cycle cost models for these technologies, to ensure an effective long-term cost benefit; and 3) developing and managing integration strategies to evolve to a balanced portfolio of platforms.

Within this framework, there are three core technology issues: infrastructure and enabling strategies and architectures; balancing legacy applications and data with new requirements; and enterprise management approaches. These topics are all best-analyzed through a dialectic process for isolating and analyzing factors that affect strategic planning.



How can users best align their transition strategies with the enterprise's culture and objectives?



Source: Gartner Group

This chart shows three models for analyzing situational business strategies, and for evolving the beginnings of technology strategies that match, and add value. Gartner Group's Type A-B-C client model is a good place to start to understand the "technology appetite" of different parts of the organization. Note that we do not believe this type of analysis works in analyzing whole companies. Rather, different parts of the same company tend to fall into different categories, often depending on customer relationship issues and the strategic impact of technology.

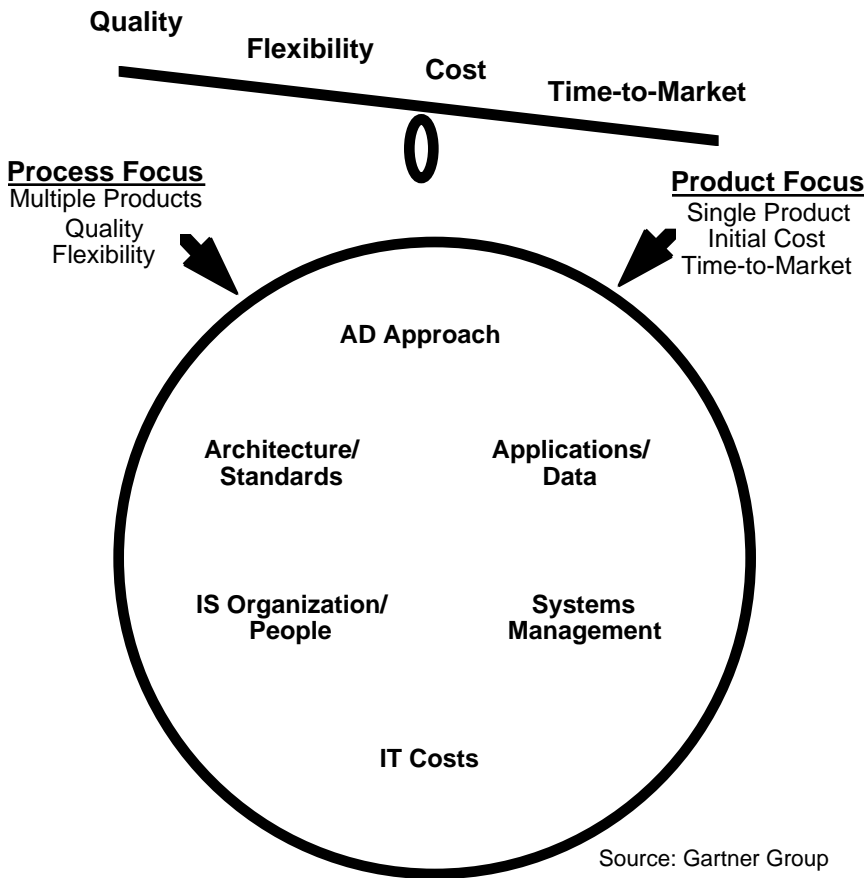
The second model examines hierarchical vs. peer-centered data flow and decision making. Predominantly hierarchical organizational structures are best-served with hierarchical data and processing structures.

The third model examines organizational maturity in light of technology, management practices, strategies and risks. The idea here is that different levels of organizational maturity require different technology and organizational approaches.



Rarely does one strategy fit all. The challenge is to blend and optimize the need for enterprise frameworks with time-to-market objectives (0.8 probability).

Reader Notes



Key Issue: How can users best align their transition strategies with the enterprise's culture and objectives?

Today's business drivers demand achieving time-to-market objectives — rapid development, lower initial applications cost, and end-user control — resulting in the distribution of IT resources, budgets and decision making. However, users must balance these requirements against the enterprise's need for quality, durability and lower ongoing operating costs. Finding the balance in terms of AD approach, IS organization, deployment of IT resources, budgets and enterprise decision making permeates the search for an appropriate enterprise transition strategy. The most successful strategies achieve the balance through frameworks and architectural consistency for quality and economies of scale, while enabling rapid response to market requirements to flourish.



What is a systems architecture, and how can it help users transition more smoothly?

Mapping Business Architecture to IT Architecture

Business Architecture	IT Architecture
Business unit autonomy	Tension between top-down and bottom-up buying (make vs. buy)
Focus on customer service; 24x365 operation	Clustering for high availability
Demand for data mining to expose sales opportunity	Parallel database operations
Reorganization of work from task-centric to process-centric	Integration of transactional semantics with workflow
Pressure on cost of sales and short sales cycles	Support for mobile computing; seamless intermittent connection
Integration of supply chain	Cross-vendor client/server; trustworthy messaging
Pressure on IT cost	Use of latest technology

Source: Gartner Group

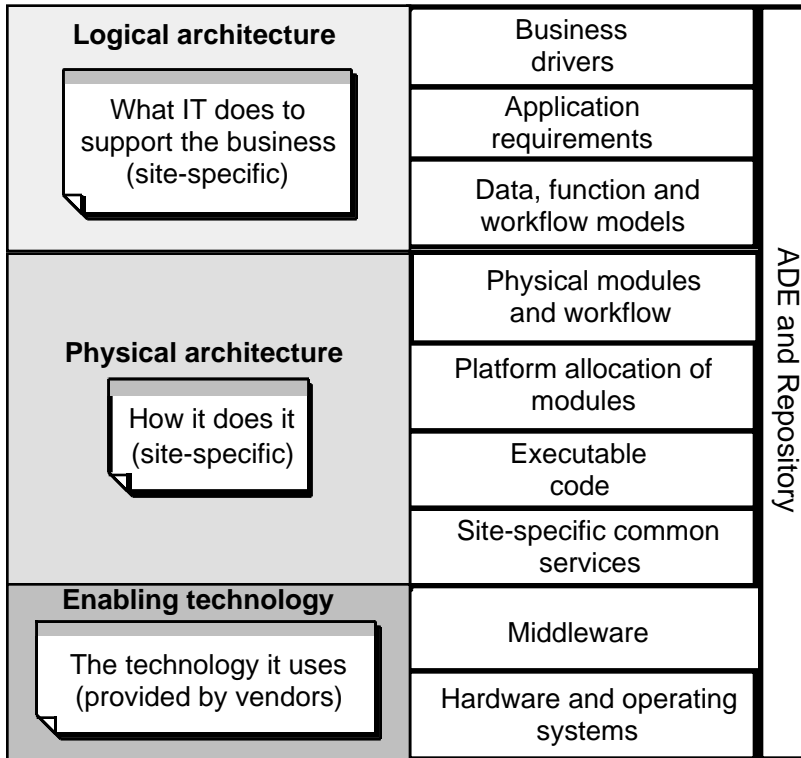
Explaining why a particular subset of the enterprise architecture is being addressed first is easier when the rationale is placed in a business context — business drivers, business-related IT goals and business-supporting architectural principles. Linking the architecture to business issues supports two key objectives: It helps in selling the architecture to users and it helps to justify the financial investments required. The chart above shows one simple way to explain how IT architectures are derived from business architectures.

The two most common reasons that enterprise architectures (or technical architectures) fail to win broad support are that they are unnecessarily restrictive, and they fail to show how they relate to and support business goals. We suggest that without a clear understanding of business drivers, the architect will not understand what the enterprise architecture must include and will be unable to sell it even if, by chance, it is technically correct.



With a well-designed architecture, decisions regarding hardware and operating-system platform selection can be made late in the application development and acquisition cycle.

An Example of An Application Architecture



Source: Gartner Group

Key Issue: What is a systems architecture, and how can it help users transition more smoothly?

An architecture is a layered set of application and computing technologies, with each layer delivering specific functions, inputs and outputs and providing a high degree of insulation from technology choices below it. We show a *template* for an architecture, with three layers: 1) the *logical architecture* — what IT needs to do to support the enterprise; 2) the *physical architecture* — how IT implements the logical architecture; and 3) the *enabling technology* — the middleware, operating system and hardware technology that enable the physical architecture to execute. A key goal is to separate the logical architecture from the physical architecture and from the underlying enabling technology, thus providing technology choices at lower layers, independent from choices at higher ones.

A context for the architecture is also needed. Governance, policies and procedures, reporting systems, and measurements of success are a necessary framework for the overall architecture.



An architecture's effectiveness will be determined by how well it is communicated and marketed to users. Strategies that reward compliance over time will achieve higher levels of adoption by users (0.8 probability).

Architectural Vision

- Goals: What are the objectives of this architecture?
- Principles: What are the underlying beliefs and values?
- Procedures (e.g., deciding when to buy an applications package)
- Standards (e.g., X.400)

Standards

- Objective: What is the standard? Why are we adopting it?
- Key Components: What does the standard cover?
- Vendor(s): Which vendors supply products that meet the standard?
- Rationale: The principal considerations in choosing a product and guidance — what should be done today, what is to be avoided, what is the mainstream direction, and what will emerge in the future (three to five years)?

	Avoid Now	Standard 1994-95	Direction 1995-96	Emerging
Products	Ultrix SunOS AIX/ESA	HP/UX AIX/6000 Digital OSF1 Solaris	X/Open-certified Unix OSs from HP, IBM, Sun Digital and others	Object standards Microkernels 64-Bit computing
Actions		Plan to move to OSF/DCE		

Source: Gartner Group

Key Issue: What is a systems architecture, and how can it help users transition more smoothly?

The reality is that the pace of change has far outstripped the ability of many IS organizations to execute an orderly planning process. Architectures must be developed real-time, often based on the needs of a few key applications. Defining the goals, principles and standards that these applications require, then documenting and marketing the results as guidance to the rest of the organization, can begin the architectural process. The result is a context and criteria for choosing products. The principles are the same; the process is accelerated.

Marketing the architecture as the enterprise's strategic direction, rationale and timetable, along with its benefits, provides guidance to users. Users aligned with this direction will benefit from reduced development and support requirements, lower initial and ongoing costs, higher-quality support, and faster implementation of their applications. These benefits must be widely publicized and marketed. Accommodate users that support too many products initially and migrate them later.



Architectures will raise political issues that must be dealt with upfront. Knowing which issues can be addressed and which can be avoided will strongly influence the architecture's acceptance.

Knowing When to Fight

Avoid These Issues	Attack These Issues
IT-initiated culture changes	Support for CEO's culture changes
Hardware ownership	Acceptance of ESP construct
One platform	Acceptance of "software first" ethic
Detailed legacy application analysis	Sore thumb and sitting duck analysis
The perfect methodology	Having a methodology (with RIP)
One data architecture (first year)	Gateway compliance
RDBMS brand wars	Standardized API, multi-target development tools, macros vs. stored procedures
Imposing enterprise standards on purchased packages	Standardized tier interfaces
One standard set of applications	Consider buffer API
Standardized client development tools	General behavior guides
One RPC, ORB, TP monitor	
Detailed GUI standards	

Source: Gartner Group

Key Issue: What is a systems architecture, and how can it help users transition more smoothly?

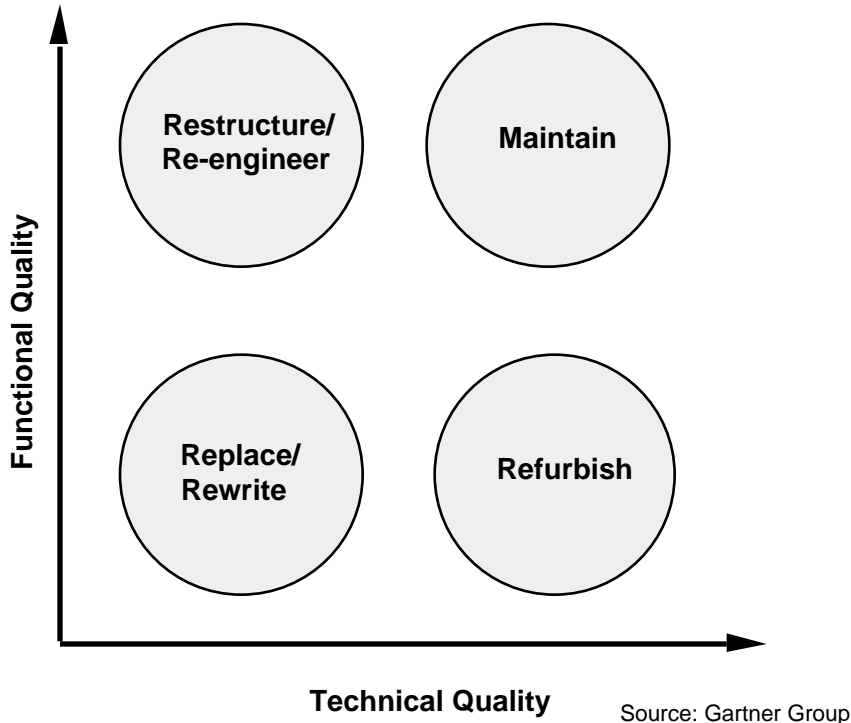
The architectural approach, by its nature, permits many of the most-virulent controversies in architecture design to be avoided. As an example, the high rate of change in the current (and foreseeable) business environment has led to a preference for purchasing applications and installing them quickly. A "standardization" approach could attempt to force data definition standards, database engine standards and language standards on all package purchases, effectively stopping them — after which a user revolt would simply override the standards. An architectural approach would suggest encapsulating the purchased applications, making them invocable through the enterprise's common user interface, and requiring them to pass data to enterprise reporting systems and data warehouses using common data definitions and common messaging infrastructures. That level of integration (and effort) would have a higher probability of success. While suboptimal in the sense of permitting technologies to proliferate (adding to support cost), "suboptimal that works" is preferable to "optimal that is ignored."



What are the options for extending legacy applications and data into new technology models?

Reader Notes

Not all legacy systems should be migrated to client/server technologies. An analysis of the technical and functional quality of legacy systems, in concert with business goals, should be used to determine the technology platform.



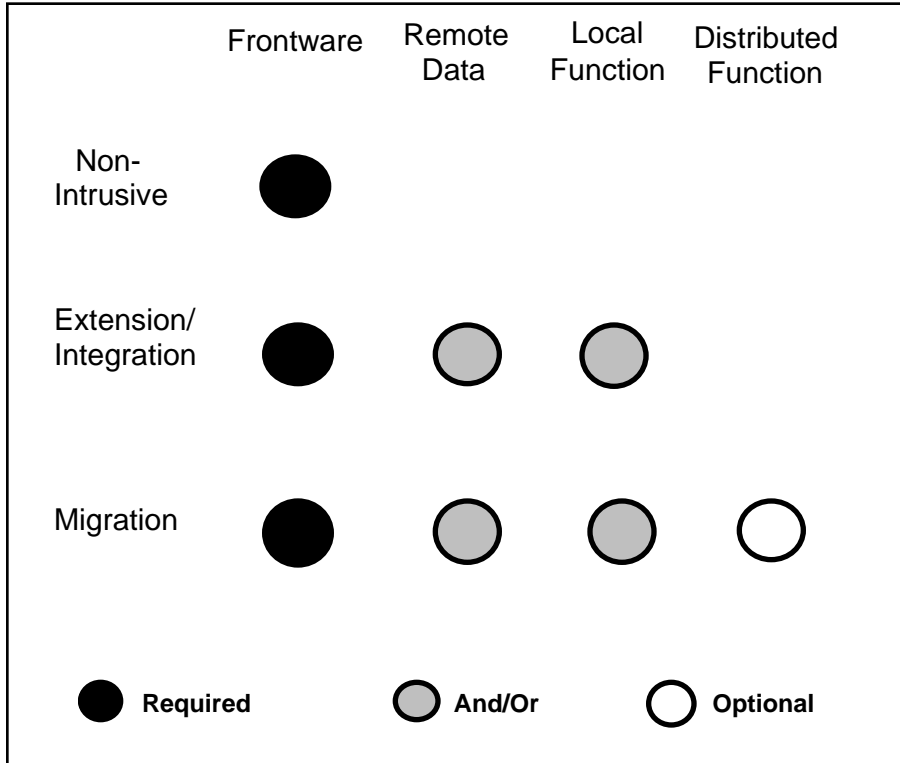
One of the most-common questions we hear from clients is, “How do we get to client/server?” The question we ask in return is, “Why do you want to — what do you perceive the benefits to be?” Beyond benefits, we also recommend that clients construct a client/server white paper, one of whose sections should be an “applications profile” that outlines the classes of applications that are suitable for reworking into a client/server production style. Cost justification, a user role, is a key piece of this analysis, because our belief is that the payback for client/server systems is in the end-user environment. The IT environment fundamentally only “enjoys” greater cost.

A single strategy that is either “all applications will be rewritten in client/server” or “all applications will be stabilized until they die a natural death” is too simplistic and naive, and violates our notions of situational decision making. Clearly, some applications, as seen in the chart above, are good candidates for rewriting.



Through 1998, IS will migrate 60 percent of existing applications to a form of client/server through the use of frontware, partial rewrites, salvaging or complete replacement (0.8 probability).

Reader Notes



Source: Gartner Group

Key Issue: What are the options for extending legacy applications and data into new technology models?

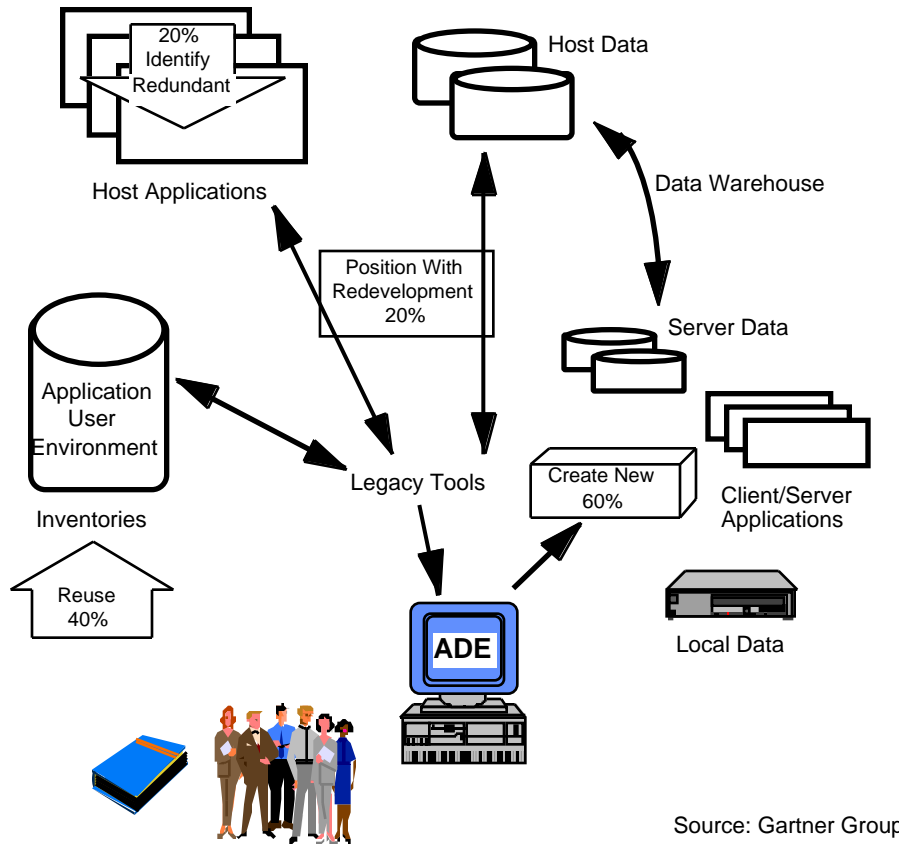
As part of a client/server strategy for delivering integrated legacy applications on the desktop, legacy extension vendors address: the user interface (frontware), the desktop data access associated with the application (remote data management), the desktop capabilities associated with the application (local functionality) and the distribution of functionality. By incorporating legacy reuse in their strategies and focusing new development on filling the “gaps,” i.e., the missing components, we believe that users can reduce the costs of transition and accelerate the results desired.

The key vendors and their products are: ClientSoft, ClientBuilder; Computer Associates, VisualRealia; Easel, Easel; IBM, VisualLift; KnowledgeWare, Flashpoint; Micro Focus, Dialog System; and Mozart Systems, Mozart.



Through 1999, integrating new development and legacy systems will allow 40 percent of legacy applications to be reused when implementing client/server applications (0.8 probability).

Reader Notes



Key Issue: What are the options for extending legacy applications and data into new technology models?

Integrating legacy systems with new development is a key component of an overall development plan. Reconciling the current environment (legacy) with the new business requirements by identifying existing capabilities and by exposing “gaps” that must be filled with new applications provides a top-down and bottom-up perspective of the application architecture needed to support the business. The goal is to reconcile inventories and match physical components of high technical quality with legacy applications that are meeting the needs of driver business units with high functional quality.

The process of incorporating legacy systems into the transition toward client/server architectures is based on an organization understanding what it has, understanding what it does not have but needs, considering its options, and then acting. The process is iterative and continuous. Organizations must think strategically and act tactically.



What is the emerging model for “enterprise IS”? How will responsibilities among central IS, decentralized IS and end users be divided in the future?

Reader Notes

Yesterday’s organizations were designed to optimize technology; the emerging IS organization will optimize delivery of products and services to end users (0.8 probability).

	Old IS	New IS
Mission	Command/control	Partner
Organization	Technology	Labor
IT Resources		
Hardware	Mainframe-centric	Network-centric
Software	MVS-centric	Middleware-centric
People/Skills	Subsystem	Interoperability
“Metric” of Success	Efficiency	Effectiveness
Business Practices	Optimize hardware	Optimize service

Source: Gartner Group

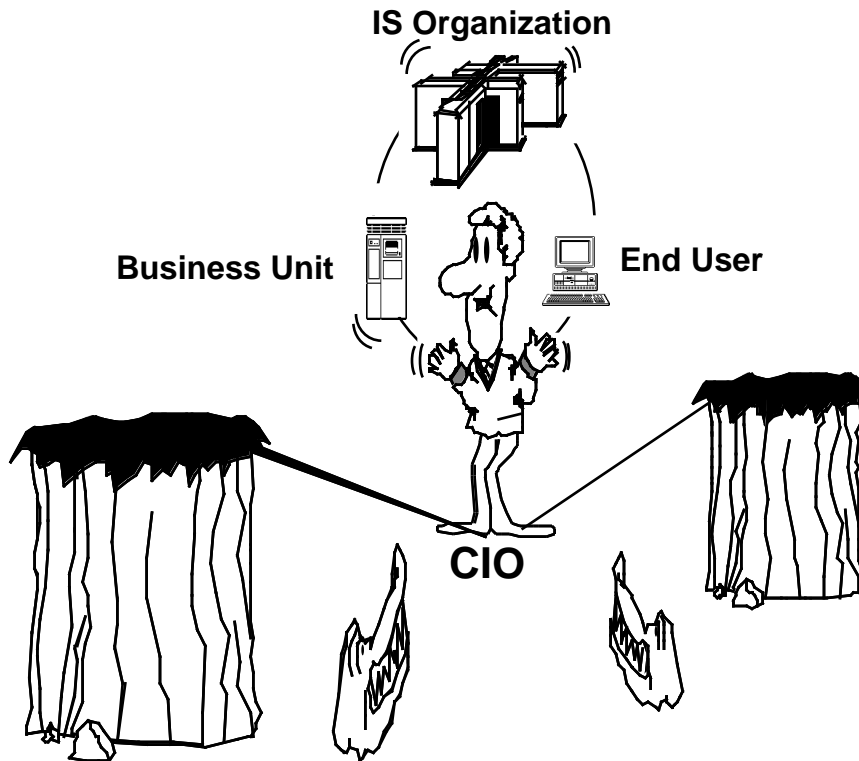
Organizational structures focus resources on a desired result. Migrations to distributed, heterogeneous computing — including blending new technologies with legacy applications, data and systems; rebalancing roles and responsibilities between central IS and business units and adjusting to a new cost structure as high technology costs are displaced by increasing labor costs — usually change the results desired.

While there is no one correct IS organization, there are common characteristics that we observe among users successfully accomplishing the transition. These users are evolving toward a service-based model that emphasizes delivery of products and services to customers. This new model changes central IS' mission, organization, people, skills and business practices. It also redistributes IS roles and responsibilities across the enterprise. The change is a multidimensional one — changing one dimension has little effect; all must change concurrently. We believe this new vision will prove to be the most successful model for IS organizations in the future.



By 1998, business management will assume primary responsibility for decisions about where and how much to invest in IT; IS' role will shift to providing the supporting architecture and infrastructure (0.8 probability).

Reader Notes



Source: Gartner Group

Key Issue: What is the emerging model for “enterprise IS”?

Enterprise governance defines roles and responsibilities between central IS and business unit IS. Increasingly, decisions about where and how much to invest in IT are becoming the primary responsibility of business management, as well as owning, operating and managing the resources that support its operations. IS' role, increasingly, is moving toward providing the supporting IT architecture and infrastructure that enables decentralized, flattened business organizations to flourish and remain responsive to market conditions.

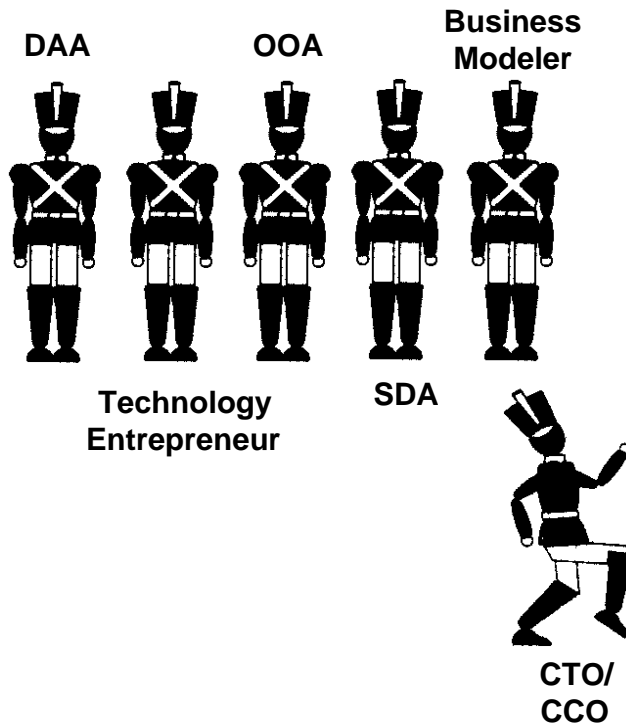
Designing the enterprise IS organization of the future means finding the balance between three organizing principles: 1) business management ownership of applications and IT resources, 2) IS ownership of the architecture, and 3) central management of shared resources (but not necessarily on a mainframe or by IS). Each enterprise will tailor these principles to their unique cultural and political needs; however, they establish a framework for balancing roles and responsibilities across the enterprise. Paradoxically, the more dispersed applications and IT resources become, the greater the need for central planning, administration and leadership.



IT organizational flexibility will be a key determinant in next-generational success. New job titles and management structures will need to be implemented within the next two years.

Reader Notes

Organizational resistance to cultural change will continue to be a barrier to architectural innovation and organizational renewal.



Source: Gartner Group

Key Issue: What is the emerging model for “enterprise IS”?

A key aspect of success will be determined by organizational flexibility. Not only will new management structures be necessary, emphasizing teams and the learning organization, but new job titles will also be necessary, reflecting technology advancement and match-up to business processes.

For example, two new architecture positions are necessary: the distributed applications architect (DAA) and the screen design architect (SDA). These positions ease the transition to client/server environments, and are necessary to specialize and move design and certain code assignments to different levels of the organization. An object-oriented administrator (OOA) stands guard over the issues of object reuse. The technology entrepreneur champions new technology, and creates appropriate pilot and investment strategies. Unless the chief technology officer (CTO) is surrounded by these people, and by change-oriented managers, stagnation is likely. In some Type A companies, the CTO also functions as a chief competitive officer (CCO).



A thorough self-analysis of IS organizational maturity leads to appropriate choices in technology, architectural principles, organizational structures and skills balance. This “roadmap” can help users determine appropriate “next steps” in their evolution.

Level of Maturity	Technology Environment	Architecture	Organization	Skills
Hunter-Gatherer	Mainframe-centric	Vendor-supplied	Organized around technology IS-directed applications and resources	IS “gurus” MVS-centric skills
Feudal	Islands of automation	Multiple IS strategy defined	Help desk in place IS and BU independent responsibilities	MVS plus distributed product skills
Renaissance	Cross-platform connected systems	Middleware standards Application-specific interoperability	Manage to customer satisfaction Shared IS/BU responsibilities	Cross-systems skills Build delivery processes One to three key projects supported Acquire outside resources
Industrial Revolution	One or two key applications interoperable	Architecture defined and marketed	BU-directed processes	Build applications development processes
Star Wars	Enterprisewide interoperability	Enterprisewide architecture adopted	BU-directed investments IS-directed backbone Costs optimized	Build management processes

Source: Gartner Group

Key Issue: What is the emerging model for “enterprise IS”?

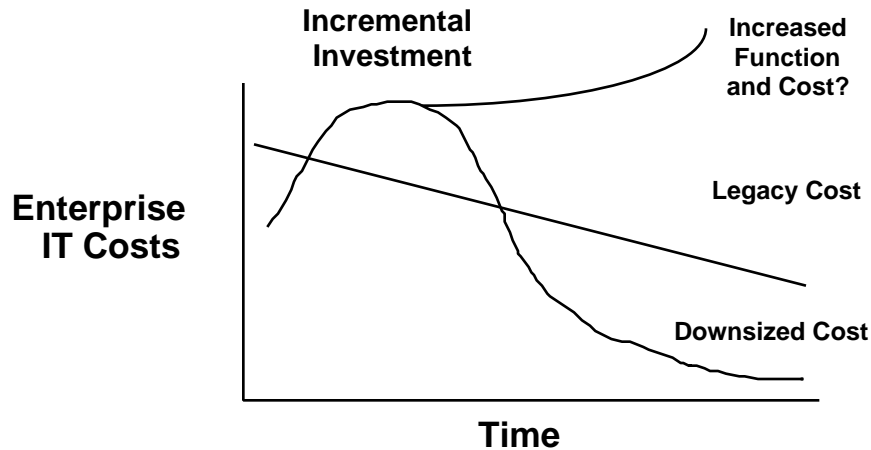
IT organizations cannot effect these changes overnight. A staged approach will work best. We believe that IT organizations can best evolve toward the "new IS" organizational model in stages. The first stage requires building credibility with users by stabilizing current operations and client service processes. The second stage entails becoming the delivery organization of choice, and targets the applications development processes. The final stage involves adding value to the business through management and human-resources processes, and through organizational alignment with the business.

Focus efforts on building architecture and infrastructure around a few (one to three) key, strategic projects while providing guidance to the rest. Gain user credibility, confidence and buy-in over time. Then gradually extend to other BUs.



How will transitions change the economics of computing? How can costs and investments be optimized at each stage of transition?

Reader Notes



Issues:
Capital vs. labor costs
Disposition of legacy assets

Source: Gartner Group

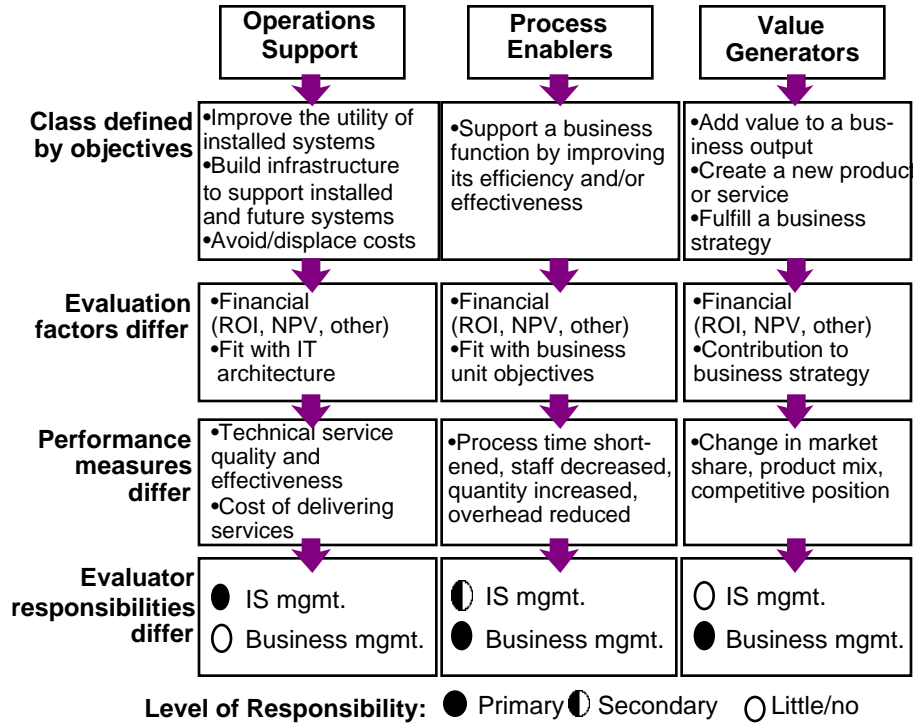
Migration strategies often presume lower hardware and software costs. This is partly due to comparing products based on commonly used industry metrics, such as dollars per MIPS or TPC-A. These metrics are useful for comparing components of the system; however, they only capture a fraction of the total costs. Only a total cost of ownership approach captures the total costs.

Traditional cost-of-ownership analysis addresses acquisition and operating costs for hardware and software. Migrating to distributed computing may lower hardware and software costs, but it also changes the IT cost structure. "Glass houses" are capital-intensive (i.e., hardware and software); end-user computing is labor-intensive. In addition, the incremental costs of acquiring and operating new technology (as well as switching costs) are usually not offset by a corresponding reduction in legacy assets.

For most large users, transitions to distributed, heterogeneous computing increases functionality and IT costs.



Through 1998, the most successful transition strategies will be based on business operations improvement and justified by business management; strategies based on lower IT costs will mostly fail (0.8 probability).



ROI Return on investment
NPV Net present value

Source: Gartner Group

Key Issue: How do transitions change the economics of computing?

Companies invest in IT for many reasons. We suggest an IT evaluation framework that classifies investments according to their primary objective.

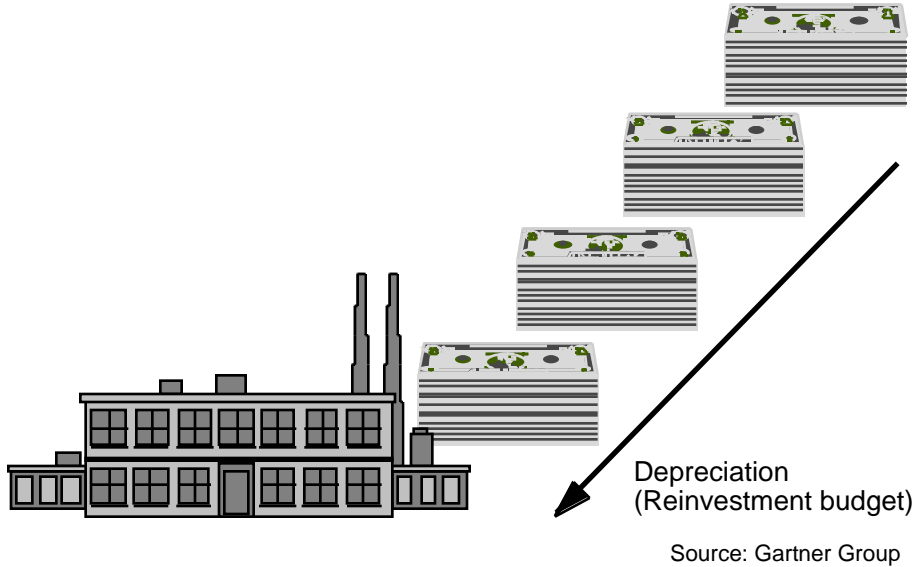
Operations support: Infrastructure (e.g., networks) and utility systems (e.g., the data center) may result in increased functionality and/or reduced service costs or, more strategically, provide flexibility to the company in responding to future business requirements.

Process enablers: IT used to significantly enhance or re-engineer a business function. The value may be compressed process time, increased output and/or improved work quality. These benefits are realized by the business organization, not IS.

Value generator: An investment that is directly responsible for generation of new business value, either because it is a marketable product (e.g., ATM systems) or the fulfillment of a business strategy. This class of investment is typically mandated by business management.



By 2000, customer-value pricing will be the mechanism by which chargeback is used. Eighty percent of enterprises will use a form of chargeback (rather than a budget-only approach) for funding IS investments in infrastructure (0.7 probability).



Key Issue: How do transitions change the economics of computing?

IT infrastructure has typically been planned on an expense basis, regardless of the budget pools or accounting treatment used. This model is breaking down in the era of client/server systems. The “information economy” requires a different approach to IT investment. Customer value can be created by aligning IT systems with current work processes, delivering information-rich services or IT-imbedded products. Process change and value-based products and services often require reinvestment.

Just as an enterprise’s other productive capacity is planned in such a way that 1) there is a business plan supporting what it will produce (that may or may not be fulfilled in the real market), and 2) capital cost allowance (depreciation) is reserved for its eventual upgrading or replacement, so too must IT infrastructure be handled. IT infrastructure shows a return only when directly put to revenue-generating purposes (e.g., a bank’s ATM network). The balance is a common facility which must be paid for, and whose rates must supply the needed future capital for its upgrading and replacement.



Designing a Winning Transition Strategy

Reader Notes

1. Build the strategy around the needs of a few, key projects.
2. Choose the projects based on business contribution and a business ally.
3. Identify the business drivers and link the architecture to them.
4. Define the architecture based on the needs of these projects. Publish it as the enterprise's strategic direction and market it as guidance to the rest of the enterprise.
5. Develop service-level agreements to focus on delivery and support requirements and the metrics of success.
6. Redefine the roles and responsibilities of central IS, business unit IS and end users, i.e., enterprise governance.
7. Re-engineer a portion of the IS organization around the delivery and support processes for these few projects.
8. Retrain key technical and managerial personnel within these projects.
9. Demonstrate architectural and organizational relevance with these projects before extending them to the next projects.
10. Think strategically; act tactically. Focus on first-year results.

Migrations from legacy to distributed, heterogeneous computing changes not only technology but IS organization. Compounding these changes: 1) the legacy mission and workload continues — there is no “timeout,” and 2) there are limited additional resources (and, frequently, fewer) to accomplish the change. The most successful transition strategies that we observe are based on focused results — one or two key, strategic applications that deliver significant value to the business.

We recommend that users incorporate these principles into their strategies. They focus on business contribution supported by business management. They pragmatically define the architecture and communicate the strategic direction without solving every standards issue or controlling every user. They redirect a subset of IS resources towards a service-based organization while continuing the legacy mission. They build experience with the new model while limiting risk. And they demonstrate proactive IS leadership. It is a winning strategy.

