N1™ Grid Architecture Realized: Strategic Flexibility

Jason Carolan, Sun Client Services
Scott Radeztsky, Sun Client Services
Paul Strong, Software Marketing
Ed Turner, Sun Client Services

Sun BluePrints™ OnLine—May 2004

http://www.sun.com/blueprints

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95045 U.S.A.
(650) 960-1300

Part No. 817-7040-10
Revision 02, 5/13/04
Edition: May 2004

Please
Recycle

N1 Grid Architecture Realized: Strategic Flexibility

Note – This article is the seventh chapter of the Sun BluePrints™ book *Building N1 Grid Solutions: Preparing, Architecting, and Implementing Service-Centric Data Centers* by Jason Carolan, Scott Radeztsky, Paul Strong, and Ed Turner, which will be available through `www.sun.com/books`, `amazon.com`, and Barnes & Noble bookstores.

Part 1 developed a more detailed understanding of N1™ Grid solutions. It provided examples of how implementing N1 Grid architectures uses familiar design fundamentals. Part 2 introduced the importance of architecture as a central theme. The architecture serves to organize and illustrate the movement from an environment of server-centric applications to an environment of network-centric services. That architecture is the focus of Part 3, beginning with the concept of strategic flexibility.

Before defining this concept, it is important to review the importance of architecture to the goal of achieving the vision of the N1 Grid. The N1 Grid vision is not a replacement for doing good architectural design work. The N1 Grid vision is best realized by using architecture as the link between business decisions and the physical environment on which people, process, and technology decisions are carried out. This view highlights the value that an architectural approach has to deliver. The N1 Grid architecture ensures the solution is focused on the business problem.

The development of the N1 Grid vision, architecture, and products is deeply rooted in identifying the business problems to be solved. Extensive interviews with IT managers and operators consistently showed that cost and complexity ruled the data center floor. Specifically, those interviews revealed the following metrics:

- Average system utilization was 6 to 15 percent.
- Number of servers per administrator was 15 to 30.
- Number of projects deployed per quarter was 3 to 5.
- Overall system availability was only about 99.9 percent.

There is a clear opportunity to improve the business of IT. The N1 Grid helps the IT organization to operate more efficiently, and it can directly influence the business as a whole. The business gains new opportunities when IT can deliver services in an efficient and highly dynamic way.

Any architectural solution must first solve the basic IT problems of cost and complexity. Then, it must strive to develop business and IT linkage to its fullest extent. That is the central theme of the architectural solution presented here, and it is articulated in the concept of strategic flexibility.

Strategic flexibility is about what can be achieved in architecting the service-centric data center. It is best explained through example. In many IT organizations, there are silos of systems dedicated to supporting specific tasks. For example, an industrial company might use one set of systems to support an online ordering function and another set of systems support a billing system. In a typical business cycle, over 50 percent of their orders are received in the first month of each quarter. Because of that, the ordered systems are sized to support that peak. On the billing side, bills are run during the last two weeks of each quarter. The billing systems are sized to accommodate their peak. It would be much more efficient for the online ordering and billing systems to share the same infrastructure to maximize the investment. The load peaks certainly support this. Unfortunately, the general maturity of IT does not.

Strategic flexibility solves this problem. It would enable the linkage between the business need—be it ordering or billing—and the most efficient use of IT infrastructure to support that need. With a standards-based, virtualized, and automatically provisioned infrastructure, and a virtualized and automatically deployed services layer, the data center can support business needs on demand.

Another opportunity for strategic flexibility is in improving the life cycle flow of business services. Today, applications are brought to life in a silo of the IT organization called the *development silo*. Usually, the development silo is a distinct organization within the IT organization, and the work is done on dedicated systems. When an application is complete and it is ready to move into production, it is tested. After testing, the application or service moves into production. Because the application or service has unique infrastructure and application requirements, each and every move requires a new and complete system installation from the ground up. Strategic flexibility can leverage an automated infrastructure and application layer to dynamically create the environments needed to bring an application to life,

saving tremendous amounts of time. Additionally, the code could be seemlessly moved between the environments (FIGURE 1). The quicker time to delivery and the reduced demand on operational resources is a direct IT cost saving. In addition, using a development and test suite that can exactly duplicate the production environment results in a significant decrease in service outages caused by untested configurations. Thus, strategic flexibility addresses the costs and complexity of IT, and it delivers value to the business by vastly improving overall service availability.

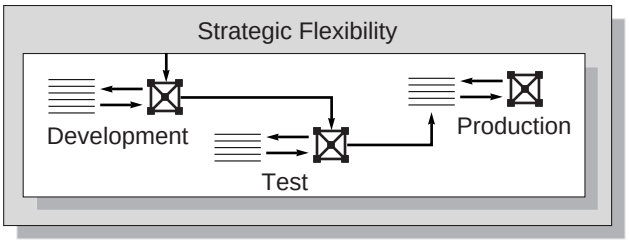

FIGURE 1 Life Cycle of Business Services

Achieving strategic flexibility is the goal of the architecture process. Because of the importance of strategic flexibility to the overall architectural discussion, it is meaningful to explicitly define strategic flexibility as:

> The ability to deliver a service-centric, systemic, and quality-driven, "just-in-time" IT infrastructure that maximizes the use of assets and minimizes cost and effort.

This definition focuses on two important aspects: the inherent business benefit of strategic flexibility and traditional IT efficiency goals. The combination of those two elements is central to the idea of strategic flexibility.

It is best to look at the second part of the definition first. The second half calls for the delivery of a just-in-time IT infrastructure that maximizes utilization of assets and minimizes cost and effort. The critical-to-quality factors (CTQs) for the N1 Grid require utilization improvement, decreased cost, and decreased complexity. Strategic flexibility focuses on these core issues. The first part of the definition is focused on deliverables that are beyond the traditional IT deliverables. That focus, expressed as a service-centric, systemic, quality-driven, just-in-time IT infrastructure, is an explicit challenge to deliver strategic business value. This business value is very important to the concept of strategic flexibility.

Linking core IT efficiency with strategic business value provides new and interesting opportunities. The industry typically drives advances into the IT organization on a nearly ongoing basis. While gradual improvements in efficiency and functionality are always appreciated by the business, they are rarely compelling business events. Strategic flexibility changes that. Strategic flexibility goes beyond traditional IT value to create strategic business value.

This discussion of the definition of strategic flexibility raises some questions. What does strategic business value mean? How does the linkage between IT efficiency and strategic business value really work? Does operational maturity still matter? All of these are good questions and each deserves an in-depth discussion.

Strategic Business Value

For the majority of companies, IT is a component of the business, not the business itself. IT services enable a company to build things, deliver things, and sell things, but IT services are usually not the product that is sold to the end consumer. A company will build cars, ship packages overnight, or sell consumer goods in retail outlets to generate the profits necessary to satisfy its shareholders and owners. In all of these cases, the IT organization provides a varying degree of services that support those primary activities. These supporting IT services vary from the mundane and necessary to the truly strategic. As expected, the value of the delivery of those IT services varies accordingly.

Strategic flexibility promises to deliver a service-centric, systemic, quality-driven, just-in-time IT infrastructure in support of the business. This rather technical definition is aimed squarely at providing a greater degree of strategic IT services to the business. Strategic services are IT services that can truly bring competitive advantage to the company that executes them appropriately, resulting in the delivery of strategic business value. The focus on delivering strategic business value through IT is a very important. There is a lot of discussion in the industry about whether or not IT can actually deliver strategic value to the business.

The May 2003 edition of the *Harvard Business Review* included an article by Nicholas G. Carr titled “IT Doesn’t Matter.” The article caused a firestorm in the IT industry because it proposed that IT no longer delivers strategic value to the business. The following quote from the article summarizes Carr’s thesis:

> “Information technology is best understood as the latest in a series of broadly adopted technologies that have reshaped industry over the past two centuries—from the railroad to the telegraph to the electric generator. For a brief period, as they were being built into the infrastructure of commerce, all these technologies opened opportunities for forward-looking companies to gain real advantages. But as their availability increased and their cost decreased—as they became ubiquitous—they all became commodity inputs. From a strategic standpoint, they became invisible; they no longer mattered. That is exactly what is happening to information technology today, and the implications for corporate IT management are profound.”

This article resulted in a flurry of responses as the ability of IT to deliver strategic value was debated. In one such response, “IT Does Matter” (*Computerworld*, May 2003), Patricia Keefe responded to Carr’s thesis with the following comments:

> “We still need IT leaders who can think conceptually, who can dream of ways to further business goals via technological advances. No matter what your present circumstances, you need to keep an eye on the future. Carr advocates taking a more defensive posture toward IT investments. But sometimes the best defense is a good offense. What’s needed now is to go back to the chalkboard and figure out the best strategy.”

While the academic argument of whether IT can deliver strategic value in the future continues on both sides of the issue, there is no doubt that companies worldwide are under ever increasing pressure to deliver more, better, faster, and cheaper. This drive to improve business services drives all of the components of that business to contribute to that improvement. So the IT organization must step up and deliver.

The best summary of the IT situation is given in “Disconnecting IT from Reality” by Thornton A. May (*Computerworld*, December 2003). While only subtly referring to the debate over the strategic value of IT services, May sums up the current global opportunity for IT very succinctly:

> “We stand at a moment unprecedented in the evolution of IT. I use the word unprecedented because at no previous time in history has technology possessed more promise or its value been so seriously doubted.”

The ongoing global economic climate demands continuing improvement in business services for companies. As a functional component of the business, IT must continue to be a part of the delivery of strategic business services. By focusing beyond the commodity nature of its own industry and remembering that innovation matters, IT can move beyond doubt by the successful rollout of the very services it is not expected to deliver. Future IT services can be strategic to the business, and with the concept of strategic flexibility, as delivered by the N1 Grid, IT can continue and even improve the opportunity to deliver a direct and compelling impact to the business it serves.

IT Efficiency and Strategic Flexibility

How does the linkage between IT efficiency and strategic business value really work? For an IT professional mired in a lack of IT efficiency, that question is tough to answer. It is good that half of the definition of strategic flexibility is focused on pure IT efficiency. That is a core requirement and unfortunately needs to be the focus of many IT organizations first and foremost. But there is more to IT than just improving core efficiency. There is a tremendous opportunity to deliver strategic value to the business. The opportunity exists, but the question remains, “How?”

The “how” of business and IT linkage is also a tough question for the business executive who never sees any strategic value from that same inefficient IT organization. Gains in IT efficiency do not offer anything compelling for the business. There is promise in strategic flexibility if the IT organization could deliver a service-centric environment. But, how does the business pull that value up through the IT organization? The answer is found in the two-part definition of strategic flexibility: It is the combination that matters.

The IT organization should not be expected to realize strategic flexibility all by itself. Likewise, the business should not be expected to articulate all of the requirements and supply all of the funding to make it happen. The IT organization should recognize the value of delivering strategic flexibility and push it toward delivery. The business should recognize the potential value of strategic flexibility and pull it in where ever possible. That combination—that linkage—truly enables the achievement of strategic flexibility.

Leverage and OMCM

Operational maturity still matters. In the introduction to Part 2, “Preparing for N1 Grid Solutions” on page 35,” the Operational Management Capabilities Model (OMCM) was defined as a key process that provides a measure of the people, process, and tools that give an organization the ability to deliver IT services to an agreed-upon service level in a predictable fashion with acceptable risk and cost. Given that definition, how can this model be leveraged?

It is clear that to deliver the kind of value expressed by strategic flexibility, the IT environment cannot be operating in “chaos” mode. Beyond aspiring to improve IT efficiency, there is value in operational maturity and the goal of strategic flexibility. The goal of strategic flexibility can be leveraged as an incentive to achieve a greater level of operational maturity. Like the business linkage, there is a trade-off between strategic flexibility and operational maturity. One way of expressing it is to say that strategic flexibility both requires and enables operational maturity.

Remember that IT efficiency is the foundation of strategic flexibility. You cannot achieve the goal without it. Analyzing the people, process, and tools in an organization associated with each OMCM level can help an organization prioritize its improvement efforts. The goal of strategic flexibility and the value it delivers is clear, so leverage it to pull IT efficiency through to higher OMCM levels.

Strategic flexibility enables strategic business value to be achieved, based on the foundation of core IT efficiency. With ever-improving operational maturity and a firm link to and from the business, the path of the IT organization is clearly focused on the potential that realizing the N1 Grid can deliver. The chapters that follow will provide directions for achieving the promise of strategic flexibility. Those directions, or guiding principles, are delivered through architecture.

Architecture and Strategic Flexibility

So far, the vision of the N1 Grid has been presented. In addition, an extensive review of the many tasks that an organization must consider has been presented. Finally, there is a specific goal of strategic flexibility. With all that as background, the temptation is to move straight to an implementation design. That would be a mistake because architecture matters.

Although there is broad expertise in system design and implementation in the personnel that typically manage a data center, too often there is a limited focus on architecture. That is not an indictment against the IT operations staff. It is simply a result of typical daily operations. The architectural work used to develop most of the IT services in production is done outside of the data center. Data center personnel typically focus on only system design or implementation. Because the benefits of the N1 Grid are focused at the data center and a robust architecture is necessary for its success, it is important to articulate the difference between design and architecture.

Dot-Com & Beyond (Sun Professional Services) discusses the issue of design versus architecture as follows:

> "The terms 'system architecture' and 'system design' are often used interchangeably, but in fact deal with two very different engineering processes.
>
> In approaching the distinct but related tasks of architecture and design, it is important to remember this key principle: a system's design is constrained by its architecture. The system architecture establishes the big rules that must be followed by the system design (which in turn determines how the desired functionality should be realized by the system implementation). The process of architecture ends, and the process of design begins, when enough big rules have been established to ensure that as long as the design complies with those rules, the system will satisfy the established QoS requirements.
>
> System architecture focuses primarily on the overall structure of a system, identifying major components and their relationships. The system architecture seeks to define:
>
> - How overall processing should be *decomposed* into component parts
> - How major components should be *organized* with respect to one another
> - How major types of operations, such as communication and storage, should be *mechanized*

> Thus, the architect focuses on issues such as which components should communicate with one another, which should be visible to one another, and which should be replicated; how components should be distributed, how they should talk to one another, and where they should be stored."

Although the N1 Grid system design, and ultimately its implementation, is vitally important, an architecture must precede that design. As stated, the architecture, focused on the goal of strategic flexibility, strives to set the "big rules" by which the end design is guided. Through a process of decomposition, followed by organization, and finally mechanization, a valid architecture comes to life. That architecture, the fundamental and unifying system structure for the N1 Grid, is the primary focus of the remaining chapters in this part of the book.

However, before focusing on the specifics of the first component of the N1 Grid architecture, a review of the development of that architecture is presented. This development uses the SunTone[SM] Architecture Methodology (SunTone AM) to focus on the key business drivers and the near term business return on investment (ROI) that is necessary to achieve strategic flexibility. Finally, the functional requirements, backed up by use cases, provide the specifics of the N1 Grid architecture.

SunTone Architecture Methodology

The SunTone AMis a significant guiding element in the development of the N1 Grid architecture. Because this methodology was covered in more depth earlier in this book and because complete references are available in Appendix B, only a quick review is provided here. Again, *Dot-Com & Beyond* provides the general reference for the description below.

The SunTone Architecture Methodology provides a detailed architectural process by which a truly services-driven, network-accessible, N1 Grid environment can be developed. It requires that system architecture be comprehensively addressed before system design because the architecture establishes the overall structure that must be adhered to by each portion of the subsequent design. The SunTone Architecture Methodology is:

- Use case focused
- Architecture centric
- Iterative and incremental
- Systemic qualities driven
- Patterns based

The SunTone Architecture Methodology is the practical realization of a team-oriented approach to specifying, constructing, and assembling software and hardware components into a working system that meets a well-defined need. This methodology, while not the only architectural guidance available, is certainly focused on the specifics needed to adequately address the real business problems that the N1 Grid solves.

KBDs and CTQs

Meeting a well-defined need is a part of the SunTone Architecture Methodology definition. Properly articulating that need is key to developing the solution that solves the problem. Identifying the key business drivers (KBDs) and critical-to-quality elements (CTQs) of the solution is essential to properly identifying the need. While research-based work developed the KBDs and CTQs of the N1 Grid problem, in an interview with *Information Week*, Sun Microsystems CEO Scott McNealy summarizes the need in very practical terms:

> "We take our cues from our customers. What's important to them is important to us. So our No. 1 priority right now is saving them money. The best way to do that is by reducing complexity. People are just absolutely going nuts with the complexity of the current computing environment. A recent study shows that about 10 percent of IT costs are hardware, 10 percent software, and the rest administration and training. Today, a system administrator can manage between 15 and 30 systems; it should be 500. System utilization is around 15 percent; it should be 80 percent. It takes weeks to deploy a new network service; it should take days or even hours. We need to fix those numbers, and we're working on it." (*Information Week*, January 2003)

Those key business drivers provide the focus of developing the N1 Grid architecture. Reducing cost and complexity in the data center is the focus that drives the goal of achieving strategic flexibility. The functional architecture shown in FIGURE 2 provides the working system to achieve that goal.

Immediate Return on Investment

An important feature of any robust architecture is flexibility. While numerous KBDs and CTQs can be identified, each individual IT environment has different needs. Those needs must be supported by the architecture in a flexible manner. The architecture presented in Part 2 is discussed in an orderly flow; however, it is not a prescriptive flow. The flexibility generally enables an IT organization to do what it wants when it wants.

With that flexible architecture in place, the question becomes, “With so many choices, where do I begin?” Referring back to the discussions about IT supporting the business, the focus should be on efforts that deliver the most immediate ROI. Utilization can be the issue in one environment, while time to market of new services can be the problem in another. Availability and IT automation are two examples of efforts that can result from the implementation of a flexible N1 Grid architecture. The key to achieving true strategic flexibility is tight alignment with the business. Use that alignment to prioritize overall IT efficiency improvements.

Use Cases

One of the cornerstones of the SunTone AM is that it is use-case focused. Use cases provide the real-world in-depth analysis of the problem that directly supports the efforts to develop the functional requirements. Use cases were defined in Chapter 4 as the functional scenarios that describe the complete flow of an operation from the perspective of an actor. Actors are entities that are external to the system, typically users or other systems. In use-case-driven design and development, every attempt is made to prioritize design and development around the realization of complete, end-to-end use cases. The advantage of this approach is that it keeps the project team focused around continuously delivering functionality directly related to the end product being developed.

A significant number of use cases have been developed in support of N1 Grid. They range from academic use cases developed within Sun to various real-world customer use cases. The use cases that support the N1 Grid are very important, but their specific review is not the focus here. This book does, however, rely heavily on the various use cases developed as direct support in the development of the functional architecture described in the next section.

Functional Requirements

The SunTone AM has guided the development of the architecture that supports strategic flexibility. In alignment with that methodology, there is clear focus on the key business drivers and use cases have been leveraged extensively. This background leads to the development of the strategic flexibility functional architecture.

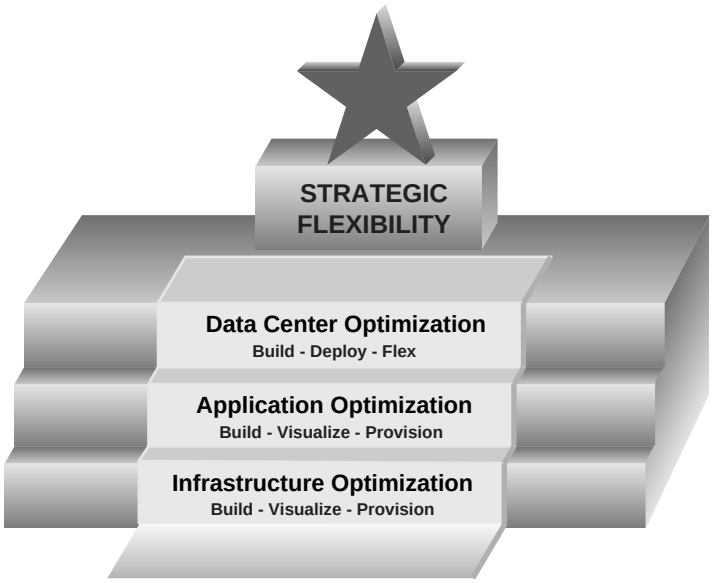

FIGURE 2 Strategic Flexibility Functional Architecture

As mentioned, there is a particular flow to the description of the architecture. While still flexible in the ways it can be implemented, the architecture is focused on a building block approach. Each block adds value that can then be increased as additional blocks are added. This building block approach is aligned with the description of layers from the SunTone AM.

The development of the first building block starts in the infrastructure layers of hardware and operating system. The process continues as the focus shifts to the application layers. Again, specific and common components are developed during this phase. Finally, the entire layered stack is addressed as a functional whole.

The building block approach of “common themes with specific variations” is an excellent summary of the functional requirements. That approach will become apparent as the specifics of the strategic flexibility architecture are reviewed in detail. The three specific functional components of the strategic flexibility architecture are:

- Infrastructure optimization

  Leveraging the compute, network, and storage hardware, infrastructure optimization is delivered through the hierarchical building-block architecture flow of *build, virtualize, and provision*. This component provides the foundation for strategic flexibility. Infrastructure optimization is the focus of Chapter 8.

- Application optimization

  Using the same building-block architecture, this component focuses on dramatically improving the efficiency of delivering business services through the automated deployment of application components. This second component of the architecture provides the critical functionality required to tie those business services to the IT infrastructure. Application optimization is the focus of Chapter 9.

- Data center optimization

  Through a focus on delivering strategic flexibility, data center optimization refactors the data center based on business demands. Directly dependent on infrastructure and application optimization for its foundation and efficiency, data center optimization enables truly business-focused IT operations. Chapter 10 explores the evolution of IT roles brought about by this business-flexible computing environment. Data center optimization is the focus of Chapter 10.

Strategic Flexibility was defined as the ability to deliver a service-centric, systemic-quality driven, just-in-time IT infrastructure that maximizes utilization of assets and minimizes cost and effort. This functional architecture (FIGURE 2), delivering the combination of infrastructure, application, and data center optimization, enables strategic flexibility and ultimately the N1 Grid. Although this functional architecture view is central to understanding and ultimately realizing the N1 Grid, it is not the end of the architectural process. The architectural process also includes identifying the technical requirements that provide the bridge to design and implementation. Because these are very specific to an implementation, they are not discussed here.

Applied Architecture

As stated, the completion of the functional architecture is not the completion of the architecture process as a whole. The architecture descriptions in this book do not provide a detailed view beyond the functional architecture. Given this, it is important to briefly review the required elements to transition from architecture, through design, to implementation. Components of this transitional or *applied* architecture are presented in this section.

Technical Requirements

The functional architecture is focused on delivering the combination of infrastructure, application, and data center optimization. This combination leads to the development of a strategically flexible functional architecture. The functional requirements summarize what that architecture looks like. The next step in this process is to identify how the architecture is delivered.

The technical requirements are the most granular requirements defined as part of the architectural process. The technical requirements are detailed specifications that explain exactly how the functional architecture will be realized in a specific implementation. Technical requirements are most often specifications to the individual product level. Due to their specificity, these requirements are unique to each individual IT environment.

The review of the different levels of the functional architecture in the following chapters does not discuss the technical requirements because technical requirements by nature are implementation specific. However, product examples are presented throughout that could be part of the technical requirements of an actual implementation. In the examples that demonstrate the application of a particular functional component, the technical requirements of same as the components.

Constraints

With the technical requirements and a solid functional architecture as a foundation, design and implementation should flow naturally into the next steps, with the exception of constraints. Basically, anything architecturally significant already in the environment or predefined as being in the environment can be a constraint. For example, if an IT organization has out-sourced the operation of their data center, that would be identified as a constraint. Constraints are static, but they are generally identified as the environment to which the architecture must be integrated.

Constraints come in different forms. To further explain the idea, three categories of constraints are identified and an example of each is given:

- Development and team constraints

  Data center system administrators are junior personnel due to recent employee turnover.

- Environment and technology constraints

  The Sun Java™ system server is the IT core standard for J2EE™ deployments.

- Delivery and deployment constraints

  A third-party consulting company has an outsourcing contract for application support.

Whatever form constraints come in, they must be identified and understood as part of the transition to implementation.

Iterative Approach

One of the final architectural issues that must be carried into implementation is the idea of iterations. An iterative approach is common in the architecture and development of software. The idea is generally summarized as “make your mistakes early and often.” This somewhat humorous view of solving this problem is derived from years of experience delivering complex software architectures. It has been proven that the end-to-end build time can be greatly reduced by iterating the design, build, and test flows. In Chapter 4, the concept of iterations was discussed as a key component of the SunTone AM. That discussion stated that integration testing should occur early, often, and throughout the development cycle, enabling the early detection and correction of requirements, design technology, or usability issues.

Following an iterative approach enables the system to evolve through a series of releases. Each release adds to the overall functionality by building on a solid foundation that has been fully tested. By identifying and addressing issues early, the iterative approach enables a more refined product, delivered with lower risk.

Beginning the Transition

With the architecture process completed, the transition to design and implementation can begin. The danger is that the transition might seem overwhelming. The architecture process can point out many issues and opportunities. There are only a limited number of resources available to solve these issues. With all the options, paths, and limited resources available, what is the best approach? An iterative approach is always best.

The opportunity to deliver real business value through strategic flexibility is real. It is important not to be trapped in “analysis paralysis” or other indecisiveness. There are a lot of options. Focusing on business-based ROI can help to focus efforts. The KBDs, and the entire architecture process they drive, can also focus your efforts.

Future of Strategic Flexibility

Before closing this chapter, it is worth while to preview what is possible in the future beyond achieving strategic flexibility. Looking forward, utility, grid, and policy-based computing can deliver even greater business advantage over and above strategic flexibility. The N1 Grid vision involves redefining the nature of the data center. This section contains a sampling of the discussions about utility, grid, and policy-based computing in Chapter 11.

Utility Computing

Utility computing (UC) is difficult to define because it seems to mean something different to every system vendor. UC can be narrowly defined as simply aligning pricing models to data center resources (that is, pay-as-you-go models). The hardware pay-as-you-go model is usually a first step. However, UC is really about aligning all IT costs with the business, including transactions, service levels, and most importantly, user costs. UC has a role in the ability of the business to create a model that supports strategic flexibility. This enables the business to move towards a service-based utility.

Grid Computing

A grid is a collection of computers that are available to perform various tasks as part of an autonomous system. The grid has a workload management system and specially written applications that enable it to parse out tasks and gather the results. This distributed system enables careful control over the quality of service, enabling grids to perform critical functions. Grids can span hundreds or thousands of nodes, and they require little ongoing care. A node can leave the system and re-enter it with little effect. The central workload management system simply stops forwarding tasks to nodes that have left the system.

The N1 Grid vision provides resources for services in a more efficient and flexible manner. In many ways, the grid technologies that have been used over the last decade are aligned with the N1 Grid vision for the data center. Grid-based computing has been around for a number of years, and because it can perform some impressive tasks, it has grown quite popular. In fact, it is measurably more successful than utility computing in the number of implementations. The N1 Grid can be used for grid computing, just as it can be used for other business services, such as web services and data warehouses.

Policy-Based Management

The world of computing is about to change. As more and more devices become connected, the way the systems need to be managed will also change. Much of the focus in device technology over the next decade will focus on increasing the intelligence of the system, whether the system is a cell phone, a Bluetooth connection to a laptop or car, or a new system like the data center based on the N1 Grid software.

Today's N1 Grid products concentrate on automating many of the manual tasks performed by administrators and providing virtualization and automated provisioning of resources. Most of these tasks are initiated by a human, based on some specific set of policies. Today, policies can be written down, in configuration files, or perhaps in a person's head. System vendors, including Sun, are rapidly moving towards enabling these policies to be specified in business terms so that the holistic system can manage itself to these policies.

Summary

Discovering the value of strategic flexibility involves IT and business linkage and an architectural view. Chapter 8 begins a more detailed discussion of the functional architecture. As the first step in the building-block architecture, infrastructure optimization builds the foundation of strategic flexibility. As this step and subsequent steps are taken, the details must not obscure the vision. The details are important, but the goals are principles that guide the success of implementing a strategically flexible architecture. Simply summarized, the architecture:

- Links business needs to IT through strategic flexibility
- Enables IT through the value of strategic flexibility
- Guides IT through the architectural process
- Refactors IT through the functional architecture
- Saves IT cost through the implementation of the architecture

About the Authors

Jason Carolan is a Senior Architect and Principal Engineer at Sun Microsystems. He has spent most of his five years at Sun in Sun Client Services, focused on developing solutions for Sun's customers, systems architecture, and improving architectural quality through patterns. Jason joined the N1 Grid software team over two years ago, contributing to many of the early internal and external documents. He has also been responsible for the design of the Service Delivery Network Architecture, which is Sun Professional Services[SM] program's data network architecture standard—one of the best examples of modular architecture. He also speaks regularly at conferences around the World about network design, security, and the N1 Grid architecture. Jason lives in Colorado where he enjoys the amazing snow and sun.

Scott Radeztsky is one of Sun's Principal Engineers. His six years at Sun have been focussed on architecture, performance and tuning, education, and working with customers to navigate uncharted architecture, technology, and operations territory in novel solution deployments. As examples of this, Scott has had the unique opportunity of directly architecting and building some of Sun's first internal and external customer-based N1 Grid software environments and is now focused in the areas of edge computing and Sun's next generation of systems. Before joining Sun, Scott earned a Ph.D. in Particle Physics from the University of Wisconsin, using the Web and related technologies for analysis and collaboration since their inception. Scott is grateful for the patient support and understanding of his family during this book creation effort.

Paul Strong is a Systems Architect at Sun Microsystems, working in the N1 Grid software product group. He has spent most of his seven years at Sun focused on resource management and availability and joined the N1 Grid software team at its inception, over three years ago. This is Paul's first official authoring credit, although he is an uncredited contributor to the Sun BluePrints book on resource management. He also speaks regularly at conferences. He is a native of England, transplanted to California, is a keen musician, and holds a degree in Physics, specializing in Astrophysics, from the Victoria University of Manchester, England.

Ed Turner is an Architect in Sun's Technology and Solutions Organization, focused on supporting major customers in the telecommunications industry. Leveraging his seven plus years of working various technical and management roles at Sun, he focuses his efforts on guiding his customer's strategic architecture. Ed had the unique opportunity of directly architecting and building some of Sun's first significant customer-based N1 Grid software environments. When not spending his off hours with his wonderful family, Ed enjoys long distance motorcycle touring. Ed resides in the Atlanta, GA, area and has a BSCS degree from the University of Maryland.

References

- Carr, Nicholas G. “IT Doesn’t Matter.” *Harvard Business Review*, May 2003.
- Keefe, Patricia. “IT Does Matter.” *Computerworld*, May 2003.
- May, Thornton A. “Disconnecting IT from Reality.” *Computerworld*, 2003.
- Sun Professional Services: The .com Experts. *Dot-Com and Beyond: Breakthrough Internet-based Architectures and Methodologies.* 1st Ed. Santa Clara: Sun Microsystems Press, a Prentice Hall Title, June 2001.
- McNealy, Scott. “CEO Visions: Toward Saner Computing.” *Information Week*, January 2003.

Ordering Sun Documents

The SunDocs℠ program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals through this program.

Accessing Sun Documentation Online

The `docs.sun.com` web site enables you to access Sun technical documentation online. You can browse the `docs.sun.com` archive or search for a specific book title or subject. The URL is `http://docs.sun.com/`

To reference Sun BluePrints OnLine articles, visit the Sun BluePrints OnLine web site at: `http://www.sun.com/blueprints/online.html`