♦ Toward a Vision for Network and Service Management

Kurt S. Silverman, Michael R. Brenner, and Gregory E. Shannon

In this paper, we consider some high-level frameworks for network and service management. This discussion is the result of a survey of the current state of this field. By offering these frameworks, we hope to encourage new system and application design, development, and delivery to fall into these or similar frameworks so that customer demands can be more easily met. As we advance an architecture for network and service management, it is important to review the main requirements of today's service providers: improved revenue via rapid service creation as well as reduced costs via reduced labor and skill needs in all operational areas across service fulfillment, assurance, and billing. Customers are demanding easy-to-run software that takes the complexity out of service and network offerings as the marketplace continues to underdeliver with respect to expectations. A major player in this marketplace, Lucent Technologies is organizing much of its network and service management development around the Telecom Operations Map (TOM) model. Essential to the job of providing software for next-generation networks are application-driven frameworks and continued execution of applications that, in the end, simplify and encourage service creation, delivery, and operations for service providers.

Introduction

The recent "dot-bomb" phenomenon has brought back to pragmatic reality many of the fundamental needs of today's service and network providers. These needs are, of course, business obvious—business and operational support systems that:

- *Encourage* new service creation,
- Are much easier to operate and maintain, and
- Require less integration expertise to insert into an operator's systems environment.

The rush of new network builds, insertion of new elements, and quick scaling of new Internet services has passed the point of most providers to have a working strategy for the underlying service and network management systems.

In a relatively static world of managed voice traffic in the 1970s and 1980s, service and network management software systems had years to catch up with user demands in a world that valued highly managed networks. This gave rise to large and functionally complex systems such as CRIS, TIRKS, SOPAD, BOSS, PREMIS, LMOS, CABS, and many others. With years of stabilization and procedures, these systems limped into the 1990s functionally rich but with huge deficiencies in flexibility due to the architecture of their day, their inability to cope with new services (much less encourage creation of new ones), and the layering of maintenance patches levied onto the software code base.

In the latter half of the 1990s, service providers created a new explosion in networks and network elements, far outpacing the ability of service management or network management software to "keep up." The result, of course, was rapid creation of point solutions for functional components, leveraging of older systems where appropriate, new custom builds by providers for functional pieces hard to come by, and an unprecedented amount of system integration

work. More often than not, the resulting systems run by providers are "patchwork quilts" of systems that are hard to operate and harder to maintain. The claims of open application programming interfaces ("open APIs") and standards-based systems did little to solve the basic problem of system interoperability. Arrival of newer technologies such as Common Object Request Broker Architecture (CORBA*), Distributed Component Object Model (DCOM), lightweight directory access protocol (LDAP), and now Extensible Markup Language (XML) still point out the fundamental issue and problem statement that software interoperability inside a patchwork quilt needs hand stitching. Once integrated, the burden of revving the software components without breaking the system still resides with the operator.

Now that the pencils are sharpening, providers are noting with alarm that software procurement, maintenance, and operational costs constitute as much as 70% of the actual network cost, and functionality, performance, and stability are still under user demand by a wide margin. Larger carriers are looking to architectural frameworks to help them plan and manage their way through this "squeeze play." They no longer have the resources to self-build and maintain, they seek less integration, and the products available to fill functional spaces are not designed to interoperate.

Against this landscape, Lucent Technologies is working to define an approach that yields a path toward realizing the unmet demand for open, interoperable, component-based software for service and network management. In this paper, we present a basic framework and commentary. As a premier provider of both networks and associated service and network management software, Lucent is, of course, in a unique position to close the gap between customer demand and the stark reality of the state of service and network management in today's environment.

Network and Service Management Vision

The vision for network and service management must be viewed against the basic shift from the voice-centric world of telephony to the data-centric, distributed-computing-based world of today. Perhaps most fundamentally, yesterday's world was network centric, with a focus on fault, configuration, accounting, performance, and security (FCAPS) management. In contrast, today's world is customer and business centric. Moreover, today's world demands simplicity of operation. It is no longer the case that service providers actually have an abundance of sophisticated network engineers on staff. Software becomes the key to rendering complex networks and services simple for users and service providers.

Today's network and service management problem is easily stated: Service providers want their vendors to have a strategy that addresses operations support systems, operations, network management, and service positioning across the entire product lines they offer and in multivendor environments.

Let us look at the basic functions for network and service management. Network management provides a network view of the world, and the associated operations tend to suppress differences in individual network elements and support network-wide functions such as maintenance and management of traffic congestion. Other functions in the network management space include topology, health, configuration, restoration, tests for connection/speed, inventory, purchas-

Panel 1. Abbreviations, Acronyms, and Terms

BOSS—billing and order support system
CABS—carrier access billing system
CORBA*—Common Object Request Broker Architecture
CRIS—customer records information system
DSL—digital subscriber line
FCAPS—fault, configuration, accounting, performance, and security
ISP—Internet service provider
LDAP—lightweight directory access protocol
LMOS—loop maintenance operations system
OA&M—operations, administration, and maintenance
PREMIS—premises information system
SLA—service-level agreement
SOPAD—service order processing and distribution
TIRKS—trunks integrated records keeping system
TOM—Telecom Operations Map
xDSL—any of various DSL technologies
XML—Extensible Markup Language

ing, planning/engineering, rerouting/redirection, mediation, and security. Basic functions for service management, which support a customer-centric and service-centric view, include account provisioning, service activation, service-level agreement (SLA) management and reporting, quality-of-service (QoS) monitoring, billing, workflow management, workforce management, trouble ticketing, revenue settlements, gateway to other providers, and customer contact management.

Today, no single vendor is meeting all service provider needs. The operator therefore has a multivendor software environment on top of heterogeneous networks of increasing complexity. What is needed is an understandable architectural framework for network and service management—that is, a coherent approach to Internet protocol (IP)-centric, multivendor, multiservice network management. The point of creating and promoting an application-driven framework is, of course, to encourage design, development, and delivery of applications that, at their core, promote a user-intuitive, application-interoperable set of working applications. The resulting systems need to be highly "componentized" so that the pain of "forklift" upgrades is minimized in the future and providers can leverage newly created products in this space.

The TOM Model

Let us look in a little more detail at network and service management, using the Telecom Operations Map (TOM) model[1] that is gaining acceptance in the telecom industry today. The TOM model, shown in **Figure 1**, presents an easy-to-grasp, role-based approach to mapping service management/network management functions. Horizontally, the TOM model considers management of the customer interface in terms of sales/service fulfillment, service assurance, and billing. Vertically, the model portrays functionality in terms of customer care processes, service development and operations processes, and network and system management processes. Of significance is what the TOM model does not show. It makes no real distinction as to the type of network (for example, optical, wireless, or circuit). This is intentional to show that the functions inside the model are network-type agnostic.

Lucent is organizing much of its network and service management development capability around the TOM model. This organizational alignment reinforces a role-based approach to the applications filling the functional spaces. In so doing, Lucent is directly addressing the need for the resulting software to require fewer operators and less operator skill on the part of its customers. Simply put, Lucent is organizing its strategy and development teams around fulfillment, assurance, and billing as opposed to network layer or type of network.

Customers are demanding easy-to-run software that takes the complexity out of service and network offerings, helping them lower labor and skill costs. They want their suppliers to offer solutions *and* to be integrators. This means suppliers must provide tools and frameworks for multiple vendors and develop multilevel partner programs. Traditional fault and performance monitoring are becoming more of a commodity. Provisioning, multidomain management, and bandwidth reservation are proving to be more difficult problem areas that no vendor has sufficiently "productized."

A coherent architecture and strategy for service, network, and element management can align with the items in the high-level view shown in **Figure 2**. At the bottom are common element interfaces. Just above them is the foundation—basic element management and network management. On top of that are value-added applications. Above those, we see closed-loop service management across the fulfillment, assurance, and billing functions of the TOM model. To the left is a robust component-based infrastructure. To the right is a need for a robust integration arm in the form of a partner program.

Figure 2 is a simple building-block view that encourages a classification of applications at one of four levels while pointing out the criticality of components for infrastructure across the applications and the requirement for robust integration to other software inside or outside the enterprise. Specific application software that is a point solution should easily fit into a block, and software that vertically crosses the horizon-

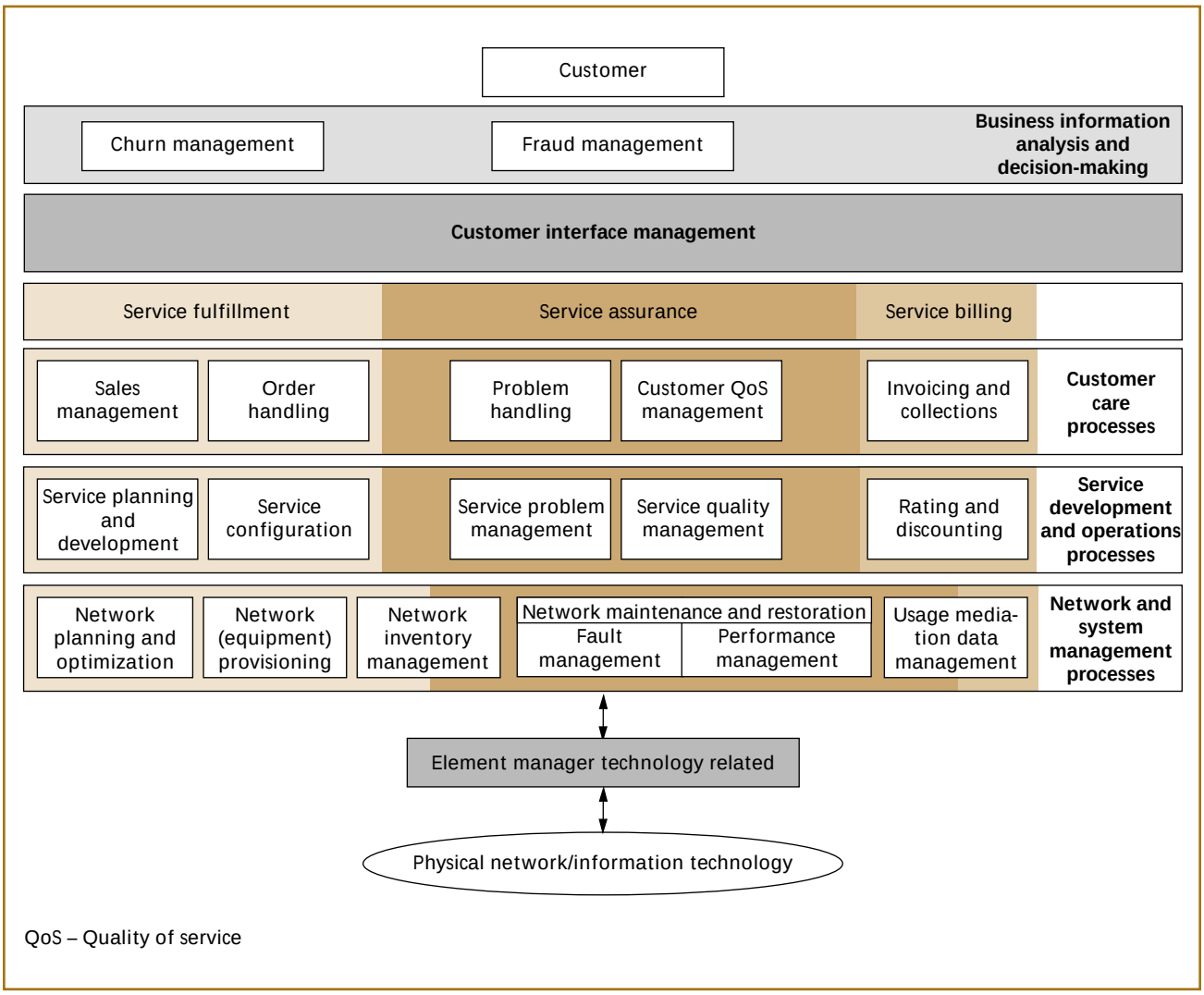


Figure 1.
The TOM model.

tal boundaries should be examined as to its robustness for the business owner, as it runs the risk of becoming the legacy system of tomorrow.

Having little or no software in the common component bin illustrates the lack of reuse across the enterprise. If, for example, the major applications all have their own mechanisms for basic alerting and alarming, it is difficult to actually perform basic fault and performance monitoring of the software itself. Similarly, if the enterprise is using a wide variety of provisioning engines for different services, it points to the inadequacy of a fundamental provisioning engine component. In the rush toward functionality and bringing software to market, reusable components have remained an objective, but a highly elusive one, particularly for small players who lack the resources or commitment to make real achievements in the area.

Too often, companies are too self concerned and self motivated to create and maintain an open integration arm. This leads to customers bearing the brunt of the integration and to various standards movements that often trail, not lead, customer concerns. The stark reality is that every customer has had and will have a patchwork of systems and has had or may have to resort to funding most of the integration activities, only to find it is an unrewarding experience to try to keep the patchwork quilt system revved into the future. The calling out of the integration/partner arm

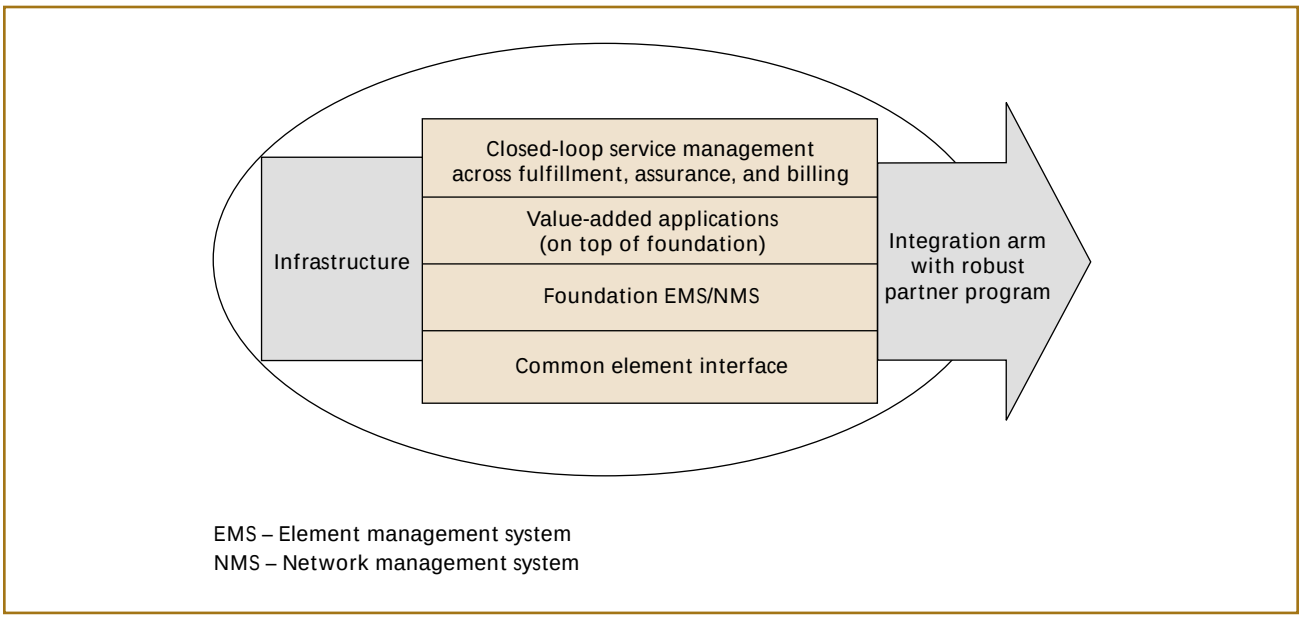


Figure 2.
Coherent architecture and market proposition for service, network, and element management.

proactively acknowledges and puts more burden on the supplier rather than the customer. Vendors must realize that it is good to partner—that a 100% product, off-the-shelf solution that comes close to client demand does not exist.

Before leaving this simple framework, let us emphasize that network and service management architecture is *viewed and measured* from the top, not the bottom, of the framework. The metric of success is customers and services, not element interface elegance. That is because the perceived value-add comes not from multidomain management, but from the enabling and measuring of new services.

To summarize to this point, we have presented the TOM model and a very high-level framework that can be used for evaluative mapping purposes.

Service Provider Requirements

Let us now briefly consider what service provider customers need and want with respect to a sample of services. This will give a sense of the functions inside the framework and lead us to our next fuller framework notion.

For any of various digital subscriber line technologies (xDSL), service providers need an order management and billing process, together with management of access inventory and line test functions. Hence, just for xDSL, a provider needs a fairly comprehensive set of service management and network management functions.

For wholesale bandwidth and bandwidth trading (to meet rapidly changing user demands in an era of abundant, commodity bandwidth), service providers need network planning tools, reservation systems, and robust network data collection linked to customers and their services. Predictive, preventive, and proactive network reconfiguration and restoration are critical to this service and customer segment. The services also require accurate inventory management that reflects actual network configuration and assignment. In addition, there must be linkage to the ordering process (that is, availability and interval) and network planning.

For Internet-based CyberCarrier services,[2] service providers need security management (both physical facility and network), which is essential for co-location services. These services depend strongly on inventory management, requiring inventory of facility, network elements, connectivity, and servers, as well as the relationships among them. It is important to provide online access to trouble resolution actions compiled from trouble history. Usage-based billing must support vari-

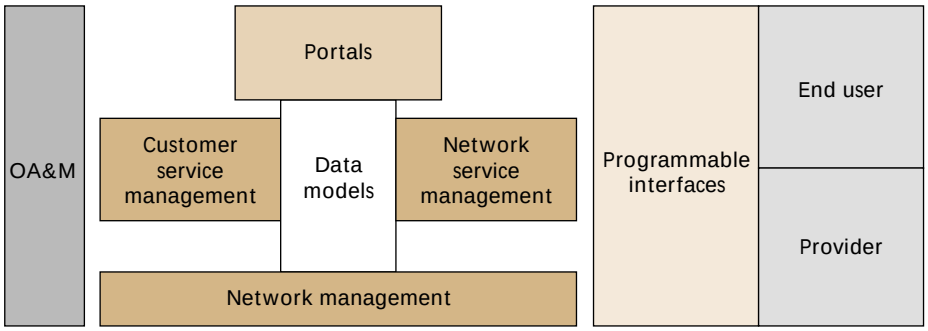


Figure 3.
Key architectural elements.

ous billing variables such as bandwidth, disk space, central processing unit (CPU) load, memory, pages cached, and storage space. In addition, software applications vended as a service need to be treated as network elements with full monitoring and restoration services. Finally, customer management must allow customers to initiate self-activation, self-administration, and retrieval of information.

For mobile Internet and voice services, service providers need automated, flow-through service provisioning, including over-the-air activation of service and administration of subscriber identity modules. This service requires fraud detection and churn-alert systems. A cellular-specific network planning tool is also vital to engineer the network and reengineer it based on changing conditions such as time of year, severe weather patterns, and changes in the wireless build-outs in the surrounding area. Service providers also need consolidated billing for subscribers of voice and data services as well as "hot billing" capability for settlements between operators for global roaming.

For global Internet service provider (ISP) services, providers need integrated billing and customer care, with on-line ordering, customer care, and billing information. They also need billing with a tiered rating structure based on usage, service bundles, and SLAs, including integration and tools to credit or debit accounts in case of SLA infringements. It is also vital to have support for service-level guarantees, encompassing network availability, reliability, and security, with aggressive penalties if guarantees are not met. In addition, network performance and monitoring of, for example, usage applications, growth, line loads, port utilization, network congestion, and number of packets must be available to customers. Finally, an Internet-based settlement and royalty system is essential for ISPs to track and comply with the myriad of revenue agreements made with content providers.

Lucent's Vision

Let us now look in more detail at Lucent's vision of network and service management architecture. Lucent's vision is based on offering a rich set of network and services management capabilities for service providers, with a next-generation orientation.

The key architectural elements of this vision are shown in **Figure 3**. To the left are common operations, administration, and maintenance (OA&M) functions for the software itself. To the right are programmable interfaces available to customers and third parties. At the top are user portals providing user access. Underneath are customer service management and network service management, and in the middle are the core data models. Network management functions are shown at the bottom.

This picture attempts to represent the large core functions of a hypothetical integrated service management and network management system. In many ways, this is the "dream" picture of most service providers—one never actually realized but hoped for. At the very center, of course, are consistent data mod-

els enabling both the customer-centric and network-centric applications to attach. All applications are accessed by Web-centric portals, utilizing common OA&M and an open architecture for extension and replacement. We now take this intuitive, idealized view and expand it using the concepts explained earlier in the TOM model and our layered framework to arrive at a fuller architectural framework.

This basic model is mapped via color coding on the more detailed architectural framework for the vision shown in **Figure 4**. This picture has been intentionally structured to present a Telecommunication Management Network (TMN)-like view of the management layers for readability. In reality, some of the layers and applications are collapsing in a way determined by the roles of the applications and the technology employed to develop them. This figure portrays an architecture open at four distinct levels, each depicted by horizontal gray bars. There are a presentation layer, two distinct or collapsible management application layers (a design choice), and a layer interfacing with the actual elements (resources). Each of these layers communicates through published, open, programmable interfaces. At the presentation layer at the top, communications will be mostly through XML or an XML variant. At the service workflow layer immediately below, communications will be mostly through exposed LDAP access, for simple data access, and transaction bus extensions, for more sophisticated access. For the network transactions layer below, communications will be mostly through CORBA, but the layer will also be open for anyone writing transaction-intensive applications to cooperate. For the resource-handling layer at the bottom, communications will be via industry-leading protocols such as distributed computing environment (DCE), simple network management protocol (SNMP), and basic command line interface (CLI).

Directly beneath the presentation layer at the top and running horizontally are the basic customer-oriented software modules for billing, ordering, service creation, and all the workflow distribution, together with the customer care, self care, and assisted care that goes with these functions. It is here that the compliance monitoring for SLAs needs to occur. In addition, two of the three major data stores (customer and service information) are located at this level, and the exposure to the outside world, aside from XML-based interfaces, is also through portals and a browser-based interface.

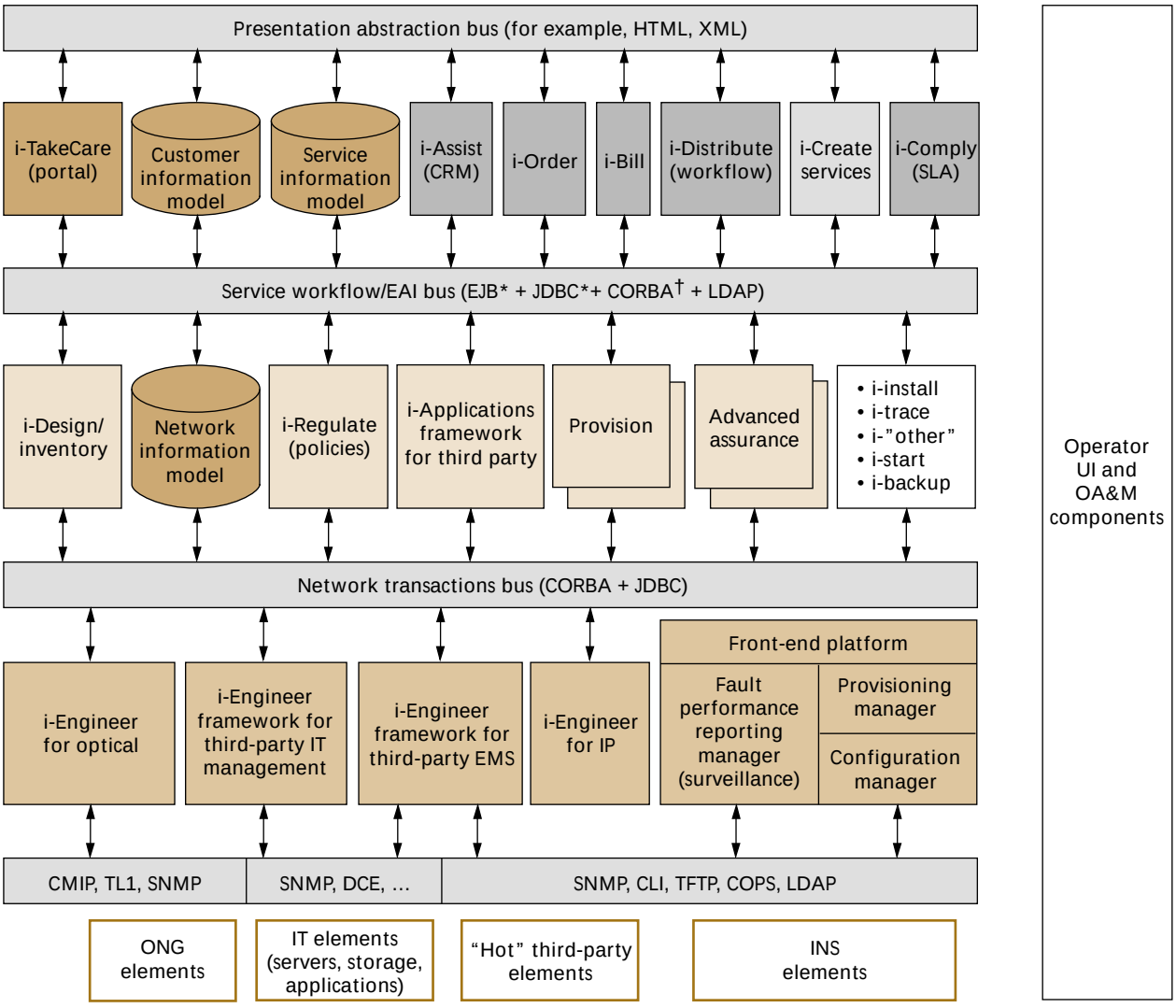
Directly under this layer is the network transactions management layer, which is oriented toward the network. In the middle of that area is the network information repository, along with the key applications of provisioning, design, and assign, with a framework to allow third parties to insert additional applications. There is also a "regulate" function for policy enforcement and protection. The advanced assurance module is for providers wanting to offer differentiated services, which require specialized assurances functions that Lucent, third parties, or the customers themselves may develop.

Below the network transactions layer is what could be regarded as the resource-handling layer, or the fundamental FCAPS management layer. Here we find network engineering, detailed provisioning, and detailed network configuration. This is where the fundamental monitoring and provisioning in detailed engineering terms occur along the network. These functions are shown just above the network element layer.

To help elaborate and further advance this simple, coherent network and service management architecture, we are working toward additional refinement and taking it to the next level of detail.

Conclusion

As we look ahead, some things seem inevitable. One is that the amount of networked infrastructure will get much more diverse. Moreover, a growing variety of intelligent devices will access next-generation networks, ranging from tiny wearable devices to medium-sized devices, such as set-top boxes and home control units, to large ones including distributed cyber-data centers and distributed video catalogs. This network complexity and the myriad of access devices will enable a wealth of services growing in both diversity and number. We will likely see remote video editing and bandwidth trading, personal health monitoring, specialized security monitoring, and all



CLI – Command line interface
CMIP – Common management information protocol
COPS – Common open policy service
CORBA† – Common Object Request Broker Architecture
CRM – Customer request manager
DCE – Distributed computing environment
EAI – Enterprise application integration
EJB* – Enterprise JavaBeans*
EMS – Element management system
HTML – HyperText Markup Language
JDBC* – Java* Database Connectivity
i – "Integrated, intelligent, and innovative"
INS – InterNetworking Systems
IP – Internet protocol
IT – Information technology
LDAP – Lightweight directory access protocol
OA&M – Operations, administration, and maintenance
ONG – Optical Networking Group
SLA – Service-level agreement
SNMP – Simple network management protocol
TFTP – Trivial file transfer protocol
TL1 – Transaction Language 1
UI – User interface
XML – Extensible Markup Language

*Trademark of Sun Microsystems, Inc.
†Registered trademark of Object Management Group, Inc.

Figure 4.
Technical framework.

sorts of network-based logistics support. Such seemingly far-fetched service offerings can ill afford to create yet another "soup-to-nuts" set of service and network management software. It will have to rely on a component-based framework approach to be achievable, scalable, and stable.

In summary, any vision of network and service management must be rooted in the wide range of needs of service provider customers. These needs vary according to different kinds of service providers. Thus, assessing and analyzing current customer needs is one essential step toward a coherent architecture for this vision. It is also valuable to use the TOM model for organizing development activities. The role-based approach of this model enables development of software products from a user perspective. Perhaps most importantly, using the TOM model to guide software development helps us develop more intelligent software that, in turn, reduces the requirement for service provider users to understand all the complexity and detail underlying networks. Service providers can thus reduce the sheer number of skilled users involved in network and service management operations.

This bears elaboration, because it is a basic paradigm shift. In the past, we applied our understanding of network complexity to the evolution of software to manage that complexity. Thus, it took many years to create a network software operation center in order to monitor what was then a very homogeneous network with a high premium on fault detection and performance grooming. In the emerging e-world, however, the basic reality is about creating and offering services. In this reversal of value assessment, software must conceal network complexity to enable customers to be more agile. This in turn puts more emphasis on networks themselves embedding intelligence, restoration, and configuration in order to simplify the task of the overlaid network management software. Many features being implemented directly into optical networks were formerly managed by ever-increasing complex software. The trend toward making networks themselves more intelligent is exactly the right strategy for enabling software to focus on encouraging new services.

Lucent is moving aggressively into this service-centric e-world with an emerging vision and reality for network and service management. We are organizing our software efforts into a customer-centric and network-centric frame of reference. We are proposing frameworks through this effort and working with the key standards bodies.

The job of providing software for next-generation networks will be accomplished not by one vendor or customer but by the interested community at large. Essential to doing that job are simple frameworks and continued execution of applications that, in the end, simplify and encourage service creation, delivery, and operations for service providers.

Acknowledgments

We thank William Goers and Chi-Yuan C. Wang of Lucent's Network Solutions, Harry Chiu of Software Strategy, and Rich Wetmore of Network Operations Software for their valuable comments and contributions.

References

1. TeleManagement Forum, "Telecom Operations Map," Document GB910, Version 2.1, Mar. 2000, <http://www.tmforum.org>.
2. M. R. Brenner, M. Chu, G. Gross, and M. Malek, "CyberCarrier Service and Network Management," *Bell Labs Tech. J.*, Vol. 5, No. 4, Oct.–Dec. 2000, pp. 44–62.

(Manuscript approved March 2001)

KURT S. SILVERMAN is President, Software Strategy, and Vice President, Software Technology, for Bell Labs. He holds a B.S. in economics from the Massachusetts Institute of Technology in Cambridge and an S.M. in management information systems and international management from the MIT Sloan School of Management.

MICHAEL R. BRENNER is a technical manager in the Network Solutions Architecture Department at Lucent Technologies in Holmdel, New Jersey. He leads a group of architects who define service and network management and application architecture in support of Lucent's network-hosted services solutions. He joined Bell Labs to work on applied expert systems and

OSSs after an extensive career in software design and development for real-time systems, medical imaging processing, and pioneering work in picture archiving and communication systems. Later, he developed and managed network management software for Lucent's ATM switches and for forward-looking technologies, such as mobile agents and policy management. Mr. Brenner holds B.S. and M.S. degrees in computer science from the Polytechnic Institute in Bucharest, Romania.

GREGORY E. SHANNON is a technical manager working in support of Lucent's corporate Architecture Council at Bell Labs in Columbus, Ohio. He received a B.S. degree from Iowa State University in Ames as well as M.S. and Ph.D. degrees from Purdue University in West Lafayette, Indiana, all in computer science. ◆