♦ A Framework for Understanding Network Intelligence

Madhav Moganti and Steven F. Knittel

The term network intelligence *is widely used in the field of communication networks. A number of new and potentially limiting concepts and products based on the concept of network intelligence have been introduced, including smart flows, intelligent routing, and intelligent web switching. Many intelligent systems focus on a specific service, function, or device, and do not provide true end-to-end network intelligence. True network intelligence requires more than a set of disconnected elements; it requires an interconnecting and functionally coupled architecture that enables the various functional levels to interact and communicate with each other. We propose a uniform framework for understanding end-to-end communication network intelligence (CNI), which is defined as the ability of a network to act appropriately in a changing environment. We consider an appropriate action to be one that increases the optimal and efficient use of network resources in delivering services, and we define success as the achievement of behavior subgoals that support the service provider's ultimate goals, which are defined external to the network system. The framework presented incorporates the functional elements of intelligence into computational modules and interconnects the modules into networks and hierarchies that have spatial, logical, and temporal properties. Based on the framework proposed, we describe an end-to-end multiservice network application spanning the network management layer, optical layer, switching/routing layer, services layer, and other layers.*

Introduction

The study of network intelligence is an extremely active area in the field of communications.[1] Thanks to the latest advances in data communications—especially in the services sector and in the communications software, photonics, and programmable technologies areas—service providers are spending millions of dollars a year on an increasingly intelligent communication infrastructure and applications. Research in the areas of learning automata, intelligent agents technologies, intelligent data-mining, knowledge discovery, data-driven task sequencing, intelligent databases, wire-speed real-time databases, virtual modeling, and sophisticated communication network modeling has provided insights into intelligent computing processes. Significant progress has been made in rule-based reasoning, planning, and problem solving. Game theory and operations research have developed methods for decision-making in the face of uncertainty. Research into speech recognition, digital signal processor (DSP), and live audio and video processing has produced advances in real-time input processing and world modeling.

Progress has been rapid, and there is a large and rapidly growing body of literature in each of the above-mentioned fields as it relates to communication networks.[2-8] However, a general theoretical model of communication network intelligence (CNI) that ties all these separate areas of knowledge into a unified framework has been lacking. This paper is an attempt to formulate the broad outlines of such a model to better understand and to build flexible end-to-end intelligent systems. The goal is to create a general structure of CNI that encompasses converged next-generation communication networks. The model presented here incorporates knowledge of communication systems and processes gained from a great variety of sources.

Service providers are looking for vendors who can provide flexible end-to-end networks so they can benefit from reduced operations costs, which translate into more competitive, cost-effective service offerings. Unlike other vendors, Lucent Technologies, with its broad product portfolio, is in a good position to deliver end-to-end intelligent network systems.

For the most part, current communications industry practice assumes that networking consists of largely predictable processes that can safely proceed without the benefit of—or need for—in-process measurement and real-time feedback. Most adjustments in networking processes are made by service provider operators, who often use intuition and experience to tune parameters. As technological changes occur at an ever faster pace, insufficient consideration is being given to the need for real-time planning and replanning, automatic service adaptability, real-time resource optimization, and the ability to adapt to changing conditions. Today, real-time schedule changes, provisioning, configuration, and process modifications are being handled primarily by manual, ad-hoc methods.

Future-generation networking will be characterized by the need to adapt to the demands of agile networking, which include rapid response to changing customer requirements, automated design and engineering, lower-cost services, transparent distributed networking, resource allocation on demand, real-time planning and scheduling, increased quality, reduced tolerance for error, and in-process measurement and feedback. These demands generate requirements for adaptability and automated decision-making that cannot be met using current manual methods. Future networking systems will require automated intelligent networking features that apply intelligence to the domain of networking in such a way as to make possible the realization of a full range of agile and adaptable networks.

Panel 1. Abbreviations, Acronyms, and Terms

ATM—asynchronous transfer mode
BG—behavior generation
CNI—communication network intelligence
CPE—customer premises equipment
CWM—communications world modeling
DSP—digital signal processor
DWDM—dense wavelength division multiplexing
IP—Internet protocol
IRP—input and response processing
QoS—quality of service
LDAP—lightweight directory access protocol
SLA—service-level agreement
VA—value assessment
VLAN—virtual local area network
VOP—virtual optical path
VPN—virtual private network
VR—virtual router
WebDNS—Web domain name server

A definition and a reference application model of network intelligence are introduced in the following section. Subsequent sections deal with the elements of network intelligence, system architecture, temporal and spatial relationships in intelligent networks, and planning in intelligent networks. The next-to-last section describes an end-to-end intelligent network application; the final section presents our conclusions.

Definition of Network Intelligence

Intelligent communication networks must at least be able to understand the communication environment, to make decisions, and to use and manage network resources efficiently. More sophisticated levels of network intelligence include the ability to recognize user, application, service provider, and infrastructure needs, as well as expected and unexpected events, the ability to represent knowledge in a world model, and the ability to reason about and plan for the future.

For the purposes of this paper, *CNI* is defined as the ability of a network system to act appropriately in a changing environment. An appropriate action is one that increases the optimal and efficient use of network resources in delivering high-quality services; success is the achievement of behavioral subgoals that support the service provider's overall goals. Both the criteria for success and the service provider's overall goals are defined external to the intelligent network system. Typically, they are defined by the service provider's business objectives and are implemented by network designers, programmers, and operators. CNI is the integration of knowledge and feedback into an input- and output-based, interactive, goal-directed, networked system that can plan and generate effective, purposeful action directed toward achieving goals.

The degree or level of network intelligence is determined by:

- The computational power of the network and the network elements,
- The sophistication of the algorithms the network system uses, and
- The information and data values the system can store and access.

Network intelligence will evolve through growth in computational power and through the accumulation of knowledge about the types of input data needed for making decisions concerning expected response, and about the algorithmic processing required in a complex and changing communications environment.[8] Increasingly sophisticated network intelligence makes possible look-ahead planning, management before responding, and reasoning about the probable results of alternative actions. These intelligent network capabilities can provide service providers with competitive and operational advantages over traditional networks.

End-to-End Network Intelligence Reference Model

Figure 1 depicts an example end-to-end classification of different flows within an intelligent network. A flow, in this context, refers to the utilization of intelligent content. (It is beyond the scope of this paper to discuss in detail the classification of the flows.) These layered flows are used as a foundation on which the framework of network intelligence is conceived. The figure depicts a functional intelligent information flow in an end-to-end interconnected and layered intelligent communications network. Several intelligent network architectures have been proposed in the literature, among them Internet protocol (IP)-centric optical networks,[9] intelligent service management and delivery,[5,10] and intelligent IP tunneling.[11] This paper discusses end-to-end network intelligence flow issues.

The network intelligence layers proposed in the reference model shown in Figure 1 are end-user layer, content layer, application layer, subscriber layer, service provider layer, programmable technology and control layer, infrastructure provider layer, and network management layer. The following sections briefly explain and justify each of these layers, which are depicted in Figure 1.

End-User Layer Intelligence

The end-user intelligence layer provides the capabilities needed at the user's premises, which are not normally considered part of the service providers' networks. Because of improvements in the access bandwidth available to the end-user, the importance of this layer is continuing to grow. New developments in the Internet, home networking, and wired and wireless technologies are fueling the growth of the end-user intelligence layer. The increased bandwidth permits expanded intelligence within the equipment deployed on the customer premises and requires additional functionality[12,13] and coordination within the service provider space. These changes could make it possible to provide content to the user premises in anticipation of user needs and at times when there is less use of the service provider's network. Furthermore, intelligence at the end-user layer will be important in supporting new services tailored to the usage patterns and interests of users.

Content Layer Intelligence

A content-based intelligent network[14,15] allocates network bandwidth on the basis of content and user requirements and safeguards content in accordance with defined access policies. Such a network offers the following services:

- Content location (for example, using WebDNS),
- Content distribution and replication,

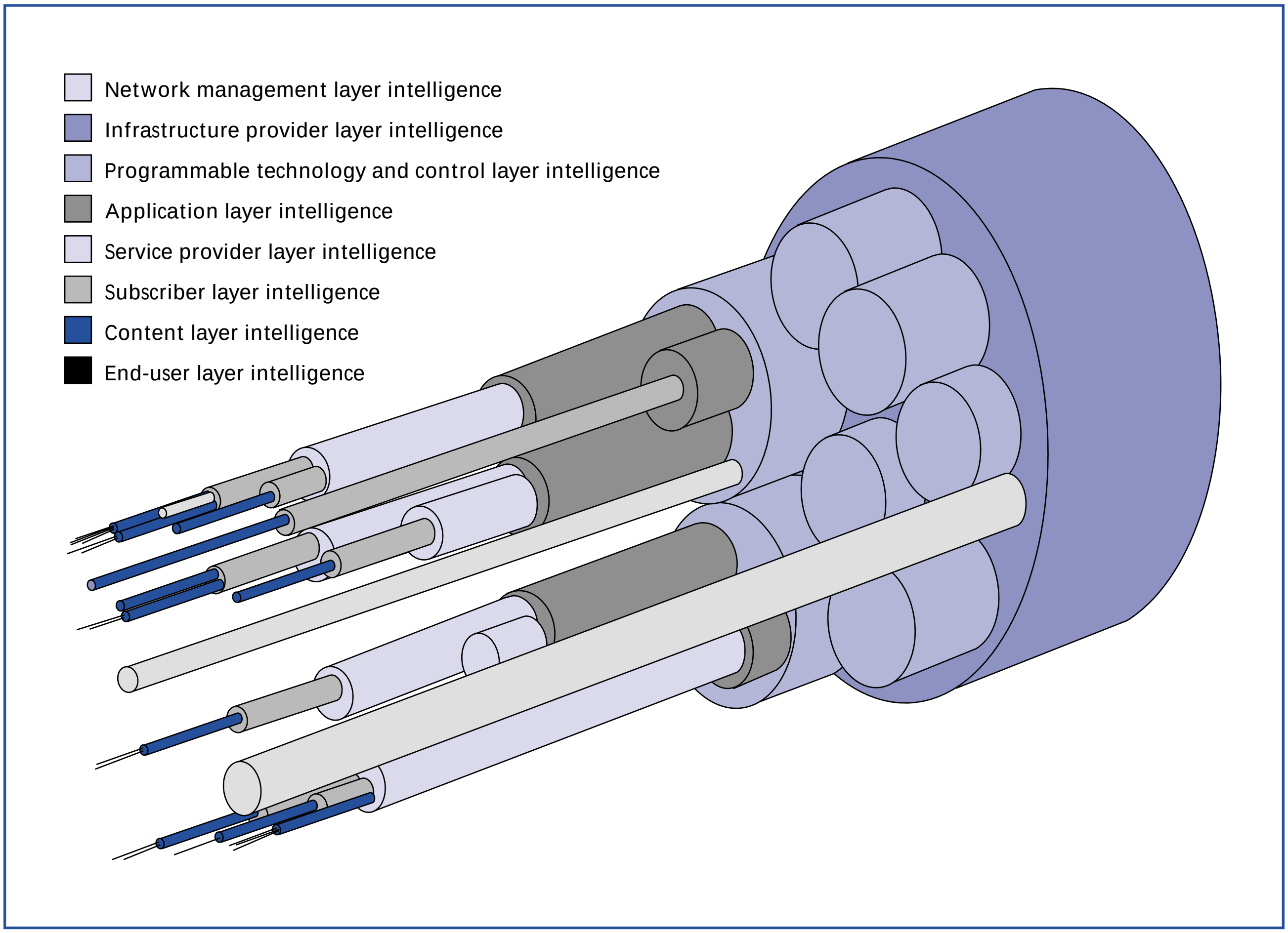


Figure 1.
Example depicting communication intelligence information flow layers that could be managed with better understanding of inter- and intra-layer network intelligence.

- Content caching, and
- Content redirection and forwarding.

The emergence of the Internet has radically changed the way individuals and corporations use networks, where information is located, and how information is accessed. Currently, if a large number of users want to access a "hot" content area at the same time, "flash" network overloads can occur that stress the infrastructure beyond its limits. As a result, service provider networks are changing the information delivery mechanism from passive content retrieval to proactive content delivery based on network policies and user identity. The passive retrieval model requires a network infrastructure that is built for predictable network and server loads. The proactive delivery model requires that content be intelligently distributed closer to clients and network access points, the better to cope with sporadic network loads driven by hot content.[16,17] In the content-based intelligent paradigm, an end user's customer premises equipment (CPE) device deals with content in the network directly. A content-aware CPE device requires a content-based intelligent network environment that facilitates the distribution of content to locations where it is requested. This minimizes unnecessary network loads that result from focused overloads or backbone constraints.

Application Layer Intelligence

The number of new network applications and services to be supported continues to increase rapidly. The traffic generated by these applications creates different traffic load and flow patterns, depending upon whether or not the applications[18,19] are:

- Real-time,
- Computation intensive,
- Network topology dependent,
- End user dependent,
- High bandwidth, and
- Delay sensitive.

In order to properly design, evaluate, and deploy efficient network equipment for an application environment,[20] a service provider must have a good understanding of the source models[21] of the network application traffic. In particular, one would like to find characteristics of how an application host generates network traffic that are invariant over time. There are a number of reasons—based on application architectures, design, and human factors—why application traffic may vary significantly. They include:

- User access type,
- Application communication methods,
- Single transaction vs. multiple linked transaction applications, and
- End user input and interaction strategy.

Intelligent service mediation[22] techniques are used to control access to network services and to customize behavior for the service provider, application provider, and end user. Application layer intelligence will allow application service providers to manage application resources more effectively—maximizing utilization and return on investment—by dynamically matching appropriate resources with application demands.

Subscriber Layer Intelligence

The subscriber-based intelligent network environment consists of a group of CPE devices communicating and sharing one or more resources in a decentralized way. This type of networking demands certain interesting relationships between the service provider's network elements and the CPE devices. Good examples of subscriber-based intelligence are peer-to-peer network applications, cluster-computing,[23] networked parallel processing, and mapping of logical storage area networks on physical or virtual network topologies.[24] Some of these applications demand particular logical network topologies to enable the applications. New applications and new business models are pushing service providers to support these environments. The advantages of using subscriber-based intelligence are that it speeds up algorithm execution, minimizes internode communication delays, improves resource utilization, and provides fault tolerance by restoring network connectivity (transparently to the user) when faults occur. Furthermore, it allows features and services to be customized for pre-designated user groups or for an individual user. In this environment, a user would be able to select preferred network resource characteristics (such as virtual topologies), to activate personalized features (from "skins" through connection speeds), and to provide information to the system to improve its performance.[25]

Service Provider Layer Intelligence

Service provider layer intelligence makes it possible to carry end user traffic by applying service provider constraints to end user needs. Intelligent tunneling, virtual network switching/routing in virtual private networks (VPNs), load balancing networks, and virtual local area networks (VLANs) are examples of service provider intelligence. This layer's features include quality of service (QoS), isolation, load distribution, and policing capabilities that allow service providers to deliver flexible, measurable, and enforceable service-level agreements (SLAs) to other service providers and to subscribers and to deliver real-time and non-real-time services from multiple sources.[11] This enables a service provider to provide large service providers with dedicated virtual resources and to allow small service providers to share virtual resources that are managed administratively by the service provider.

Programmable Technology and Control Layer Intelligence

This layer provides interoperability and adaptability across heterogeneous networks that support a wide range of signaling protocols. The programmable switches translate industry signaling protocols into a generic signaling format, simplifying the addition of new protocols. This capability allows legacy service providers and new service providers to provide rich, seamless interoperability between their network domains and enables signaling interworking between multiple vendor gateways. The Lucent Softswitch[8] is an example of an enabling programmable switching technology in this layer. Adaptable physical layer and active network concepts that enable applications to react better to changing conditions are being developed.[13,26-28] These concepts enable applications to optimize physical layer performance proactively, using an application-defined set of metrics.

Infrastructure Provider Layer Intelligence

Broadband technology promises a convergence of technologies and service platforms. A variety of factors are forcing today's leading service providers to change how they do business and, hence, the way they build and evolve their next-generation networks. Increasingly, service providers are being pressured to build networks capable of supporting a variety of old and new infrastructures and to provide new value-added services at the lowest possible initial and incremental price. The problems inherent in simultaneously supporting an existing network and deploying a new multiservice infrastructure point to a solution that avails itself of the benefits of the frame relay, asynchronous transfer mode (ATM), IP, and dense wavelength division multiplexing (DWDM) technologies.[29] Infrastructure layer intelligence provides the capabilities to deal with these complexities. Technologies such as all-optical DWDM and multiservice platforms are making possible a convergence of technologies and service platforms; in this layer, network solutions can operate in dynamic, reconfigurable, multivendor, multitechnology, and multiprotocol environments.

Network Management Layer Intelligence

Network management layer intelligence deploys, integrates, and coordinates all the resources necessary to configure, monitor, test, analyze, evaluate, and control the communication network to meet service-level objectives. The goals of network management are efficient use of resources, control of strategic assets, minimization of downtime, management of constantly changing communications technology and services, and reduction of the cost of network operations. Intelligence within the network management layer must integrate diverse services, networks, technologies, and equipment.

The network structure in **Figure 2** captures the communications intelligence information flow depicted in Figure 1. In Figure 2, the layers with clouds represent virtual entities or soft devices. Even though Figure 2 depicts network management as a separate layer, some network management functions are actually distributed across the other layers embedded in element management systems. For simplicity and convenience, this embedding is not shown.

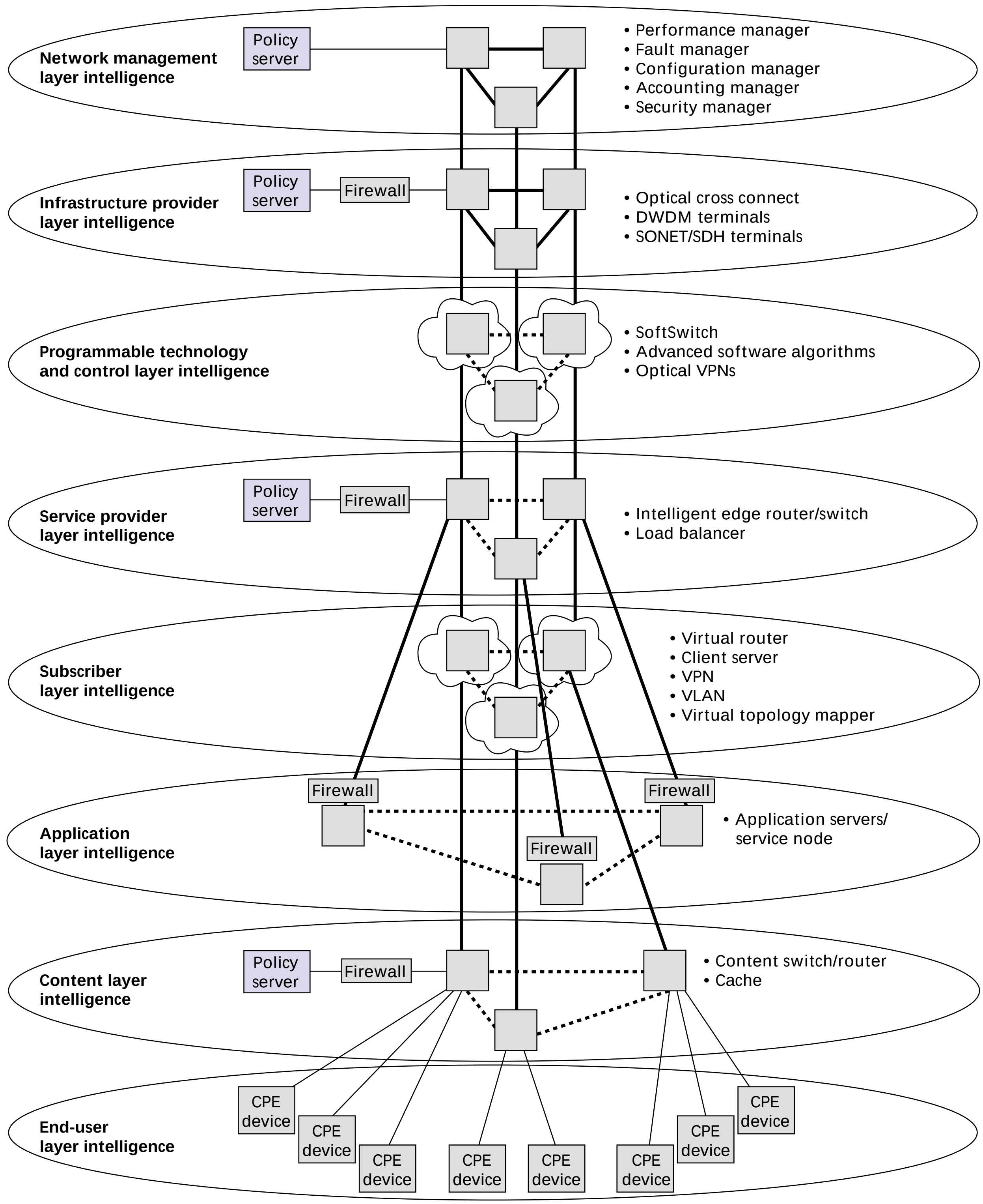
It is a well-known fact that many vendors and the research community have independently studied the intelligence required in the layers shown in Figure 2.[5,9,10] As a result, most areas identified in Figures 1 and 2 have been throughly investigated, and there is an abundance of information on each of them. The following products can be mapped to the reference model shown in Figure 1: content-based switching and routing gear by Lucent (IMMINET™) Alteon and ArrowPoint, and Lucent's intelligent virtual routers—Spring Tide® equipment, all-optical transport networks, and WaveStar™ LambdaRouters with DWDM transport network gear. However, important structural questions remain, including:

- What is the dependence of one layer on another?
- What are the interrelationships among the layers?
- How does a change in one layer affect and proliferate to other layers?
- How do changes in the network environment affect the overall network and, hence, each of the layers?
- How does the addition of new technologies, new applications, and new services affect the next-generation network paradigm?

These are the current challenges the industry is working to address. Lucent Technologies is an industry leader uniquely positioned to address these issues as part of its mission to satisfy the end-to-end infrastructure needs of service providers. The framework of network intelligence presented here provides a mechanism to capture and address these structural issues.

The Elements of Network Intelligence

In an end-to-end communications network, each individual level of network layers and elements is reasonably well understood by researchers and vendors. But overall network intelligence requires more than a set of disconnected elements; it requires an intercon-



CPE – Customer premises equipment
DWDM – Dense wavelength division multiplexing
SDH – Synchronous digital hierarchy
SONET – Synchronous optical network
VLAN – Virtual local area network
VPN – Virtual private network

Figure 2.
Mapping of layers of network intelligence on end-to-end network infrastructure.

necting and functionally tightly coupled system architecture that enables the various functional levels to interact and communicate with each other.

There are four system elements of network intelligence within the proposed framework:

- Input and response processing (IRP),
- Communications world modeling (CWM),
- Behavior generation (BG), and
- Value assessment (VA).

Input and Response Processing

Input to and response (that is, output) from intelligent network systems are driven by end-user interactions/needs and the results of network element activities.

Input. Input to an intelligent network system is produced by interactions between the network and the network environment. For example, input to an intelligent network system can be produced by interactions with an end user, which may include end-user requests specifying type of information sought, quality of information sought, ability to use higher bandwidths at higher prices, types of services requested, duration of the service, and the type of user (for example, business, residential, or real-time). The system can use inputs to monitor both the state of the external world and the internal state of the network. The inputs collected are provided to an input processing system.

Input processing. An input processing system element compares observations of input with expectations generated by an internal communications world model. Input processing algorithms integrate similarities and differences between observations and expectations over time and space in order to detect events and recognize features, patterns, and relationships in the external world. The input data from a wide variety of sources over extended periods of time are combined into a consistent, unified perception of the state of the communications world. Input processing algorithms also compute several network system characteristics, including both physical and logical dynamic attributes of network elements and events of interest.

An example of the work done by the input processing system is the network address translation of IP addresses, which makes use of both the content of the end user's input and information from previous interactions with the network.

Response Response in an intelligent network system is produced by the response processing system, which makes it possible to communicate effectively with and to interact with the network environment. For example, response from a circuit-packet gateway switch could be the translation of a circuit signaling protocol to a packet signaling protocol to enable communication with the packet network.

Response processing. Response processing in an intelligent network system is the result of the execution of behavior generation algorithms upon the world model.

An example of this is the all-optical WaveStar™ LambdaRouter micromechanical mirror control plane response. In this case, the output response from an intelligent network system is produced by micro-mirror actuators that move microscopic mirrors to cross connect wavelengths dynamically. An intelligent network system node may have hundreds of these actuators, all of which must be coordinated in order to perform end-to-end tasks and satisfy a service provider's dynamic routing needs, which in turn satisfy users' needs for bandwidth.

Communications World Modeling

The communications world model is the intelligent network system's best estimate of the state of the network and its environment. It includes a memory (for example, a distributed main memory database, or lightweight directory access protocol [LDAP] servers) that contains information about the network and its environment, and a memory management system that stores and retrieves that information. The communications world model is used to generate expectations and predictions about such things as network resources, operations, and usage, and can respond to requests for information about the present, past, and probable future states of the network. The communications world model also provides information services to:

- The *behavior generation* system element (discussed in the following sections), to enable it to generate communication environment planning and network behavioral choices;
- The *input processing* system element, to enable

it to perform correlation and matching and to recognize states, patterns, and events in the network; and
- The *value assessment* system element (discussed in the following sections), to enable it to compute cost, benefit, risk, uncertainty, importance, and attractiveness of choices.

The communications world model is kept current by the input processing system.

The demographic database of a country, customer needs, market needs, service profiles, logical network topologies, and customer service-level agreements are all part of the communications world model.

The communications world model provides the intelligent network system with the information necessary to reason about network services, network needs, network resources, and time. The communications world model contains information on items that are not directly and immediately observable, and it enables the system to integrate input from many different sources into a single reliable representation of the network intelligence domain. The communications world model may be represented in intelligent network systems by data such as traffic matrices, traffic estimates, service profiles, and policy agreements, and may be maintained in database structures.

CWM modules. In each node of the network hierarchy, the CWM modules:

- Maintain a knowledge database by keeping it current and consistent. In this role, the CWM modules perform the functions of a database management system. The CWM state estimates are updated on the basis of correlations and differences between communications world model predictions and observations of input data at each intelligent network node. The CWM modules save newly generated or recognized entities, states, and events in a knowledge database and delete entities, states, and events that the input processing modules determine no longer exist in the communication environment.
- Generate predictions of expected input values for use by the appropriate input processing modules. In this role, a CWM module performs the functions of a state predictor, generating predictions that enable the input processing system to perform correlation and predictive filtering. CWM predictions are based on both the state of the network/service task and the estimated states of the external parameters.
- Answer "What is?" queries from the automated planners and executors in the BG modules. CWM estimates of the current state of the network and its environment are also used by the planners in the BG module as a starting point for planning.
- Answer "What if?" queries from the planners in the BG modules at the same level. In this role, the CWM modules perform a simulation function by generating the status to be expected as a result of the actions hypothesized by the BG modules. The results predicted by CWM simulations are then sent to VA modules for evaluation. For each action hypothesized by the BG modules, a CWM prediction is generated and a VA evaluation is returned to the BG modules. This BG-CWM-VA linkage allows the BG modules to select (as the plan to be executed) the sequence of hypothesized actions producing the best results.

Data structures for representing explicit knowledge reside in a knowledge database that is hierarchically structured and that is distributed in such a way that there is a knowledge database for each CWM module in each node at every level of the intelligent network system hierarchy. The communication system provides services that make the CWM modules and the knowledge databases behave like a global virtual common memory in response to queries and updates from the BG, IRP, and VA modules. The communication interfaces with the CWM modules in each node provide a window into the knowledge database for each of the computing modules in that node.

Knowledge representation in the communications world model. The knowledge database of the communications world model contains both a priori information, which is available to the intelligent network system before action begins, and a posteriori knowledge, which the intelligent network system obtains by monitoring the environment as the network functions. The database contains information about space, time, entities, events, and the states of the network elements and the network environment. For example, a priori information could include the knowledge that an optical transport node has received data in the range (100Mbps—minimum, 400Mbps—most likely, 600Mbps—maximum) every Monday between 1 p.m. and 2 p.m. for the past year. The knowledge database also includes information about the intelligent system itself, such as the values assigned to goals, objects, and events; the parameters embedded in dynamic models of the virtual routes and optical paths; and the states of all the processes currently executing in each of the BG, IRP, CWM, and VA modules.

Knowledge about traffic engineering rules, network element constraints, capacities, and the rules of logic and statistics is represented as parameters in the CWM functions that generate predictions and simulate the results of hypothetical actions. Physical knowledge may be represented as algorithms or formulas, or as IF/THEN rules[30-32] describing what happens in certain situations, such as when a network node fails, a link is cut, or a new service request appears. The correctness and consistency of communications world model knowledge is verified by input processing mechanisms that measure differences between communications world model predictions and collected trace observations.

Entities in the communications world model. The communications world model contains information about stored network entities. The knowledge database contains a list of all the entities that the intelligent network system knows about. A subset of this list is the set of current entities known to be present in any given situation. A subset of the list of current entities is the set of entities-of-attention based on locality of reference properties.

There are two types of entities: generic and specific. A *generic entity* is an example of a class of entities. A generic entity structure contains the attributes of its class. A *specific entity* is a particular instance of an entity. A specific entity structure inherits the attributes of the class to which it belongs. An example of an entity structure is shown in **Table I**.

Table I. An example entity structure.

Entity name	Name of entity
Kind	Class
Type	Generic or specific
Area	Access, transport, routing, switching
Position	World/virtual map coordinates
Dynamics	Mobile, fixed
Path	Sequence of positions/routes
Geometry	Size, shape
Links	Sub-entities, parent entity
Properties	Physical, logical, topology
Behavioral	Protocols, standards, semantics
Performance	Delay, loss, load characteristics
Reliability	Availability, fault-tolerance
Capabilities	Bandwidth, range, configuration types, capacity
Interfaces	Communication, and control interfaces
Value state-variables	Success-failure, thresholds, COS parameters
Management	Provisioning, administration, and configuration
Security	Access control lists, filters

COS – Class of service

Entity relationship in the network communications world model. Entity representation is cross referenced and tightly coupled by real-time computing hardware. Many of the attributes in an entity structure are time-dependent state variables. Each time-dependent state variable may possess a short-term memory queue that describes its temporal history. At each network node, temporal traces stretch backward at least as far as the planning horizon at that level stretches into the future. At each network hierarchical level, the historical trace of an entity state variable may be captured by summarizing data values at several points in time within the historical interval. Each state variable in an entity structure may have value state variable parameters that indicate levels of confidence and measures of network environmental uncertainty. These are computed by value-assessment functions that reside in the VA modules.

The network entity database hierarchy. The entity database is hierarchically structured. Each entity consists of a set of subentities and is part of a parent entity. For example, a network resource (that is, hardware or software) may consist of a set of network components (hardware or software) and be part of a larger network resource. An intelligent network node is task- or goal-driven; the structure of the communications world model entity database is defined by the nature of these tasks and goals.

Communication events. A *communication event* in an intelligent network node is a state, condition, or situation that exists in a network at a point in time or occurs over an interval of time. Events are represented in the communications world model with attributes in time and space. Event attributes may indicate such things as start and end time, duration, type, and relationship to other events. An example of an event structure is shown in Table II.

State variables in the event structure may have confidence levels, degrees of support, and measures of environmental uncertainty similar to those in the entity structures. Confidence state variables may indicate the degree of certainty that an event actually occurred or that it was correctly recognized.

Behavior Generation

Behavior results from a behavior-generating system that selects goals and plans and executes tasks. Tasks are recursively decomposed into subtasks, and subtasks are sequenced so as to achieve goals. Goals are selected and plans generated by cyclical interaction among BG, CWM, and VA modules. The behavior-generating system hypothesizes plans, the CWM module predicts the results of those plans, and the value assessment element evaluates those results. The behavior-generating system then selects for execution the plans with the highest evaluations. The behavior-generating system element also monitors the execution of plans and modifies existing plans whenever the situation requires. For example, events such as congestion, network node overload, and major changes in traffic patterns should be detected quickly and appropriate corrective actions should be taken to deal with them.

Behavior in a network or network node is the result of executing a series of communications tasks.

Table II. An example event structure.

Event	Name of event
Kind	Class
Type	Generic or specific
Modality	Voice, video, data, etc
State	Simple, composite, pseudo, final
Time	When event detected
Interval	Period over which event took place
Position	Map location where event occurred
Links	Sub-event, parent event
Guard	Boolean expression attached to a transition
Transition	Relationship between a start and final state
Alarms	Visual, message
Value	Benefit-cost, risk

A *communication task* is a piece of work to be done or an activity to be performed by a network node. Each intelligent network system is capable of performing a set of tasks. Each task in this set is assigned a name. The communication task vocabulary is the set of task names assigned to the set of tasks the system is capable of performing. This set of tasks can be expanded through learning, training, or programming. Typically, a communication task is performed by one or more intelligent communication agents on one or more entities. The performance of a task can usually be described as an activity that begins with a start event and is directed toward a goal event. A *goal* is an event that successfully terminates a communication task; it is the objective toward which task activity is directed. A communication task command is an instruction to perform a named task; it may have the following form:

PERFORM <*TaskName(parameters)*>

AFTER <*Start Event*> UNTIL <*Goal Event*>

Communication task knowledge is knowledge of how to perform a task, including information about the algorithms, protocols, parameters, time, events, resources, information, and conditions required and the costs, benefits, and risks to be expected. In a network node, task knowledge may be expressed implicitly in algorithms, software, or hardware, or explicitly in data structures or a network node database.

A communication task structure is represented as a data structure in which task knowledge is stored. In systems in which task knowledge is explicit, a task structure[33] can be defined for each task in the task vocabulary. An example of a task structure is shown in Table III.

The task knowledge represented in task structures has a variety of uses. For example, network planners and operators may use it to generate hypothesized actions. The communications world model may use it to predict the results of hypothesized actions. The value assessment system may use it to determine how important the goal is and how many resources to expend in pursuing it.

Behavior generation is inherently a hierarchical process. At each level of the behavior generation hierarchy, communication tasks are decomposed into subtasks that become task commands for the next lower level. At each level of a behavior generation hierarchy there is a task vocabulary and a corresponding set of communication task structures.

In the CNI architecture, each level of the hierarchy contains one or more BG modules. At each level, there is a BG module for each function in each network layer. The function of the BG modules is to decompose task commands into subtask commands. Input to BG modules consists of commands and priorities from BG modules at the next higher level, evaluations from nearby VA modules, and information about past, present, and predicted future states of the network environment from nearby CWM modules. Output from BG modules consists of subtask commands to BG modules at the next lower level, status reports, and "What is?" and "What if" queries to the CWM modules about the current and future states of the network environment.

Value Assessment

The value assessment system element is used to determine the importance, risk, and probability associated with the events and actions involved in the intelligent network. The value assessment system evaluates both the observed state of the communications world and the predicted results of hypothesized plans. It can compute costs, risks, and benefits of both observed situations and planned activities.[34] It computes the probability of the correctness of state variables and assigns

Table III. An example task structure.

Task name	Name of the task
Type	Generic or specific
Actor	Agent performing the task
Action	Activity to be performed
Object	Thing to be performed
Subject	Thing to be acted upon
Goal	Event that successfully terminates or renders the task successfully
Parameters	• Priority • Status (for example, active, halted, waiting, inactive) • Timing requirements • Source of task command
Requirements	• For example, tools, time, resources, and events needed to perform the task • Enabling conditions that must be satisfied to begin, or continue, the task • Information that may be required
Procedures	• A plan for executing the task • Functions that may be called • Algorithms that may be needed
Effects	• Expected results of task execution • Expected costs, risks, benefits • Estimated time to complete

believability[35] and uncertainty[36] parameters to them. The value assessment system provides the basis for making decisions, for choosing one response rather than another. Without value assessment, any intelligence in a network would soon become obsolete.

The challenge for today's service providers is to provision and meet QoS-based SLAs. When SLAs cannot be met, traffic congestion controls should always attempt to minimize penalties and maximize revenue when deciding which traffic to admit.[37] If the monitoring process indicates that a customer's contracted SLA is not satisfied, the service provider is deemed non-compliant to the SLA and thus liable to the customer. Hence, every lost flow contributes to penalty in the VA modules.

Architecture for Understanding Network Intelligence

A number of system architectures for intelligent network systems have been conceived at various network layers, and a few have been implemented.[3,5,8,11,27,38,39]

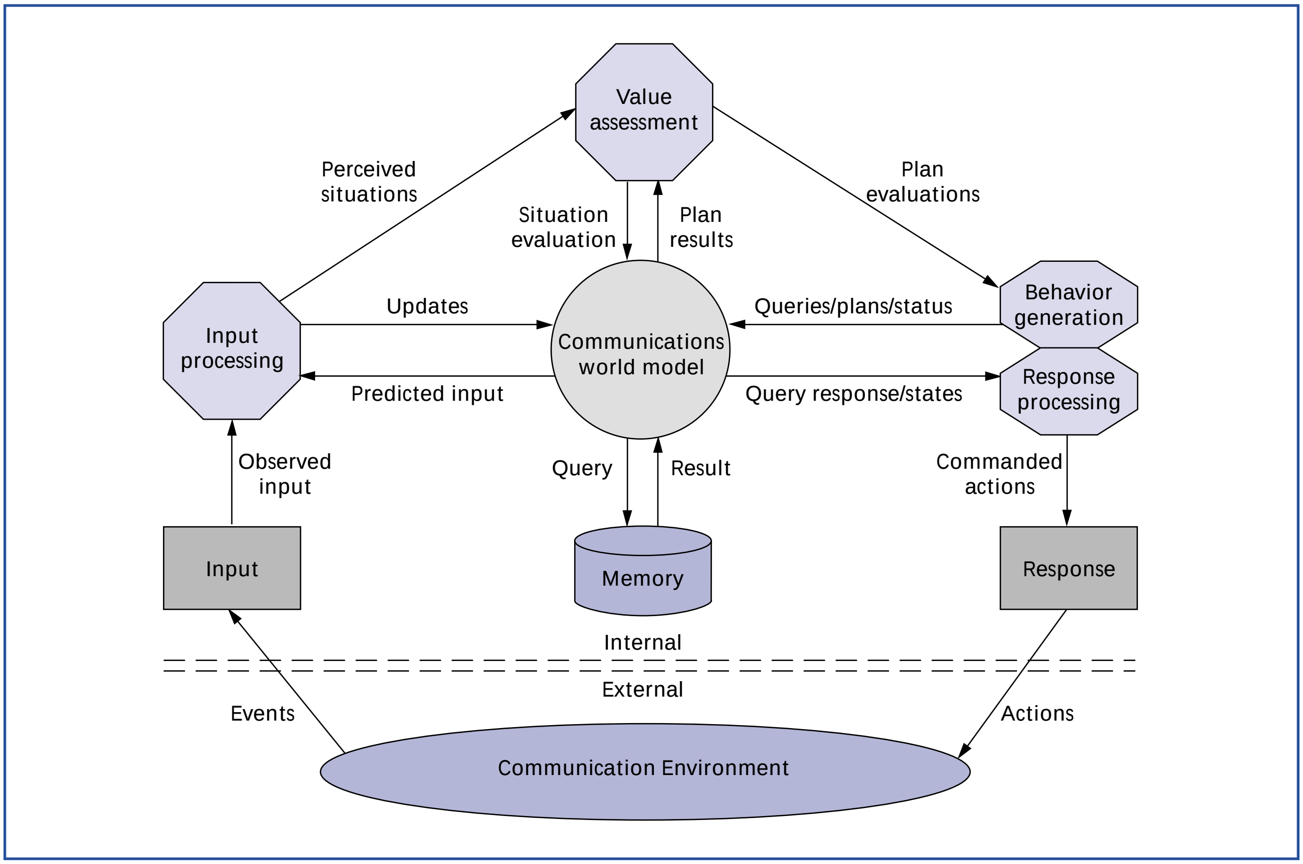


Figure 3.
Elements of network intelligence and the functional relationships.

The proposed framework and architecture organize the elements of network intelligence described in the previous section to create the functional relationships and information flow shown in **Figure 3**. The proposed functional architecture replicates and distributes the relationships shown in Figure 3 over a hierarchical structure with the logical and temporal properties illustrated in **Figure 4**. Each node in the organizational hierarchy contains four types of computing modules: BG, CWM, IRP, and VA. Each chain of command in the organizational hierarchy—from each response and each input to the highest level of control—can be represented by a computational network hierarchy, such as that shown in the center of Figure 4.

In Figure 4, the left side of the figure is an organizational hierarchy consisting of a tree of command centers, each of which possesses one supervisor and one or more subordinates. In the center of the figure is a computational hierarchy consisting of BG, CWM, IRP, and VA modules. A computational hierarchy services each response and each input. On the right side of the figure is a behavioral hierarchy consisting of event progressions through state-time-space. At each functional level, the nodes—and the computing modules within the nodes—are tightly interconnected. As shown in Figure 3, within each computational node, the communications system provides internetwork functional layer communications as follows:

- Queries and task status are communicated from BG to CWM modules;
- Responses, including retrieval of information, are communicated from CWM modules back to the BG modules making the queries;
- Predicted input data is communicated from CWM to IRP modules;
- Updates to the communications world model

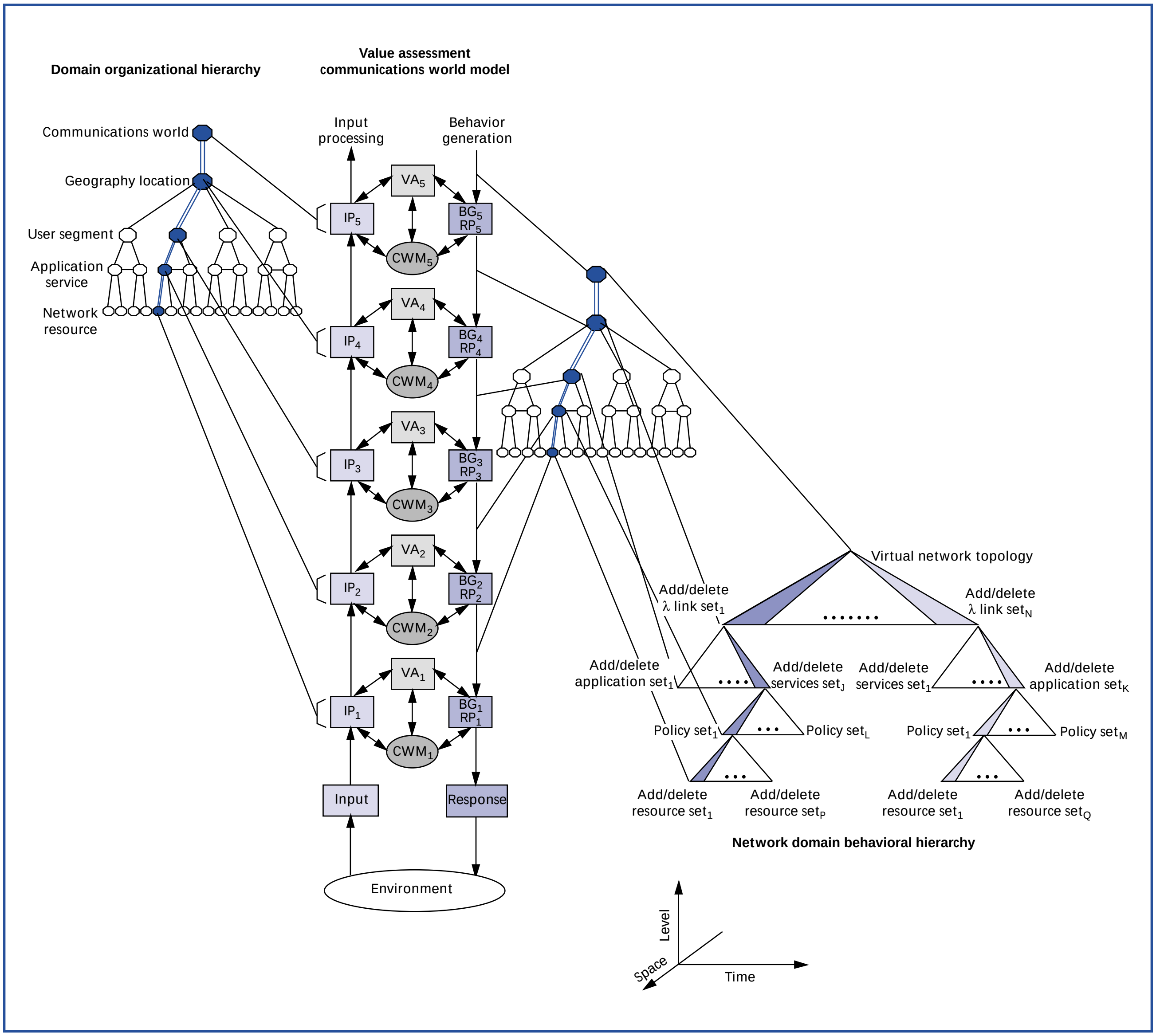


Figure 4.
Relationships in hierarchical network structure.

are communicated from input processing to CWM modules;

- Observed entities, events, and situations are communicated from input processing to VA modules;
- Values assigned to the communications world model representations of these entities, events, and situations are communicated from VA to CWM modules;
- Hypothesized plans are communicated from BG to CWM modules;
- Results are communicated from CWM to VA modules; and
- Evaluations are communicated from VA modules back to the BG modules that hypothesized the plans.

The communications system also provides communications between functional layers at different levels:

- Instructions and commands are communicated downward from supervisor BG modules in one level to subordinate BG modules in the level below.

- Feedback and status reports are communicated upward through the CWM modules from lower-level subordinate BG modules to the upper-level supervisor BG modules from which they received commands.
- Observed entities, events, and situations detected by IRP modules at one level are communicated upward to IRP modules at the next higher level.
- Predicted attributes of entities, events, and situations stored in CWM modules at a higher level are communicated downward to CWM modules at a lower level.
- Input to the bottom-level input processing modules comes from input information collected from different sources.
- Output from the bottom-level response processing modules is communicated to the response subsystem.

The intelligence within the network system can be realized in a variety of ways. One way is to embed intelligence functions in the network management system and the network node elements. Communication between the management system and the node elements can be achieved by using a management communication network.[40] In the system architecture proposed here, the input/output relationships of the communications system produce the effect of an intelligent network, the functionality of which may be equated to that of a blackboard system.[41] The input command string to each of the BG modules at each level generates a response through state-space as a function of time. The set of all command strings creates a behavioral hierarchy, as shown on the right side of Figure 4. The responses (not shown in Figure 4) correspond to observable output behavior.

The next two sections discuss two important aspects of intelligent network systems: temporal and spatial relationships between processes, and planning, which is central to the operation of intelligent network systems. Other important topics related to intelligent network systems that are within the scope of the proposed framework are detailed data and knowledge representation for capturing network intelligence, communication language and its mapping to communication protocols, representation of the network environment, learning and network behavior pattern detection, and generation algorithms. These topics are beyond the scope of this paper.

Temporal and Spatial Relationships in Intelligent Networks

This section begins by discussing the general characteristics of temporal and spatial attributes and concludes with an example of their use.

Characteristics of Temporal and Spatial Attributes

Each level in the behavior generation hierarchy is defined by temporal and spatial decomposition of communication goals and tasks into subgoals and subtasks of varying complexity. Temporal granularity is expressed in terms of bandwidth, sampling rate, and state-change intervals, while temporal span is measured by the length of historical traces and planning horizons. Spatial granularity is manifested in the branching of the task tree, while a span is measured by the span of control and the range of service/application/user domains. Levels in the input processing hierarchy are defined by temporal and spatial integration of input data into levels of aggregation. Temporal aggregation is best illustrated by diurnal and seasonal parameters, such as busy hour and busy season. Spatial aggregation is best illustrated by environmental characteristics, such as demography and geography. Levels in the communications world model hierarchy are defined by the temporal granularity of events, by the spatial granularity of the service/application/user domain, and by the parent-child relationships between network entities (for example, service nodes serving access nodes serving customer premises nodes).

Numerous theoretical and experimental studies in other fields support the concepts of hierarchical planning (discussed in the next section) and natural *clustering* for both temporal and spatial entities.[42,43] These studies also indicate that, in a hierarchically structured, goal-driven, input- and-response-based, interactive, intelligent communications system architecture:

- Communication bandwidth decreases about an order of magnitude at each higher level,
- Computational granularity of temporal and

spatial patterns decreases about an order of magnitude at each higher level,

- Goals expand in scope and planning horizons expand in space and time about an order of magnitude at each higher level, and
- Models of the world and storage requirements for events decrease in granularity and expand in temporal and spatial range by about an order of magnitude at each higher level.

Network traffic modeling and monitoring techniques[44,45] seem implicitly to assume the four assertions listed above. From these assertions, a timing diagram can be constructed, as shown in **Figure 5**. In the figure, the range of the time scale increases—and its resolution decreases—exponentially by about an order of magnitude at each higher level. Hence the planning horizon and event summary interval increase—and the communication bandwidth and frequency of subgoal events decrease—exponentially at each higher level. The seven hierarchical levels span a range of time intervals that vary from a few milliseconds to one year. (One year was arbitrarily chosen as the longest historical-memory/planning-horizon to be considered.) Even shorter intervals could be included by adding layers to the bottom of the figure, while longer intervals could be included by adding layers to the top, or by increasing the difference in communication bandwidths and input and response clustering intervals between levels.

The origin of the time axis in Figure 5 (that is, the point at which t = 0) is the present. Future plans lie to the right of t = 0, past history to the left of it. The empty (unfilled) triangles in the right half-plane represent task goals in a future plan. The blue triangles in the left half-plane represent recognized task-completion events in past history. At each level there is a planning horizon and a historical event summary interval. The dark shading on the right shows the planning horizon for the current task. The light shading on the right indicates the planning horizon for the next anticipated task. The dark shading on the left shows the event summary interval for the current task. The light shading on the left shows the event summary interval for the previous task.

The timing diagram in Figure 5 illustrates the temporal flow of activity in the task decomposition and input processing systems. At the communications world level, high-level input events and periodic user, server, application, market behaviors, and daily routines generate plans for the day, the year, and so on. Each element of the plan is decomposed through the remaining six levels of task decomposition into a set of specific actions. At each hierarchical level, planner modules decompose task commands into planned subtasks for execution. At each level, events are summarized, integrated, and clustered into single events at the next higher level. A high-level formalized event specification language can be used to capture events.[46,47]

An Example of the Use of Temporal and Spatial Attributes

The following example describes how temporal and spatial attributes are captured in the CWM modules and are used by the BG modules. The example presented in this section employs an intelligent all-optical DWDM network layer. An all-optical network uses optical cross connects to route wavelengths. Lucent's WaveStar LambdaRouter allows wavelengths to be assigned and provisioned on demand in a transport network, making it possible for a service provider to offer dynamic bandwidth delivery in seconds. Traditionally, transport networks are dimensioned using busy hour/busy season traffic and static traffic matrices. But dynamic bandwidth trading requires calculating routes and traffic flows dynamically in real time. This, in turn, requires creating and maintaining *dynamic traffic matrices*. Traditional transport network designs, because of their static nature, allow network planners and service provider operators to dimension and fine-tune designs using tools; but with dynamic traffic matrices, human intervention is not possible, because of the dynamic nature of the traffic and the amount of information to be handled. The example that follows presents an outline of the automatic generation of dynamic traffic matrices for an intelligent optical network layer to use.

Let us assume that a service provider is offering bandwidth delivery on demand, using an all-optical transport network on which traffic changes every few minutes, and that the logical topology of the network must be computed and reconfigured accordingly.

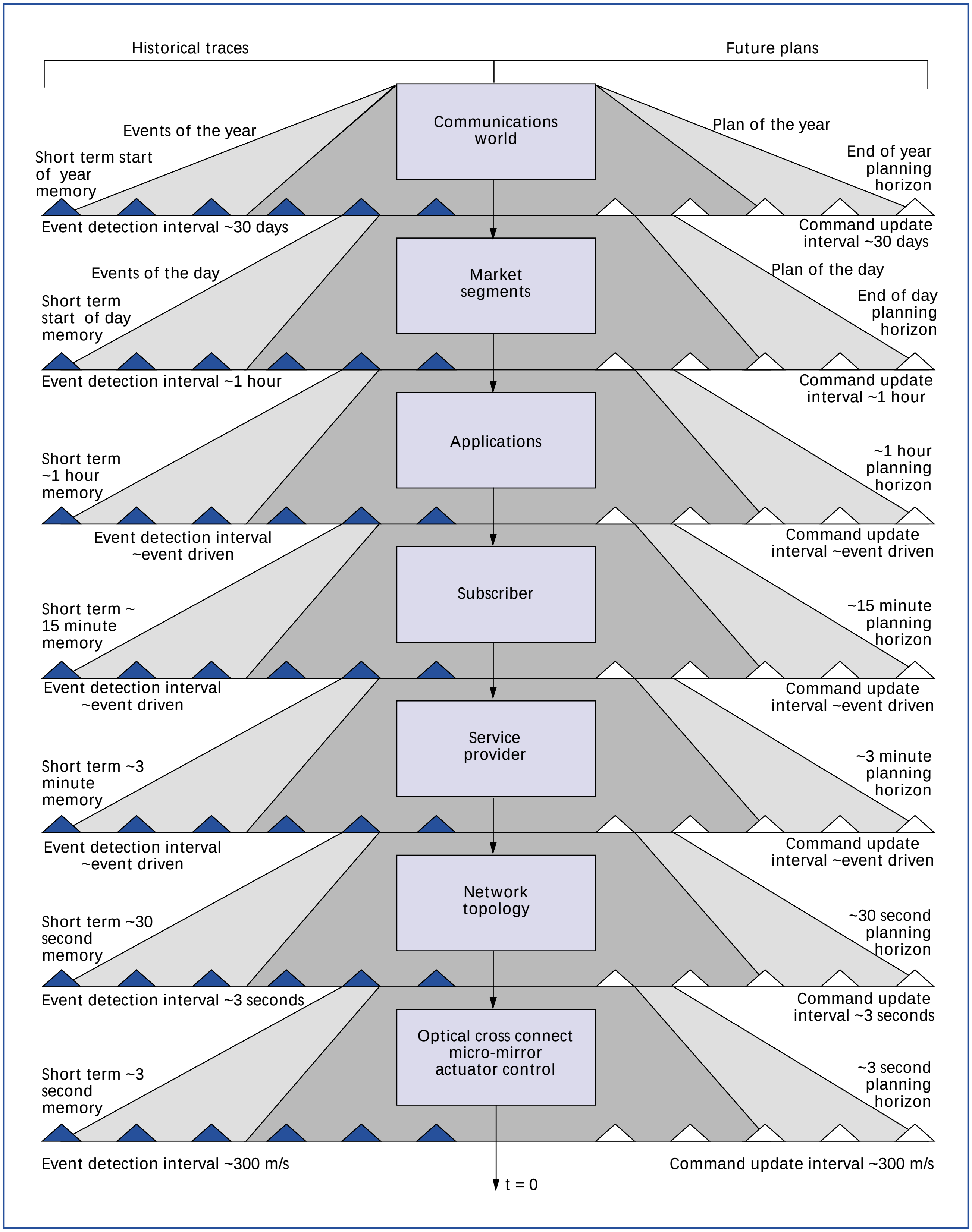


Figure 5.
Timing diagram illustrating the temporal flow of activity.

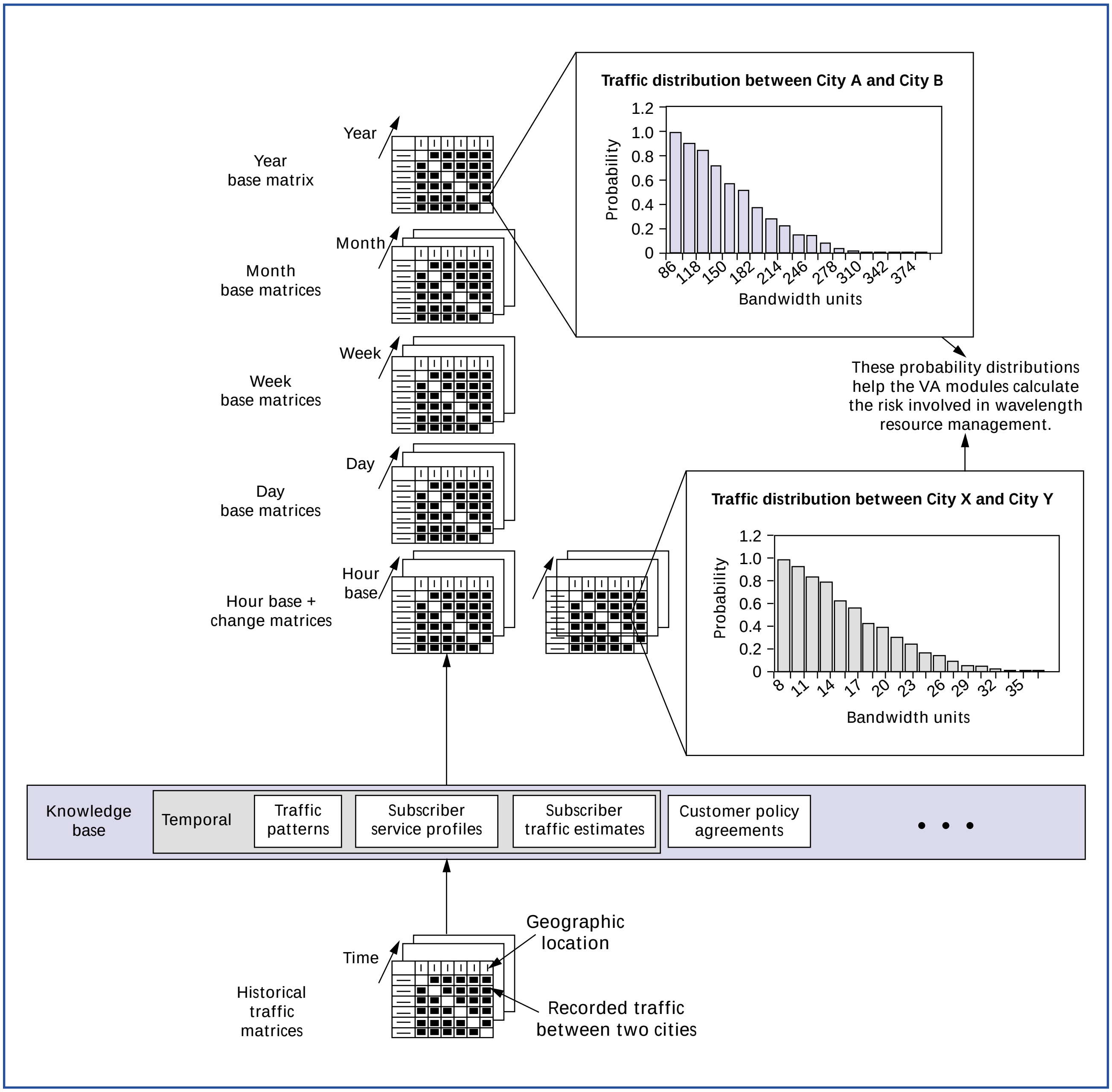


Figure 6.
Dynamic traffic matrices generation and representation.

Furthermore, let us assume that the CWM module in the intelligent optical layer monitors and keeps track of the historical traffic matrices in its database. Finally, let us assume that the optical network's communications world model—with the help of input processing modules—generates traffic patterns, up-to-date service profiles, customer service policy agreements, and subscriber growth estimates. Based on the output of the optical network's CWM module and the input processing modules, traffic estimates can be generated.

Figure 6 depicts the temporal and spatial knowledge represented in the *dynamic traffic matrices*. The matrices are organized into hierarchical clusters based on time, and values in the matrices are represented as distributions, to optimize space. From the historical traffic matrices and the knowledge base, traffic matrices for an hour are generated. A particular hour's traffic is represented by two sets of traffic matrices: one is composed of *base* (that is, *invariant bandwidth*) *matrices*; the other, of *dynamic change* (that is, *variant bandwidth*) *matrices*. From the set of all *hour invariant traffic* matrices for a particular day of the week, an invariant traffic matrix for that day is generated. From the set of all invariant traffic matrices for the days in a month, an invariant matrix for the month is generated. From the set of all invariant traffic matrices for the months in a year, an invariant traffic matrix for the year is generated. This process of generating invariant and variant traffic matrices can be carried out on a small time scale or on a scale of years, depending upon the needs of the service provider. Because of the large amount of information available, the matrix values are represented as distributions that consolidate the data from several matrices. On the right-hand side of Figure 6, the distributions for *year base matrix* and *hour change matrix*—generated by the BG module, using simulation algorithms—are shown. These distributions help the VA modules calculate risk, based on the service provider's risk-acceptance levels for use of their transport resources. The logical all-optical network design algorithms[48] (for example, mixed-integer programming models) in the BG modules use the traffic matrices to route wavelengths in the network to allocate bandwidth on demand. The hierarchical organization of the matrices into matrices that are variant and invariant reduces computational overhead and improves the performance of the algorithms in responding to changes in real time. This helps in the automation of the bandwidth delivery technology needed for adaptable optical networks.

Planning in Intelligent Communication Networks

Planning in communication networks implies an ability to predict future states of the network environment. Network behavior and traffic prediction algorithms typically use recent historical data to compute parameters that can be used to extrapolate future network states.[49] Experience shows that predictions made by such methods are typically not reliable for periods longer than the historical interval over which the parameters were computed. Thus, at each level, planning horizons extend into the future only about as far, and with about the same level of detail, as historical traces reach into the past. Predicting the future state of the network environment world often depends on assumptions as to what actions are going to be taken and what responses are to be expected from the communication environment, including what actions may be taken by other intelligent agents or by the end users. Planning of this type requires searching and stochastic modeling[50] over the space of possible future actions and probable responses.

Several levels of planning, illustrating the planning horizon and successive lower levels of the hierarchy, are depicted, as an example, in **Figure 7**. At the top level, a single task is decomposed into a set of planned subtasks for each of several subsystems. At each of the following levels, a planned task is further decomposed into subtasks at the next lower level. Planning complexity grows exponentially with the number of steps in the plan. If planning is to succeed, every planning algorithm must operate in a limited domain space. If there is too much granularity in the time line—or in the space of possible actions—the size of the search space can easily become too large for timely response.[49] One method of resolving this problem is to use a multiplicity of planners (that is, automated planning algorithms) in hierarchical layers[43] so that at each layer no planner needs to search more than a given number of steps deep in a search space, and at each level there are no more than a given number of subsystem planners that need to generate and coordinate plans simultaneously. At each level, communication plans consist of several subtasks. In a complex network environment, communication plans must be regenerated periodically to cope with changing and unforeseen conditions in the network. Cyclic replanning may occur at periodic intervals. Emergency replanning begins immediately upon the detection of an unexpected event or condition, such as a cable cut or node failure in a network, or a fraud event.

Intelligent network plan executor algorithms at each level are responsible for reacting to feedback at every response cycle interval. If the feedback indicates the failure of a planned subtask, the executor branches immediately to a preplanned emergency subtask.

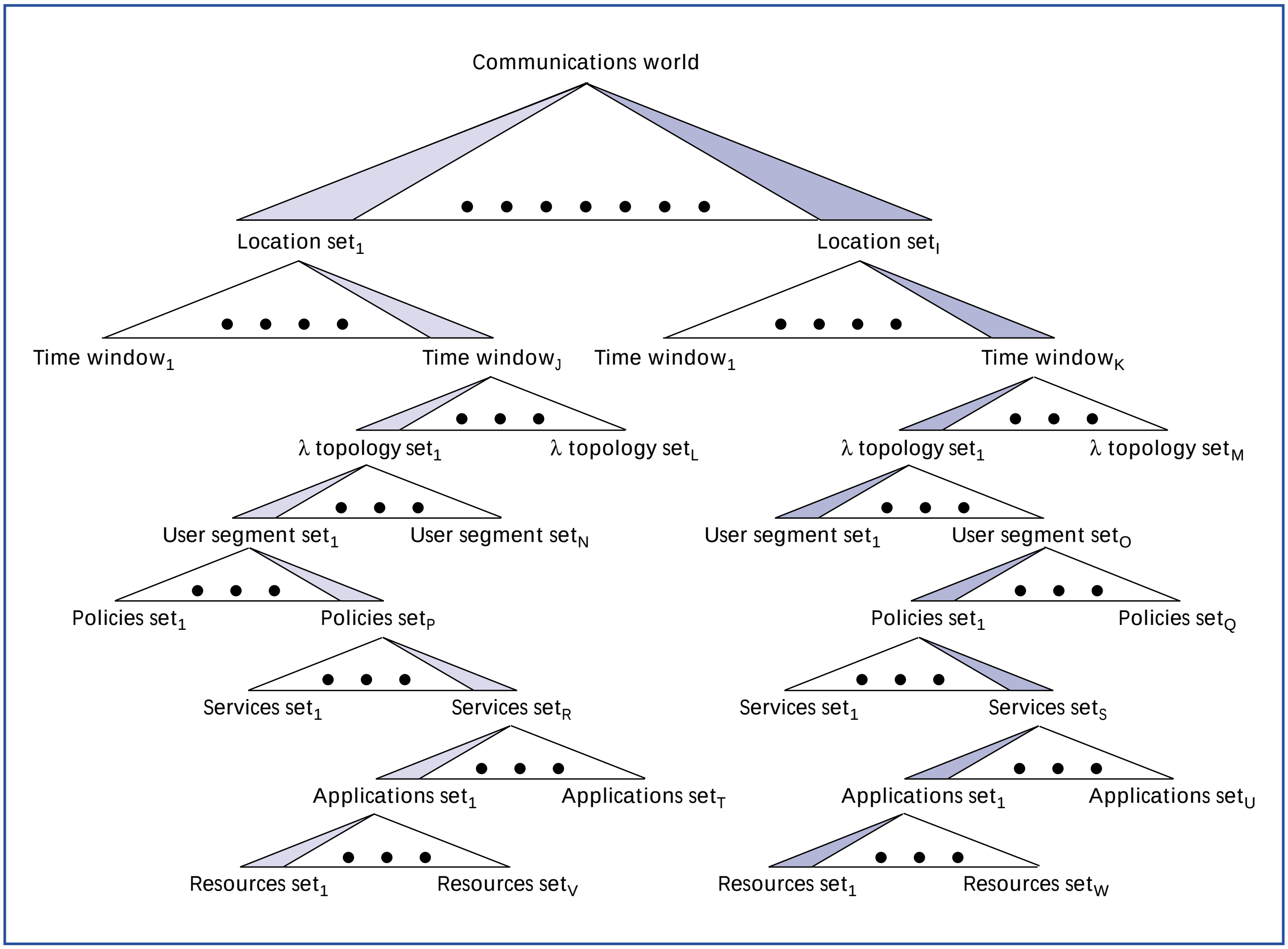


Figure 7.
Levels of planning illustrating planning horizon.

The planner simultaneously selects or generates an alternate or error recovery sequence that is substituted for the plan that failed. When a task goal is achieved at time t = 0, it becomes a task completion event in the historical trace. To the extent that an historical trace is an exact duplicate of a former plan, it can be said that there were no surprises; that is, the plan was followed and every task was accomplished as planned. To the extent that an historical trace is different from the former plan, it can be said that there were surprises. The average size and frequency of surprises (that is, differences between plans and results) is a measure of the effectiveness of the intelligent planning algorithms.

Self-Organizing Intelligent Networks

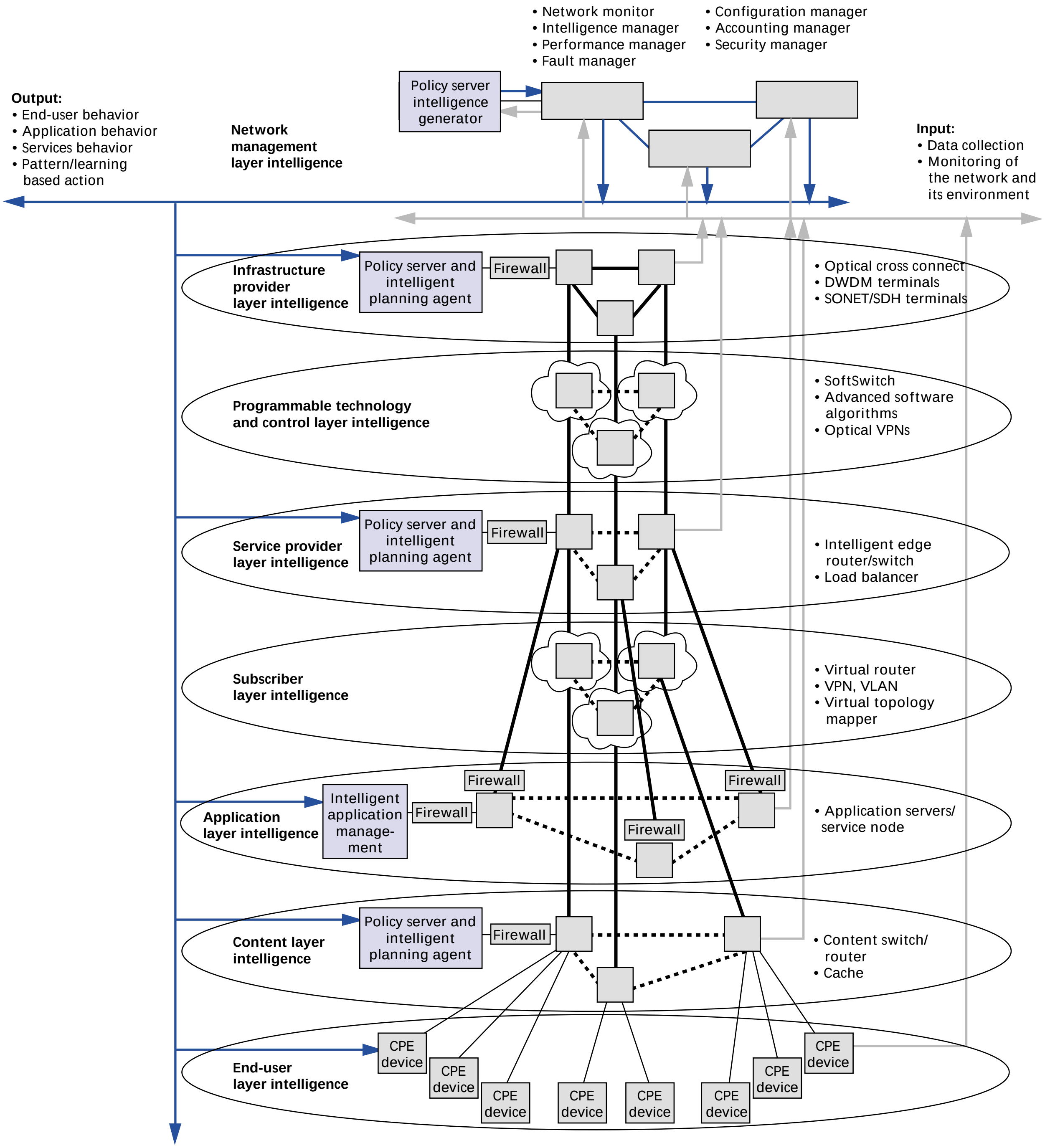
Fundamentally new and disruptive technologies are producing new networking applications, which, in turn, are creating a far greater demand for communication services and resources than in the past.[37] Today, there are tens of thousands of private networks, and service providers have an opportunity to replace them with VPNs. There are also tens of millions of residential and commercial users who use the network for various communication needs. Understanding the behavior of the users, applications, services, markets, and other factors in the fast-changing world of communications and applying this knowledge to network management is difficult to do, using the approaches that are traditionally employed in the current communications paradigm. Therefore, we introduce a new paradigm based on *self-organizing networks* that makes it possible to manage large, complex networks and environments efficiently and to deploy, provision, and manage new, high-value services rapidly, with an overall reduction in manual intervention.

This section describes a self-organizing intelligent networks paradigm in which networks can reconfigure network topologies and provision resources and services dynamically. It also describes an end-to-end intelligent network application, shown in **Figure 8**, in which a network monitors itself, learns about its environment and the environment's impact on network resources, makes intelligent decisions, and takes appropriate actions based on prior network behavior—as observed over time—on an application or service basis. Figure 8 is essentially the same as Figure 2, with the addition of a self-organizing capability made possible by the use of an IRP feedback loop. It is assumed that the network elements in Figure 8 are interconnected by a system similar to the management communications network[40] system. In Figure 8, these interconnections are shown as gray lines for input processing and dark-blue lines for output response processing (or feedback). The figure illustrates the important role the network management system plays in the self-organizing network paradigm.

Figure 9, which is based on the architecture proposed in the section "Architecture for Understanding Network Intelligence," shows the self-organizing network in more detail, illustrating both the hierarchical and horizontal relationships involved. The architecture is hierarchical in that the commands and status feedback flow hierarchically up and down a behavior-generating chain of command and the input-processing and CWM functions have hierarchical levels of temporal and spatial aggregation, as described in the section "Temporal and Spatial Relationships in Intelligent Networks." During network operation, goal-driven mechanisms in the BG modules assess aggregate priorities, negotiate for resources, and coordinate task activities in order to select among the possible communication paths. To maintain control of this operation, each BG module accepts task commands from only one supervisory process at a time; hence, at any given moment, the BG modules form a command tree. The architecture is horizontal in that data is shared horizontally among heterogeneous functional modules at the same level of the network. At each hierarchical level, the architecture is horizontally interconnected by communication pathways between BG, CWM, IRP, and VA modules in the same node, and between nodes at that level, especially within the same command subtree.

Figure 9 shows an organization of processing nodes in which the BG modules form a command tree. The functional characteristics of the BG modules at each level, the type of environmental attributes or entities recognized by the IRP modules at each level, and the type of processing subsystems form the command tree. The specific configuration of the command tree is service- and application-dependent and, therefore, not necessarily stationary over time. Figure 9 illustrates three possible dynamic configurations that may exist at different points in time. These different configurations are shown by links in three different colors, which are associated with three different time windows. During operation, relationships between modules in the same or different layers of the hierarchy may be reconfigured in response to changes in goals, priorities, or task requirements. This means that any particular computational node—with its BG, CWM, IRP, and VA modules—may belong to one subsystem at one time and to a different subsystem a short time later. The specific configurations are determined by the application of the automated planning process (discussed in the section "Planning in Intelligent Networks") and by the information collected from the spatio-temporal properties of the network elements and the network environment (discussed in the section "Temporal and Spatial Relationships in Intelligent Networks").

The reconfiguration of the command tree can be implemented through multiple pathways that exist, but are not always activated, among BG modules at different hierarchical levels. These multiple pathways define a layered graph of nodes and directed arcs. They enable each BG module to receive input messages and parameters from several different sources. Each layer of the system architecture contains a number of nodes, each of which contains BG, CWM, IRP, and VA modules. The nodes are richly, but not fully, interconnected. For example, in an all-optical DWDM network elements layer (such as that shown in Figures 8 and 9), some



CPE – Customer premises equipment
DWDM – Dense wavelength division multiplexing
SDH – Synchronous digital hierarchy
SONET – Synchronous optical network
VLAN – Virtual local area network
VPN – Virtual private network

Figure 8.
Completely automated end-to-end self-organizing intelligent network application.

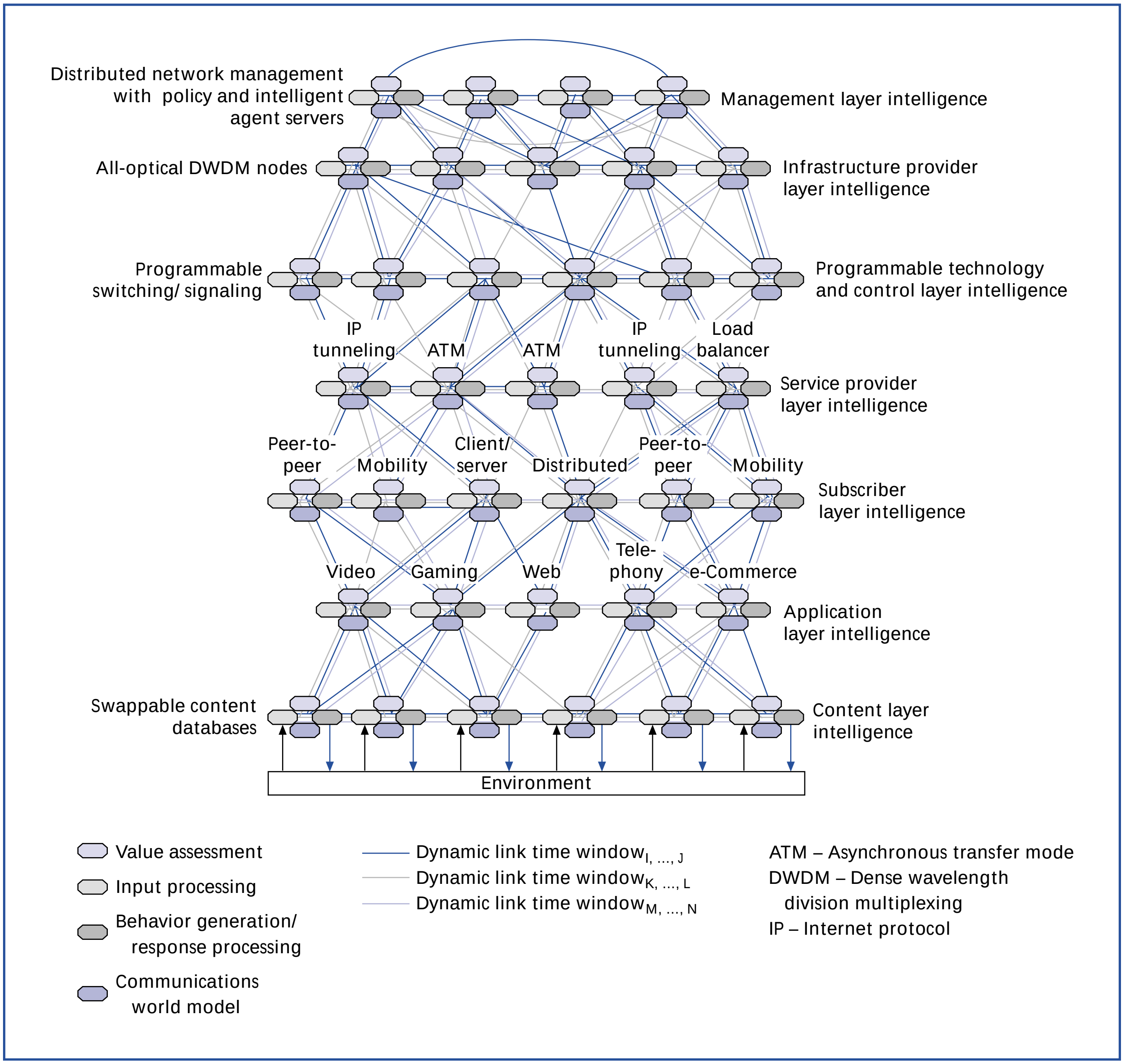


Figure 9.
Dynamically interconnected layered graph representing self-organizing network.

of the outputs from the BG modules drive the micromechanical mirror motor actuators, and the inputs to the layer's IRP modules convey data from the environment. During operation, goal-driven communication path selection mechanisms configure this lattice structure into the organization tree shown in Figure 9. The IRP modules are also organized as a layered graph. At each higher level, input information is processed into increasingly higher levels of abstraction, as input processing pathways branch and merge in different ways.

The following subsections provide a few examples of the functioning of and the interactions between different layers in a self-organizing network.

Network Management Layer

The network management layer plans activities and allocates resources for one or more layers below it for a period specified in the historical trace patterns.

At the management layer, such data as requests for provisioning and bandwidth orders are consolidated into batches to optimize the use of resources, and a schedule is generated for the layer(s) to process the batches. At the management layer, the CWM module maintains a knowledge database containing the names, contents, and attributes of batches, as well as an inventory of the resources required to provide the requested bandwidth and services. Historical traces may describe the temporal bindings of services, routing, and bandwidth between nodes. The IRP module processes compute information about the flow of services, the level of inventory, and the operational status of all the nodes involved in the network. The VA module computes the cost and benefits of various batches and routing options and calculates statistical service-confidence data. An operator interface allows service provider technicians to visualize the status of bandwidth and service requests, inventory, the flow of resources, and the overall situation within the entire network. Operators can intervene to change priorities and redirect the flow of resources and services. Planners can keep track of how well plans are being followed and can modify parameters as necessary to keep to a plan. The output from the management layer provides work-flow assignments for the underlying nodes.

Interactions Between the Infrastructure Provide Layer and the Service Provider Layer

This subsection briefly describes how multilayer intelligence, using IP tunneling and optical switching nodes, can provide agile networking. Service providers can optimize the use of their network resources by using IP tunneling and the reconfigurability of the optical network, and customers can manage their own VPN network resources by using automated network management and provisioning[51] to establish service whenever and wherever it is needed.

When bandwidth is available in a *lightpath* (that is, an unused but allocated resource), the network management layer uses the unused bandwidth to create a secure IP tunnel for another customer's bandwidth request. For this kind of operation, the CWM, BG, VA and IRP modules of the optical layer and the IP tunneling layers have to work synchronously, sharing their CWM knowledge about such data as:

- Endpoints,
- SLA parameters, such as path characteristics, diverse routing requirements, and inclusion or exclusion of certain nodes in the path,
- QoS parameters, such as maximum delay, jitter, and restoration types,
- Billing options, such as flat-rate and usage-based billing, and
- User authentication and authorization.

The intelligent tunneling and optical virtual path BG modules provide processing-intense filtering, forwarding, accounting, and QoS/SLA functions in the tunneling and optical routing switches. In addition, the CWM modules maintain—and the BG modules process—features offering capabilities (such as QoS, isolation, and policing) that allow operators to deliver flexible, measurable, and enforceable SLAs, making possible the delivery of real-time services over the network. This flexibility allows the switches to configure multiple dynamic virtual routers (VRs) and dynamic virtual optical paths (VOPs) with routing and policy domains that can be shared by any number of service providers. Large providers can be assigned dedicated VRs or VOPs on the optical lightpath, and small providers can share VRs or VOPs that are managed administratively by the service provider. The VA modules evaluate and enforce network policies on admission control and rate limitations to ensure that all SLAs can be met.

Application Layer

One of the functions of the VA modules that support the application layer of intelligence is to provide the CWM and BG modules with ratings reports for the applications at regular intervals. This information allows the BG modules to isolate the most popular and highly rated applications on the network on demand. Such ratings generation provides service providers with a competitive edge, by allowing them to change subscription plans to increase revenue and to redirect unused resources to support especially popular applications.

End-User Layer

In an end-to-end intelligent network, intelligence in an end-user device, when integrated properly with the other intelligent layers, would play a major role, contributing to and making use of the intelligence in the network. The importance of this layer is that it makes it possible to control the amount of end-user behavioral information that is generated at the layer where it originates, so that only useful information is communicated to the other layers. In this context, it is assumed that an end-user gateway device (for example, a residential gateway) has an integrated intelligence-gathering functionality, and that it monitors end-user needs and behavior patterns and encodes the information it gathers into a data structure called an *end-user diary*. Intelligence-gathering mechanisms can automatically and invisibly keep track of end-user behavior without creating overhead on the network; the information collected may be processed and communicated securely to other intelligent network layers when traffic on the network is minimal (for example, during the night). **Figure 10** shows the intelligence exchange flow between the IRP-CWM-BG-VA modules of an intelligent residential gateway and the IRP-CWM-BG-VA modules of the intelligent network management layer. The intelligent network management layer receives all the end-user diaries and updates the user profile database. In return, it sends each end-user gateway device information about the predicted traffic load on the network layers for the following day, based on expected end-user service needs for that day. The information about the status of the network allows the BG modules to make better choices when selecting network routes and application servers from among the available alternatives.

Within the proposed framework, each layer controls and resolves issues within its purview. At each layer, inputs and requests are received and responses planned by the IRP module, evaluated—based on information from the CWM module—by the VA module, and implemented by the BG module. In the above examples, the network management layer would create and modify the networking topological structure and assign end users within that structure to provide the administrative structure that best accords with the rules and policies of the service provider.

The specific functions given in the above examples are for illustrative purposes only. They are meant only to illustrate how the generic structure and function of the proposed framework might be instantiated by a service provider. The point of these examples is to illustrate how multilevel hierarchical architecture integrates real-time planning and execution behavior with dynamic CWM, knowledge representation, and input. At each level, behavior generation is guided by VA modules that optimize plans and evaluate results. The system architecture organizes the planning of behavior, the control of action, and the focusing of computational resources. The result is an intelligent real-time self-organizing network system that is both driven by high-level goals and responsive to input feedback.

The major benefit of the self-organizing intelligent network paradigm is that it enables service providers to use network resources efficiently, based on network needs that are dependent upon both spatio-temporal changes in the network and changes in the network environment.

Conclusion

Because of the way network technology is being developed today, it is difficult to know—before it is deployed—if an end-to-end setup of networking devices will really be beneficial in the field. One reason for the difficulty is that the behavior of the network layers is not well understood and is still evolving; another is the rapid growth in the number of applications on the networks. A variety of technology trends have made it possible to incorporate computational capability in all network devices. Unfortunately, although shrinking technology has made networking devices smart, it has not always improved their usability.

The easiest way to improve the applicability of networking devices (and the first step toward doing so) is to provide a uniform framework for understanding network functional interconnectivity between the devices and the systems that interact with them. This paper has presented a framework for understanding end-to-end intelligent communication networks. (An example reference model has been presented as part of the framework.) The framework incorporates the functional elements of intelligence into IRP, CWM, VA, and BG computational modules. These modules

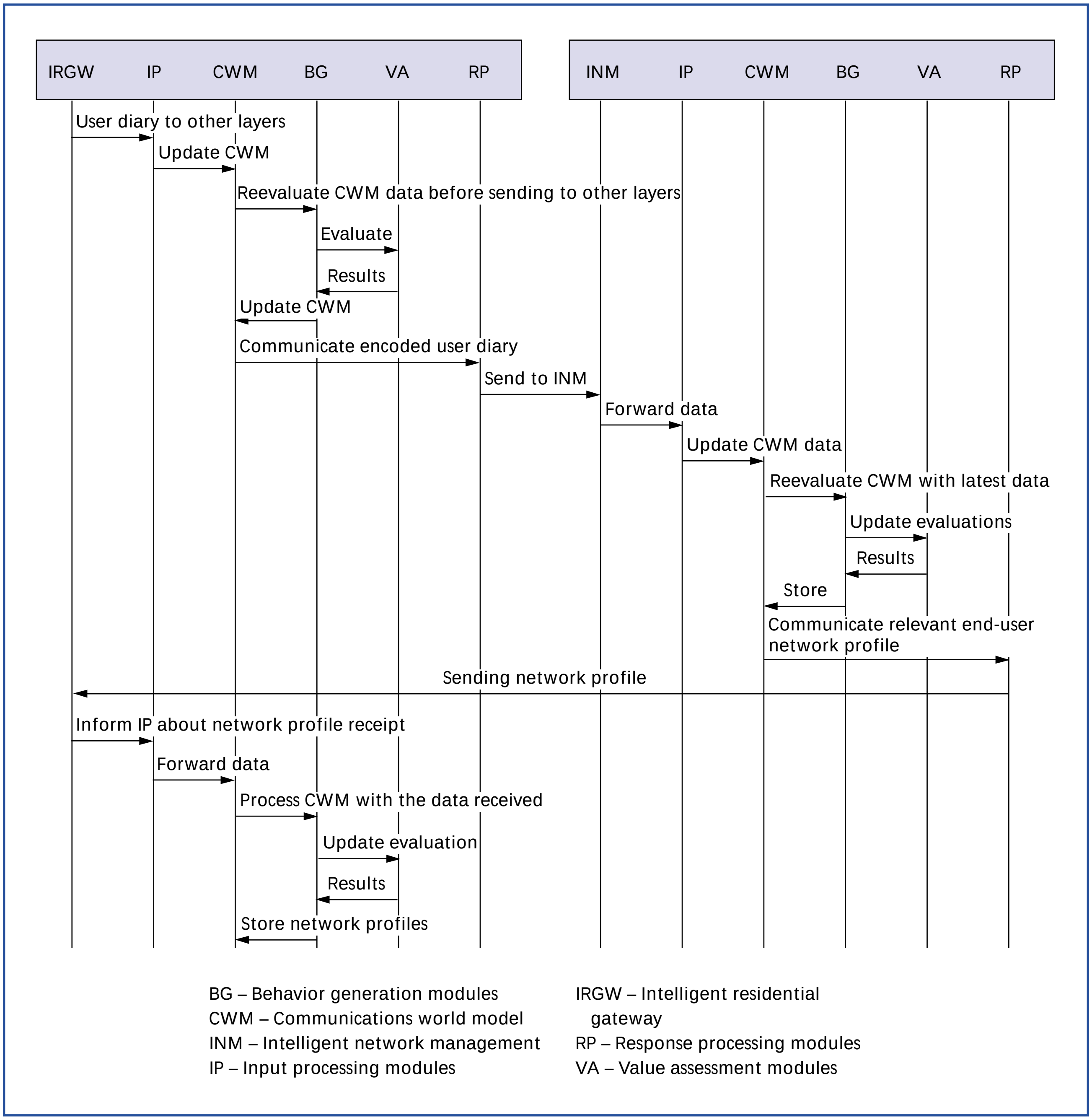


Figure 10.
Intelligence update control flow between end-user gateway and network management.

are interconnected into networks and hierarchies with logical, spatial, and temporal properties. Within this framework, the BG modules direct and focus input processing algorithms on network objects of importance, ignoring tasks that are not important to current goals and priorities. The CWM modules provide information in response to queries from the BG modules, make predictions, and receive updates.

Based on the framework presented, a networking paradigm of self-organizing networks has been introduced as an application. Self-organizing network systems can accelerate the introduction of a new generation of services and resource management. End-to-end service and resource management can be enabled through customer and service focus, automation, and open interfaces that facilitate integration of these systems with service provider systems and third-party applications.

The implementation and embodiment of the proposals presented will require a consistent application across a number of products over a period of time. To date, such efforts have generally been confined to a single layer of the overall framework.

Acknowledgements

The authors would like to acknowledge Bharat T. Doshi and Bruce L. Nelson for their valuable suggestions for improving the quality of this paper. The authors would also like to acknowledge Pablo Martinez, Steve D. Simon, Krishna K. Dhara, Thomas Mueller, Thomas S. Brancati, and Sailaja Mummidi for their support and for their comments on and discussions of the subject matter of this paper.

*Trademarks

Content Smart is a trademark of ArrowPoint Communications.

References

1. T. Horan, S. Kamman, J. M. Firlings, "Smart Packet Networks—Battle for the Soul of a New Network," CIBC World Markets, Equity Research Report, Oct. 16, 2000.
2. F. M. Fenton and J. D. Sipes, "Architectural and Technological Trends in Access: An Overview," *Bell Labs Tech. J.*, Vol. 1, No. 1, Summer 1996, pp. 3–10.
3. S. Corley, M. Tessellar, J. Cooley, J. Meinkohn, F. Malabocchia, and F. Garijo, "The Application of Intelligent and Mobile Agents to Network and Service Management," *Proc. 5th Intl. Conf. on Intelligence in Services and Networks (IS&N)*, Antwerp, Belgium, May 25–28, 1998, pp. 127–138.
4. R. A. Lakshmi-Ratan, "The Lucent Technologies Softswitch—Realizing the Promise of Convergence", *Bell Labs Tech. J.*, Vol. 4, No. 2, Apr.–June 1999, pp. 174–195.
5. A. Bieszczad, P. K. Biswas, W. Buga, M.Malek, and Hai Tan, "Management of Heterogeneous Networks with Intelligent Agents," *Bell Labs Tech. J.*, Vol. 4, No. 4, Oct.–Dec. 1999, pp. 109–135.
6. J. Anderson, J. S. Manchester, A. Rodriguez-Moral, and M. Veeraraghavan, "Protocols and Architectures for IP Optical Networking," *Bell Labs Tech. J.*, Vol. 4, No. 1, Jan.–Mar. 1999, pp. 105–124.
7. D. S. Furman, M. J. Cosky, D. L. Thomson, S. A. O'Brien, and E. E. Summer, Jr., "Speech-Based Services," *Bell Labs Tech. J.*, Vol. 4, No. 2, Apr.–June 1999, pp. 88–97.
8. A. N. Netravali, "When Networking Becomes Second Nature: The Next 25 Years…and Beyond," *Bell Labs Tech. J.*, Vol. 5, No. 1, Jan.–Mar. 2000, pp. 203–214.
9. B. Rajagopalan, D. Pendarakis, D. Saha, R. S. Ramamoorthy, and K. Bala, "IP Over Optical Networks: Architectural Aspects," *IEEE Communications Magazine*, Vol. 38, No. 9, Sept. 2000, pp. 94–102.
10. D. Kerr. D. O'Sullivan, R. Evans, R. Richardson, and F. Somers, "Experiences Using Intelligent Agent Technologies as a Unifying Approach to Network Management, Service Management and Service Delivery," *Proc. 5th Intl. Conf. on Intelligence in Services and Network, (IS&N)*, Antwerp, Belgium, May 25–25, 1998, pp. 115–126.
11. Spring Tide Networks–Lucent Technologies, White Paper, "IP Service Intelligence at the Edge," Apr. 2000, <http://www.springtidenetworks.com/pdfs/ipservintelligence_cm.pdf>.
12. "Specification of the Bluetooth System," Dec. 1999, <http://www.bluetooth.com/developer/specification/specification.asp>.
13. M. Moganti, A. Sankalia, K. K. Dhara. "Cable-Telephony Services Using a Media Terminal Adapter with Integrated Cellular Transceiver Device," U. S. Patent Office Application Nos. 9/677,060 and 09/677,054, filed Sept. 29, 2000.
14. ArrowPoint Communications–Cisco Systems, White Paper, "Content Smart* Cache Switching," <http://www.arrowpoint.com/solutions/white_papers/printer/cache_switching.html>.
15. Alteon Web Systems–Nortel Networks, White Paper, "Layer 7 Web Switching: Technical overview on scaling Next Generation Web Infrastructure with Content-Intelligent Switching," <http://www.alteonwebsytstems.com/products/whitepapers/layer7switching/>.
16. P. Krishnan, D. Raz, and Y. Shavitt, "The Cache Location Problem," *IEEE/ACM Trans. on Networking*, Vol. 8, No. 5, Oct. 2000, pp. 568–582.
17. M. E. Anagnostou, "Optimal Distribution of Service Components," *Proc. 5th Intl. Conf. on Intelligence in Services and Networks (IS&N)*, Antwerp, Belgium, May 25–28, 1998, pp.17-30.
18. D. M. Ogle, K. Schwan, and R. Snodgrass, "Application-Dependent Dynamic Monitoring of Distributed and Parallel Systems," *IEEE Trans.*

on Parallel and Distributed Systems, Vol. 4, No. 7, July 1993, pp. 762–778.
19. A. Feldmann, A. C. Gilbert, P. Huang, and W. Willinger, "Dynamics of IP traffic: A study of the role of variability and the impact of control," *ACM SIGCOMM '99*, Cambridge, Mass., Aug. 30–Sept. 3, 1999.
20. J. R. Dianda, R. O. Colbert, P. J. L. Herve, and T. Yang, "Programmable Service Platforms for Converged Voice/Data Services," *Bell Labs Tech. J.*, Vol. 5, No. 3, July–Sept. 2000, pp. 43–58.
21. M. S. Borella, "Source Models of Network Game Traffic," *Computer Commun.*, Vol. 23, No. 4, Feb. 15, 2000, pp. 403–410.
22. M. L. F. Grech, R. D. McKinney, S. Sharma, J. J. Stanaway, Jr., D. W. Varney, and K. V. Vemuri, "Delivering Seamless Services in Open Networks Using Intelligent Service Mediation," *Bell Labs Tech. J.*, Vol. 5, No. 3, July–Sept. 2000, pp. 186–202.
23. H. J. Wang, B. Raman, Chen-Nee Chuah, R. Biswas, R. Gummadi, B. Hohlt, Xia Hong, E. Kiciman, Zhuoqing Mao, J. S. Shih, L. Subramanian, B. Y. Zhao, A. D. Joseph, and R. H. Katz, "ICEBERG: An Internet Core Network Architecture for Integrated Communications," *IEEE Personal Commun.*, Vol. 7, No. 4, Aug. 2000, pp. 10–19.
24. T. Clark, *Designing Storage Area Networks: A Practical Reference for Implementing Fibre Channel SANs*, Addison-Wesley, Reading, Mass., 1999.
25. K. G. August, V. B. Lawrence, and B. R. Saltzberg, "An Introduction to Future Communications Services and Access," *Bell Labs Tech. J.*, Vol. 4, No. 2, Apr.–June 1999, pp. 3–20.
26. S. K. Rao, "An Architecture for Adaptable Wireless Networks," M.S. Thesis, Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, 2000.
27. W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," *Proc. of 5th Annual Joint ACM/IEEE Intl. Conf. on Mobile Computing and Networking* (MOBICOM '99), Seattle, Wash., Aug. 15–20, 1999.
28. J. M. Smith, K. L. Calvert, S. L. Murphy, H. K. Orman, L. L. Peterson, "Activating Networks: A Progress Report," *Computer*, Vol. 32, No. 4, Apr. 1999, pp. 32–41.
29. Chromatis™ – Lucent Technologies, White Paper, "Selective WDM: Removing the Economic Barriers to Metro DWDM Deployment," Oct. 3, 2000, <http://www.lucent.com/ins/products/chromatis>.
30. D. Litman, A. Mishra, and P. F. Patel-Schneider, "Modeling Dynamic Collections of Interdependent Objects Using Path-Based Rules," *Proc. 12th Annual Conf. on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA '97)*, Atlanta, Ga., Oct. 5–9, 1997, pp. 77–92.
31. R. J. Brachman, "Reducing Classic to Practice: Knowledge Representation Theory Meets Reality,'' *Proc. of 3rd Intl. Conf. on Principles of Knowledge Representation and Reasoning*, Boston, Ma., Oct. 25–29, 1992, pp. 247–258.
32. M. Moganti, "Rule Matching and Indexing Techniques," in *Handbook of Data Management*, edited by B. Thuraisingham, Auerbach Pulbications, Boca Raton, Fl., 1998.
33. M. Minsky, "A framework for representing knowledge," in *The Psychology of Computer Vision*, edited by P. H. Winston, McGraw-Hill, New York, 1975, pp. 211–277.
34. L. Sorbello, and P. Demeester, "Network Planning in Uncertain and Dynamic Environments," *Networks 2000*, Toronto, Canada, Sept. 10–15, 2000.
35. R. R. Yager, J. Kacprzyk, M. Fedrizzi, *Advances in the Dempster-Shafer Theory of Evidence*, Wiley, New York, 1994.
36. G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty, and Information*, Prentice Hall, Englewood Cliffs, New Jersey, 1988.
37. D. Mitra, K. E. Sahin, R. Sethi, and A. Silberschatz, "New Directions in Services Management," *Bell Labs Tech. J.*, Vol. 5, No. 1, Jan.–Mar. 2000, pp. 17–34.
38. S. Yazid and B. Bou-Diab, "Emulation of a Vast Adaptive Network," *Networks 2000*, Toronto, Canada, Sept. 10–15, 2000, <http://www.ntonc.org/docs/PW_Emulation-Net2000.pdf>.
39. S. V. Ahamed and V. B. Lawrence, *Design and Engineering of Intelligent Communication Systems*, Kluwer Academic Publishers, Boston, Mass., 1997.
40. R. V. Shankar and A. K. Chatterjee, "Managing the Management Communications Network in Optical Transport Systems," *Bell Labs Tech. J.*, Vol. 4, No. 4, Oct.–Dec. 1999, pp. 155–170.
41. B. Hayes-Roth, "A blackboard architecture for control," *Artificial Intelligence*, Vol. 26, No. 3, July 1985, pp. 252–321.
42. G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information," in *The Psychology of Communication—Seven Essays*, Basic Books, New York, 1970.
43. A. Meystel, "Theoretical foundations of planning and navigation for autonomous robots,"

International Journal of Intelligent Systems, Vol. 2, No. 2, 1987, pp. 73–128.
44. J. Jiao, S. Naqvi, D. Raz, and B. Sugla, “Towards Efficient Monitoring,” *IEEE Journal on Selected Areas in Communications (JSAC)*, Vol. 5. No. 18, May 2000, pp. 723–732.
45. K. Thompson, G. J. Miller, and R. Wilder, “Wide-Area Internet Traffic Patterns and Characteristics,” *IEEE Network*, Vol. 11, No. 6, Nov.–Dec. 1997.
46. T. Mowbray and R. Zahavi, *The Essential CORBA: Systems Integration Using Distributed Objects*, Wiley, New York, 1995.
47. S. Baker, *CORBA Distributed Objects Using Orbix*, Addison-Wesley, Reading, Mass., 1997.
48. H. Harai, M. Murata, and H. Miyahara, “Heuristic Algorithm for Allocation of Wavelength Convertible Nodes and Routing Coordination in All-Optical Networks,” *J. of Lightwave Tech.*, Vol. 17, No. 4, Apr. 1999, pp. 535–545.
49. D. Lambert, J. C. Pinheiro, and D. X. Sun, “Estimating millions of dynamic timing patterns in real time,” *J. of the American Statistical Assoc.*, Vol. 96, No. 453, Mar. 2001, pp. 316–330.
50. P. Kall and S. W. Wallace, *Stochastic Programming*, Wiley, New York, 1994.
51. A. R. McGee, S. R. Vasireddy, K. J. Johnson, U. Chandrashekar, and S. H. Richman, “Dynamic Virtual Private Networks,” to appear in *Bell Labs Tech. J.*, Vol. 6, No. 2, July–Dec. 2001.

(Manuscript approved April 2001)

MADHAV MOGANTI is a member of technical staff in the Network Solutions Architecture Organization at Lucent Technologies in Holmdel, New Jersey. He holds B.S. and M.S. degrees in computer science and artificial intelligence from Andhra University, Visakhapatnam, India, and the University of Hyderabad, Hyderabad, India, respectively. He received dual M.S. and Ph.D. degrees in engineering management and computer science, respectively, from the University of Missouri-Rolla. His areas of interest include next-generation network architectures, network intelligence, service and behavior characterization, and network design and simulation algorithms. He has also worked extensively in the areas of image understanding, multi-dimensional indexing in large databases, parallel computing and processing, and distributed artificial intelligence. Dr. Moganti has more than 15 journal and refereed conference publications, including book chapters. He holds one patent and has four patents pending in the networking area.

STEVEN F. KNITTEL is a technical manager in the Network Solutions Architecture organization within the Network Solutions Group at Lucent Technologies in Holmdel, New Jersey. He leads the Verification Requirements systems engineering group. He holds a B.S. degree in electrical engineering from Columbia University in New York City and an M.S. degree in electrical engineering from Syracuse University in New York. Mr. Knittel has a broad interest in services and network architectures and holds two patents in the area, one of which was recently recognized as an AT&T Strategic Patent. Currently, he and his team are engaged in designing and verifying Network Hosted Services data centers and end-to-end solutions. ◆