Μεσολαβητικό Λογισμικό - *Μεσολογισμικό*

Middleware

Μ. Δικαιάκος

ΕΠΛ602

Παραπομπές

- “Network Policy and Services: A Report of a Workshop on Middleware ,” RFC 2768, IETF

Εισαγωγή

- In advanced networks, middleware consists of services and other resources located between both the applications and the underlying packet forwarding and routing infrastructure.
- No consensus currently exists on the precise lines of demarcation that would define those domains.
- RFC 2768 defines middleware core components within the framework of the current status of middleware-related standards activities, especially within the IETF and the Desktop Management Task Force (DMTF).

Πλαίσιο

- Middleware can be defined to encompass a large set of services. For example, we chose to focus initially on the services needed to support a common set of applications based on a distributed network environment.

- There is really no core set of middleware services in the sense that all applications required them. Many communities (e.g., Internet2, NGI) may decide on their own set of common middleware services and tools; however, they should strive for interoperability whenever possible.

- Although middleware could be conceptualized as hierarchical or layered, such an approach is not helpful. The better approach would be to consider middleware as an unstructured, often orthogonal, collection of components (such as resources and services) that could be utilized either individually or in various subsets.

Τι είναι το Μεσολογισμικό;

- There is agreement on the existence of middleware, but the definition of middleware is dependent on the subjective perspective of those trying to define it.
- Definition depends on when the question is asked, since the middleware of yesterday (e.g., DNS, PKI, and Event Services) may become the fundamental network infrastructure of tomorrow.
  - Application environment users and programmers see everything below the API as middleware.
  - Networking gurus see anything above IP as middleware.
  - Those working on applications, tools, and mechanisms between these two extremes see it as somewhere between TCP and the API.
- Middleware can be further classified into:
  - application-specific upper middleware
  - generic middle middleware
  - resource-specific lower middleware.
- Middleware often extends beyond the "network" into the compute, storage, and other resources that the network connects.

Ορισμοί Μεσολογισμικού

- We characterize middleware as those services found above the transport (i.e., over TCP/IP) layer set of services but below the application environment (i.e., below application-level APIs).
- Middleware can be viewed as a reusable, expandable set of services and functions that are commonly needed by many applications to function well in a networked environment.
- This definition could further be refined to include:
  - Persistent services, such as those found within an operating system, distributed operating environments (e.g., JAVA/JINI), the network infrastructure (e.g., DNS).
  - Transient capabilities (e.g., run time support and libraries) required to support client software on systems and hosts.

Παραδείγματα Μεσολογισμικού

- A video serving application will want to access resource discovery and allocation services not just for networks but also for the archives and computers required to serve and process the video stream.
- Grids: effective high performance distributed computing requires distributed common computing and networking resources, including libraries and utilities for resource discovery, scheduling and monitoring, process creation, communication and data transport.
- Middleware components that support networked information discovery (Archie or Harvest). Site logs used by Archie or the broker system and harvest agents are an important middleware tool, and additional work is urgently needed in order to improve the efficiency and scope of web-based indexing services.
- Workshop on "Information Infrastructure for the Internet" (1994, RFC 1862) recommended for an increased focus on general caching and replication architecture, rapid deployment of name resolution services, and the articulation of a common security architecture for information applications.

Οπτική Γωνία των Εφαρμογών

- From an applications perspective, the network is just another type of resource that it needs to use and manage.

- To define the requirements for middleware services of the future, we take into consideration anticipated applications such as: distributed computing, distributed data bases, advanced video services, teleimmersion, extensions with haptics, electronic commerce, distance education, interactive collaborative research, high-rate instrumentation including use of online scientific facilities.

- Such applications manage large amounts of data, computation and information Grids, adaptable and morphing network infrastructure, proxies and agents, and electronic persistent presence (EPP).

- Many of these applications possess the capability to change the way the network is used as well as our definition of infrastructure, much as the Web and Mosaic changed it in the early 90s.

Απαιτήσεις Εφαρμογών για το Μεσολογισμικό

- Recent trends on Internet: Increasing amount of HTTP, voice, and video traffic. Voice and video particularly need some form of QoS and associated middleware to manage it.

- Online instruments: For security and efficiency we need to steer the devices and change parameters as a direct result of real-time analysis performed on the data as it is received from the instruments. Therefore, network requirements encompass high bandwidth, low latency, and security, which must all be coordinated through middleware.

- Digital libraries: The requirements they place on the network and on middleware are extensive, including support of authentication, authorization, access management, quality of service, networked information discovery and retrieval tools, naming and service location. They also require middleware to support collection building and self-describing data.

Απαιτήσεις Εφαρμογών για το Μεσολογισμικό

- Distributed computing environments (e.g., Globus, Condor, Legion, etc.) are quickly evolving into the computing and information Grids of the future. These Grids require:
  - adaptive and manageable network services
  - a sophisticated set of secure middleware capabilities to provide easy- to-use APIs to the application

- Authentication and access control, as well as security, are required for all of the applications mentioned above, albeit at different levels.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- **Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signaling)**
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- Πολιτική (Policy)
- Κατάλογοι (Directories)
- Διαχείριση Πόρων (Resource Management)
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Διαπροσωπείες και Σηματοδότηση

- Applications require the ability to explicitly request resources based on their immediate usage needs.
- These requests have associated network management controls and network resource implications; however, fulfillment of these requests may require multiple intermediate steps.
- There currently is no common framework or method, for an application to signal its need for a set of desired network services, including quality and priority of service as well as attendant resource requirements.

Διαπροσωπείες και Σηματοδότηση

We need a framework of standards that should probably include:

- Signaling methods.
- Access/admission controls.
- A series of defined services and resources.
- Service levels.
- Priority considerations.
- Scheduling.
- Service-Level-Agreement (SLA) functions.
- Feedback mechanism for notifying applications or systems when performance is below the SLA specification or when an application violates the SLA. Implies capabilities for:
  1. An interaction with some type of policy implementation and enforcement,
  2. Dynamic assessment of available network resources,
  3. Policy monitoring,
  4. Service guarantees,
  5. Conflict resolution, and
  6. Restitution for lack of performance.

Διαπροσωπείες και Σηματοδότηση

- Application programmers need a single API (minimal, common set of interfaces) for accessing common middleware services.
- Examples of common APIs that may be achievable are:
  - Environmental discovery interface, whether for discovering hardware resources, network status and capabilities, data sets, applications, remote services, or user information.
  - Remote execution interface, whether for distributed metacomputing applications, or for access to a digital library presentation service, or a Java analysis service.
  - Data management interface, whether for manipulating data within distributed caches, or replication of data between file systems, or archival storage of data.
  - Process management interface, whether for composing data movement with remote execution, or for linking together multiple processing steps.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- **Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)**
- Πολιτική (Policy)
- Κατάλογοι (Directories)
- Διαχείριση Πόρων (Resource Management)
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Ταυτοποίηση, Εξουδιοδότηση, Απογραφή

- Requirements arising for middleware:
  - Defining processes for access/admission control and identification (process for determining a unique entity)
  - Authentication (process for validating that identity)
  - Authorization (process for determining an eligibility for resource requests/utilization).
  - Accounting (at least resource utilization is recorded).
- Currently, AAA protocols exist, although not as an integrated model or standard. Considerations:
  - provide for various levels of granularity
  - provide for basic AAA mechanisms that can be used as a basis to support SLAs

Ταυτοποίηση, Εξουδιοδότηση, Απογραφή

- Requirements arising from the need to support distributed interoperation of middleware services that enable the distribution of application support across multiple autonomous systems: Third-party mechanisms for authentication and data movement
  - An application may need data under control of a remote collection to support the execution at a third site.
  - The data flow needs to be directly from the collection to the execution platform (for efficiency).
  - The procedure needs access permission to data-set while acting on behalf of the requestor.
  - Support for transitivity of trust.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- **Πολιτική (Policy)**
- Κατάλογοι (Directories)
- Διαχείριση Πόρων (Resource Management)
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Πολιτικές

- Are an integral component of all middleware services.
- Link high-level business requirements (like those specified in an SLA), to low-level device implementation mechanisms, e.g., specific access control & management of services, object and resources, configuration mechanisms necessary to provide a given service, etc.
- Are often represented as an "if condition then action" tuple.
- Can be both complex and numerous; therefore, policy management services must be able to identify and resolve policy conflicts. They also need to support both static (i.e. loaded at boot time via a configuration file) and dynamic (i.e., the configuration of a policy enforcing device may change based on an event) modes.

Πλαίσια Πολιτικών

- The IETF Policy Framework working group is addressing:
  - A policy framework definition language
  - A policy architecture model
  - Policy terminology
  - A policy model that can be used for signaled as well as provisioned QoS.
- A generalized policy management architecture includes:
  - A policy management service: supports specification, editing, administration of policy through a GUI and programmatically.
  - A dedicated policy repository: provides storage & retrieval policies, and policy components. Policy components can be used to derive more complex policies, and as part of policy decision and/or enforcement process.
  - At least one policy decision point (PDP) – e.g., a resource manager – handles events, makes decisions accordingly, updates the PEP configuration appropriately.
  - At least one policy enforcement point (PEP): router, firewall, or host. Enforces the policy based on “if condition then action” rule sets it receives from the PDP.
  - Policy information is communicated between PDP to PEP with protocols like COPS or DIAMETER.

Αλληλεπιδράσεις Πλαισίων Πολιτικής

- Policy requires AAA functions for access control and to establish the trust relationships that will enable distributed policy interactions. PDPs may require the requesting end systems and applications to be authenticated before the PDP will honor any requests.
- Policy requires interactions with entity identification mechanisms, resource identification mechanisms, and allocation mechanisms, because many policy processes link entities to resources. For instance, Globus uses its Resource Services Language (RSL) to define the resources and policies associated with them. The IETF has several policy definition languages in varying stages of development (RPSL, RPCL, SPSL, PFDL, PAX, and Keynote).
- Directories play a crucial role in policy systems.
  - Directories are ideal for storing and retrieving policy information: high read rates, ability to replicate their information, per-attribute access control, use of containment.
  - IETF Policy Framework working group is developing a core information model to represent policy information . This core model is used to provide common representation and structure of policy information in LDAP.
  - Applications can then subclass all or part of this core schema to meet their own specific needs, while retaining the ability to communicate and interoperate with each other.

Αποθήκευση Πολιτικών: Κατάλογοι

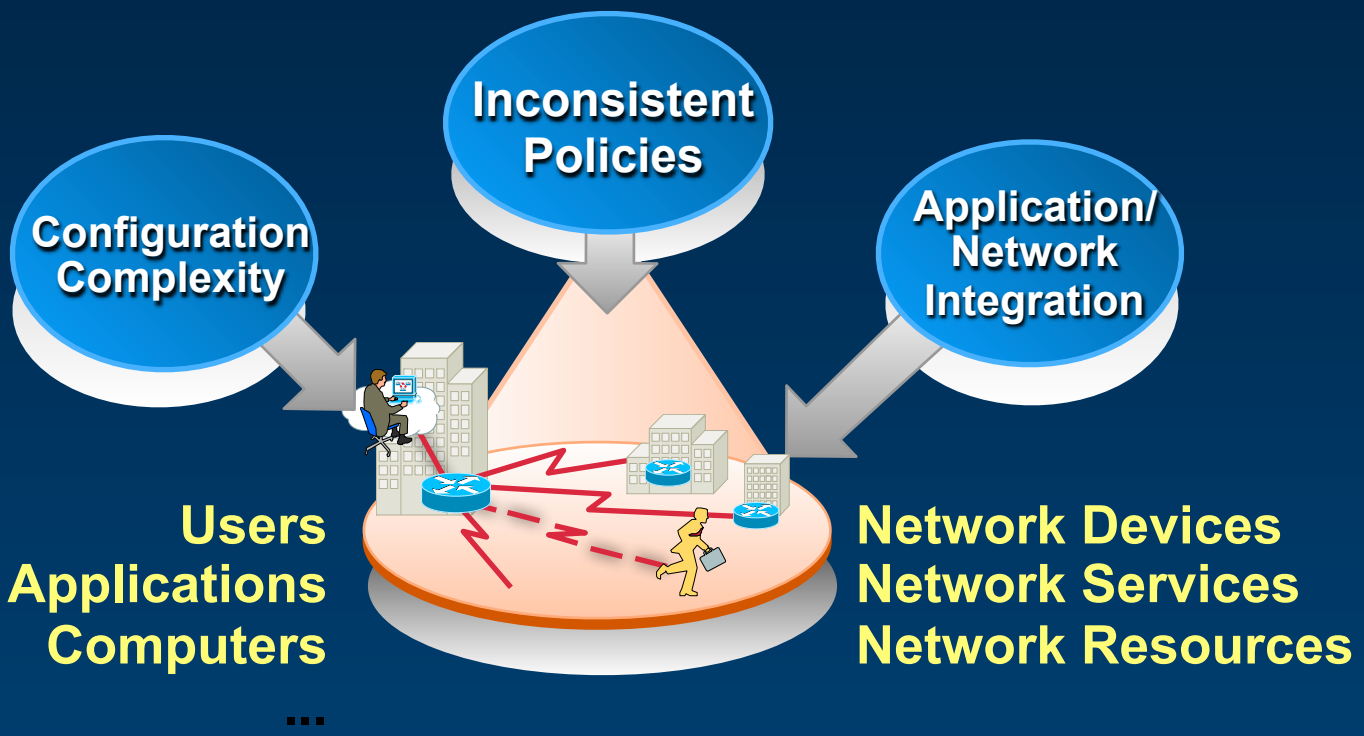
- Currently, each individual application stores policy data (e.g., configuration) in its own private data store:
  - These data include relevant information concerning network resources and services as well as clients wanting to use those resources (e.g., people, processes, and applications).
  - The same resource may be represented in several data stores. Even if a resource is modeled the same way in each data store, each application only has access to its own data. This leads to duplication of data and data synchronization problems.
- Alternatively: enable applications to store data describing the resources in a single directory using a common format and access protocol.
- Defining a logically centralized common repository, where resources and services are represented in a common way, enables applications of different types to utilize and share information about resources and services that they use.
- Directories are critical resource components that provide support to many other elements in the middleware environment, especially policy.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- Πολιτική (Policy)
- **Κατάλογοι (Directories)**
- Διαχείριση Πόρων (Resource Management)
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Ορισμός Καταλόγων

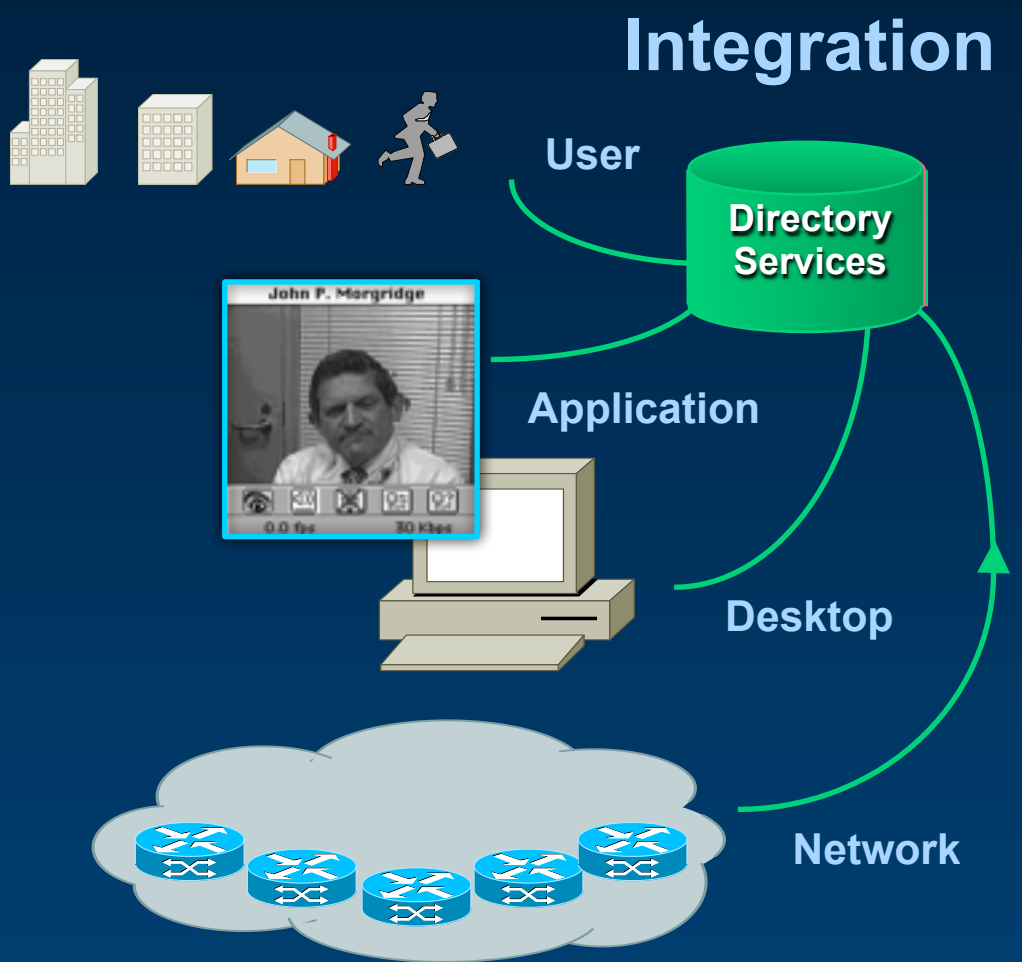
- The purpose of a directory is to store, identify and retrieve a specific set of resources and information about them in a computing environment.



Πλεονεκτήματα των Καταλόγων

- Avoid data duplication.
- Avoid synchronization problems.
- Provide inherent extensibility in describing the characteristics of an object:
  - A single entity can be represented by multiple directory objects, each representing a different aspect of the entity.
  - Different applications can be responsible for managing the different objects that together make up a higher-level object, even if applications themselves cannot communicate with each other.
- Directories enable applications to effectively share and reuse data:
  - In the short term, users and applications will benefit from having all of the data in one place.
  - In the long term, users and applications will be able to take advantage of data managed by other applications.

Ο Ρόλος των Καταλόγων



- Common information model
- User profiles, applications and network services
- Single user identity
- Integrated policies

Μοναδική Ταυτότητα Χρήστη

Who: Bob

Role: Finance Director

SLA: Finance Gold - Active

Services: Finance Applications—Yes
Service Level—Silver

Information: Data Warehouse—No

Access From: Anywhere in Intranet

Protect What: All Database Information

Χαρακτηριστικά Καταλόγων

- A directory associates attributes with objects.
- Directories differ from classic databases in four key aspects:
  - Directory objects are essentially independent of each other, whereas database objects are related to each other (sometimes in very complex ways).
  - Directories organize their information using the notion of containment, which is not naturally implemented in databases.
  - Directory objects can have specific access controls assigned to an object and even to attributes of an object.
  - Directories are optimized to perform a high number of reads vs. writes.
- Directories use a common core schema, supporting a common set of syntaxes and matching rules, that defines the characteristics of their data. This enables a common access protocol to be used to store and retrieve data.

Χρήσεις Καταλόγων

- Directories are used mostly to represent people, servers, printers, and other similar objects.
- Conceptual information models like CIM and DEN have encouraged the use of directories for containing common objects in a managed environment. For networked applications, this enables clients of the network to be bound to services available in the network in a transparent manner.
- The "Grid" community is making extensive use of directory services to maintain information about the structure and state of not only networks but also computers, storage systems, software, and people.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- Πολιτική (Policy)
- Κατάλογοι (Directories)
- **Διαχείριση Πόρων (Resource Management)**
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Διαχείριση Πόρων

- Need to determine:
  - How different types of resource managers learn and locate about each other.
  - How do they deal with cross-domain security issues.
  - A resource definition language that can describe the individual elements of the resource being utilized: network, processor, agent, memory or storage.
  - Appropriate metadata representation and underlying meta schema that can be applied to multiple resource types.
  - Languages, clients, and servers to support accessing various types of distributed computing resources.
- Broad interest in developing an automated access control architecture, using policies, to support the evolving IETF differentiated services architecture.

Διαχείριση Πόρων

- Many resource managers being deployed today rely on:
  - Directory services for storing policy information.
  - X.509 for certificate-based authentication and authorization to these resources.

Απαιτήσεις για Διαχείριση Πόρων

- Middleware will be required to translate the needs of distributed and parallel computing applications within and across different policy domains.
- It is crucial that a standard means for representing and using resource management be developed.
- Advance reservation of resources, as well as dynamic requests for resources, is a crucial aspect of any resource management system.
- It is important to address the issue of possible deadlock and/or the inefficient use of resources (i.e., the time period between a request, or set of requests, being initiated and honored and resources being allocated).
- There is also a need for rendezvous management in resource allocation services, where an application must gather resource reservations involving multiple sites and services.

Υποδομές Διαχείρισης Πόρων: Μοντέλα κ. Απαιτήσεις

Model for a resource management infrastructure:

- A mesh of cooperating resource managers, which interact with each other using standards-based protocols (e.g. COPS).
- Each resource manager may manage different sets of resources. E.g., one may be a bandwidth broker managing only network bandwidth, while another may be a general-purpose resource manager that manages security, IP address allocation, storage, processors, agents, etc.
- Middleware resource managers may not only allocate resources but also manage the composition of a group of services from security services, billing services, shaping of multimedia composite images, etc.).
- Resource managers depend on the use of locator services to find other resource managers and AAA server(s).
- Resource managers may need to query the network to determine if a policy request for bandwidth can be satisfied.
- It is essential that these (and other) different uses of resource management be integrated to provide an end-to-end service for applications and users alike.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- Πολιτική (Policy)
- Κατάλογοι (Directories)
- Διαχείριση Πόρων (Resource Management)
- **Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)**
- Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)

Ανακάλυψη και Ανάκτηση Πληροφοριών

- A wide range of middleware services related to discovery and retrieval of networked information. A broad range of applications requires these services.
- Most basic service in this area is persistent naming (URI's, URL's) and location services that resolve names to locations.
- Need for extensive infrastructure around resolvers:
  - How resources are assigned?
  - Ongoing management of data about current location of resources identified with some URI.
  - Operation of sets of resolvers for various name-spaces.

Ανακάλυψη και Ανάκτηση Πληροφοριών

- Other services revolve around metadata - descriptive, ratings, rights management, etc., that may be associated with objects on the network.
- The Resource Description Framework (RDF) from the W3C provides a syntax for attaching such descriptions to network objects and for encoding them; additional middleware work is needed to locate and retrieve metadata associated with objects that may be stored in repositories.
- Validation of metadata is a key issue requiring a complex set of trust relationships and hierarchies, and policies that will need to be specified for the use of these trust relationships in retrieval.

Ανακάλυψη και Ανάκτηση Πληροφοριών

- There is a need for middleware services which build upon work already integrated into services such as Archie and Harvest.
- These services permit the efficient extraction of metadata about the contents of network information objects and services without necessarily retrieving and inspecting those services.
- This includes the ability to dispatch "indexing agents" or "knowbots" that can run at a site to compute such indexing, under appropriate security and authentication constraints.
- A set of "push-based" broker services which aggregate, filter and collect metadata from multiple sites and provide them to interested applications are also required.
- Such services can provide a massive performance, quality, comprehensiveness and timeliness improvement for today's webcrawler-based indexing services.

Ενδεικτικά Δομοστοιχεία Μεσολογισμικού

- Διαπροσωπείες Εφαρμογών και Σηματοδότηση (APIs & Signalling)
- Ταυτοποίηση, Εξουσιοδότηση, Απογραφή (Authentication, Authorization, Accounting)
- Πολιτική (Policy)
- Κατάλογοι (Directories)
- Διαχείριση Πόρων (Resource Management)
- Ανακάλυψη και Ανάκτηση Πληροφοριών (Networked Information Discovery and Retrieval Services)
- **Δικτυακή Ποιότητα Υπηρεσιών (Network QoS)**

Δικτυακή Ποιότητα Υπηρεσιών

- Applications need to explicitly request resources available in the network to meet their requirements for certain types of communication, or in order to provide service with an appropriate guarantee for bandwidth, jitter, latency, loss, etc.
- Moving towards services beyond best effort, particularly with respect to services running over IP, through quality of service (QoS) and class of service (CoS) mechanisms. IETF models:
  - Integrated Services (IntServ): uses RSVP as signaling mechanism. Requires state in every router for every session. It is generally recognized to have scaling limits.
  - Differentiated Services (DiffServ), grew out of a reaction against the perceived scalability problems with the IETF IntServ model. DiffServ is an architecture for implementing scalable service differentiation in the Internet. Scalability is achieved by aggregating traffic through the use of IP-layer packet marking.
- The practical realization of either or both architectures depends on many middleware components. Both IntServ and DiffServ make sense only if linked to a policy mechanism. This mechanism must be able to make policy decisions, detect and resolve conflicts in policies, and enforce and monitor policies.

Σύνοψη και Συμπεράσματα

- Middleware infrastructures may have:
  - Components and services that exist in the persistent infrastructure.
  - Components outside the persistent infrastructure that enable and support end-to-end (i.e. application to application or host to host) interaction across multiple autonomous administrative domains.
- Core persistent middleware services:
  - Required to support the development of a richer set of middleware services which can be aggregated or upon which applications will be based.
  - Core middleware services will help applications leverage the services and capabilities of the underlying network infrastructure, along with enabling applications to adjust in changes to the network.
  - Basic and core middleware services include, but are not limited to: directories, name/address resolution services, security services (i.e., authentication, authorization, accounting, and access control), network management, network monitoring, time servers, and accounting.

Σύνοψη και Συμπεράσματα

- Network level capabilities, such as multicast and DiffServ, are not classified as middleware; rather, they are enabling infrastructure services upon which middleware will be built or which middleware may use and manage.
- A second level of important middleware services, which builds upon core set of services, include accounting/billing, resource managers, single sign-on services, globally unique names, metadata servers, and locators.
- We are seeking middleware services that enable access to and management of the underlying network infrastructure and support applications wishing to make use of that network-based infrastructure. It appears necessary to agree to a framework of services for the support, provisioning and operations, and management of the network.
- Efforts require the integration and management of many infrastructure components, not just networks; however, so far there is no overall framework that pulls all of these together, or a mechanism to coordinate all of these activities.

Σύνοψη και Συμπεράσματα

- Inter-domain resource management architecture and protocols (e.g., inter-domain bandwidth brokers).
- Resource languages that describe and enable the management of a wide variety of resources (e.g., networks, data bases, storage, online facilities, etc.
- Avoiding deadlock and ensuring efficiency with resource managers.
- Network management tools and APIs that provide macroscopic and microscopic real-time infrastructure
- Information to middleware services and applications (not just MIBs and SNMP access).

Σύνοψη και Συμπεράσματα

- Domain and inter-domain accounting and billing.
- Monitoring and verification services of contracted infrastructure services.
- Enhanced locators that can locate resources and resource managers cross administrative policy negotiation and authentication.
- Middleware bypass (i.e. access to raw system or network resources metadata (i.e., data that is used to describe data found in directories or exchanged between services such as resource managers, PDPs, PEPs, directories, accounting and billing services, etc.).
- Middleware support for mobile or nomadic use.
- Support for availability of resources (i.e. replication and load balancing).