Networking of Information And Cloud Computing

A seminar report
submitted in partial fulfilment of
the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering
(University of Calicut)

by

Kiran V

Department of Computer Science & Engineering

MES College of Engineering, Kuttippuram

(ISO 9001-2000 Certified Institution, Affiliated to University of Calicut)
Thrikkanapuram PO, Malappuram Dt, Kerala - 679573

2011-12

Certificate

*This is to certify that the seminar report entitled "***Networking of Information And Cloud Computing***" is a bonafide record of the work done by* **Kiran V** *(Roll no: 08103026) under our supervision and guidance. The report has been submitted to the* **Department of Computer Science and Engineering** *of* **MES College of Engineering** *in partial fulfilment of the award of the Degree of* **Bachelor of Technology in Computer Science and Engineering.**

Dr. P.P. Abdul Haleem
Professor and Head
Dept.of Computer Science and Engineering
MES College of Engineering

Mr. Aneesh M Haneef
Seminar Guide
Assistant Professor
Dept.of Computer Science
and Engineering
MES College of Engineering

Abstract

Cloud computing is making it possible to separate the process of building an infrastructure for service provisioning from the business of providing end user services. Today, such infrastructures are normally provided in large data centres and the applications are executed remotely from the users. One reason for this is that cloud computing requires a reasonably stable infrastructure and networking environment, largely due to management reasons. Networking of Information (NetInf) is an information centric networking paradigm that can support cloud computing by providing new possibilities for network transport and storage. It offers direct access to information objects through a simple API, independent of their location in the network. This abstraction can hide much of the complexity of storage and network transport systems that cloud computing today has to deal with. This paper analyze how cloud computing and NetInf can be combined to make cloud computing infrastructures easier to manage, and potentially enable deployment in smaller and more dynamic networking environments. NetInf should thus be understood as an enhancement to the infrastructure for cloud computing rather than a change to cloud computing technology as such. To illustrate the approach taken by NetInf, this paper also describe how it can be implemented by introducing a specific name resolution and routing mechanism.

Acknowledgements

I wish to record my indebtedness and thankfulness to all who helped me in preparing this report seminar report titled **Networking of Information And Cloud Computing** and present it in a satisfactory way. This report is part of my work related to the paper **CS04 707(P) Seminar**.

I am especially thankful to **Dr.Abdul Haleem P P** the Head of **Computer Science & Engineering** for his guidance and encouragement throughout the seminar. Also my guide and supervisor **Mr. Aneesh M Haneef** Asst. Professor in the Department of **Computer Science & Engineering** for giving me valuable suggestions and critical inputs in the preparation of this report. I am thankful to the seminar coordinators **Mr.Arun Anoop M & Ms. Jabin A & Aneesh M Haneef** Asst. Professors in the Department of **Computer Science & Engineering** for their support throughout the seminar.

My friends in my class have always been helpful and I am grateful to them for patiently listening to my presentation.

Kiran V

Contents

Tables

Table

Figures

Figure

Chapter 1

Introduction

The topic of Cloud Computing is gaining more and more attention in the service research community. The main idea is to make applications available on flexible execution environments primarily located in the Internet. Several flavours are known, and three important ones are depicted in the Figure 1.1 .

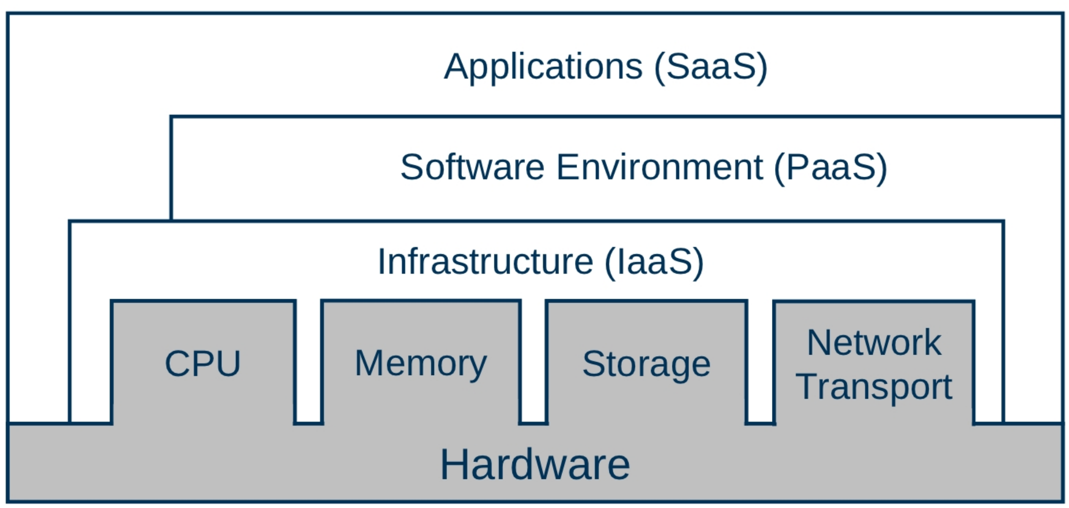

Figure 1.1: Cloud Computing Overview

Infrastructure as a service refers to the sharing of hardware resources for executing services, typically using virtualization technology. With this so-called Infrastructure as a Service (IaaS) approach, potentially multiple users use existing resources. The resources can easily be scaled up when demand increases, and are typically charged for on a per-pay-use basis. In the Platform as a Service (PaaS) approach, the offering also includes a software execution environment, such as an application server. In the Software as a Service approach (SaaS), complete applications are hosted on the Internet so that e.g. your word processing software isnt installed locally on your PC anymore but runs on a server in the network and is accessed through a web browser.

At the same time, the networking research community is working on exploring the benefits arising from the new paradigm of content-centric or information-centric networking. Traditional networking architectures for the PSTN and the Internet solve the problem of connecting terminals or hosts in order to support a specific application such as telephony or WWW. To this end, traditional naming and addressing schemes employ box- or domain-oriented identities such as E.164 numbers for telephony, or IP addresses and URLs for the Internet. However, the end user is typically interested in reaching a destination object that sits behind or in the host, such as a human being or a file, rather than communicating with the host itself. As the destination objects move to new hosts, the host- or network-dependent identities of these objects must be updated. Information-centric networking provides a solution to these issues by directly addressing the information objects instead of using the host-dependent or domain-dependent addressing schemes.

While URLs are also used to identify information objects, there is an important difference to how NetInf names information objects. URLs include the domain name or locator of the host at which the target object is stored and are therefore based on the traditional location-oriented communication paradigm. Consequently, links based on such URLs break when the host of a target object moves to a new location, or when the address of the host of the object changes. In the current Internet, there are several fixes to circumvent this problem, such as HTTP redirects and dynamic DNS. The location-independent object naming scheme of NetInf avoids this problem altogether, as the NetInf object names remain the same independent of any topology events, including location updates.

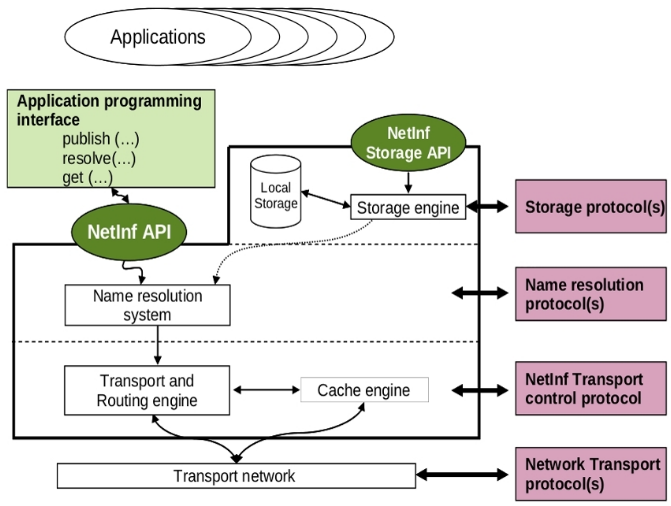

Figure 1.2: Networking of Information Overview

The basic idea of Network of Information (NetInf) is to move from todays host centric networking paradigm to an information centric networking paradigm where the information objects are the primary components of networking. One key aspect is that the information objects are named independent of the hosts they are stored on. In NetInf,users request the information through a well-defined API by specifying the name of an information object using the get() primitive at the NetInf API in Figure 1.2. The name can have cryptographic properties, e.g. a hash of the content of a file can be part of the name. The name of the information object can then be used to verify the authenticity of the file. How and from where the object is retrieved is decided by the NetInf Transport and Routing engine. This makes it possible to react to changes in the network, both in terms of topology and load situation, in a flexible way. Information can be stored at arbitrary caches in the network (for short-term optimization, cf. Caching engine in Figure 1.2) and storage units (for long-term persistent storage, cf. Storage engine in Figure 1.2). It can also be retrieved from NetInf hosts that have already received the information and have stored it in their local cache. The NetInf Name Resolution System constitutes a flexible and scalable mechanism for handling the bindings between the object names and location, e.g. to support host, user and object mobility. From a cloud computing point of view, the network of information therefore offers new technology for dealing with the networking resources and storage boxes in Figure 1.1.

Chapter 2

How Cloud Computing Can Benefit From NETINF

Cloud computing is today offering an efficient environment for quick deployment of new services. In particular, it offers unrivalled opportunities for quickly scaling up the capacity for services that suddenly become popular. Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) approaches have proven to be effective means for separating the service deployment from the provisioning of the required service infrastructure. The latter obviously is a much slower process as it includes hardware installation and possibly extended physical facilities, as well as establishment of new communication infrastructure. One of the major hurdles with current cloud computing is the management costs of the computing, storage and networking facilities. To make the management tractable, todays approach to cloud computing depends on the site infrastructure and the networking topology being relatively stable. The main benefit of what is presented in this paper is not a way to improve the cloud computing functionality as such by use of NetInf. It rather argue that by basing the cloud computing infrastructure on NetInf, many of todays issues with management of cloud computing infrastructures can be eased. In this light, distributing cloud computing service objects across a more dynamic networking environment like moving networks, or a home networking environment, is a very difficult task with todays cloud computing technology. In addition, cloud computing service objects need to be copied between servers, e.g. to realize load sharing. When using existing technologies, such issues would need to be handled by a combination of more or less automated network management tools, routing protocols, and add-on protocols for host and site multihoming, as well as host and network mobility. This adds substantial complexity to the system, and the mechanisms may interact with each other as well as with NATs and firewalls in an unpredictable fashion.

The following paragraphs illustrate how introducing NetInf technology can help addressing the above cloud computing challenges.

The Network of Information paradigm allows for a new approach to the problem of dynamic network topologies. While cloud computing brings about revolutionary technology to resource sharing, the Network of Information is a new approach to accessing information (in its widest sense) on a network. It brings a new abstraction to the networking layer: the notion of information objects. Compared to existing networking paradigms, this new approach is designed for directly accessing the information, rather than addressing it indirectly via the host or network domain containing the information. The location of the information thereby becomes secondary, which makes it much easier to deal with configuration changes and mobility. The addressing used in NetInf fully relies on naming the information itself, and not the location or network domain it is retrieved from. For a cloud computing platform, this significantly simplifies the way data is handled. This property also allows for hiding reconfigurations of the supporting nodes and networks from the cloud computing platform. It could potentially also make cloud computing deployable on a smaller scale, like home networking, and even in environments with mobile nodes and networks.

When the access of information objects is no longer done in an application-specific fashion, integration and composition of applications become easier, since they all access information using the same naming scheme.

The media distribution capabilities of NetInf provide a part of implementing

IaaS. A part of IaaS or PaaS offerings is often the capability to efficiently distribute large amounts of content, e.g. video files. This can be realized via content distribution overlays, such as the well-known Akamai CDN. Content distribution functionality is provided natively in a NetInf-enabled network, realized by the functions introduced in the previous section. Information is simply requested by specifying its name, while a NetInf-enhanced networking layer takes care of choosing the optimal distribution mechanism, including caching and source selection. Similar to the popular peer-to-peer systems, clients can also serve as a new source for already downloaded information. NetInf functionality will also support services and service composition. Without aiming to become a service platform in itself, the idea is that services can be treated as a type of information objects. Also a service needs to be unambiguously identified, and can have multiple instances in the network. Meta-data can be used to describe the characteristics of the service, and to support the proper selection of a service. This applies both to cases where the selection takes place manually, e.g. by a user browsing a service registry, and to cases where this happens in an automated fashion, e.g. in a service composition engine. In this way many mechanisms such as load balancing and mobility that are available through NetInf can be reused also for services.

Through a close cooperation of cloud computing and networking of information, network nodes and network resources, including various storage systems, have the potential of becoming truly transparent to the applications and the users. The natural evolution of networking is thus to move from networking of nodes to networking of information objects. But for users to feel comfortable with the, initially appealing, idea of just dropping their information objects into the network, for storage, processing and distribution, there are a number of issues that need to be addressed and are currently being worked on in the 4WARD project. These include security issues like confidentiality, integrity, privacy and access rights. Also requirements on reliability and availability will pose challenges. While these all are first rank challenges, the major hurdle on the way to networking of information lies in the scalability issues associated with changing the granularity of networking from, relatively, few nodes to the extremely plentiful information objects themselves.

Chapter 3

Scenarios

This section briefly discuss some of the scenarios : A content distribution example, an Augmented Internet and a personal mobile scenario. They were chosen to represent different perspectives of the possibilities of the NetInf approach. Each scenario is discussed with the problem it is addressing, some of the requirements it puts on NetInf as well as the advantage such a scenario will obtain when combined with NetInf.

3.1 Content distribution

Contemporary content distribution has effectively become an assortment of individual and specialized solutions. Depending on the field of application, these solutions range from carefully engineered and managed Content Distribution Networks owned by individual corpo- rations to more ad-hoc and unmanaged peer-to-peer networks such as BitTorrent based file exchange overlays . In addition, a number of technologies, such as content personal- ization and adaptation have emerged. Although these technologies seem quite different in the way they are approaching the problem, effectively they are in fact all dealing with getting content from information producers to information consumers. It is, therefore, natural to investigate how to employ Networks of Information in the field of content distribution.

Here envisage a solution where NetInf provides a uniform mechanism for accessing content, seamlessly incorporating different content types and distribution methods. These include novel schemes that either blend existing approaches or define new ones. A key enabler to this approach is the notion of information objects, a NetInf architecture cornerstone. In this archi- tecture, the particular representation of information (e.g. a MP3 file) and how it is retrieved (e.g. HTTP or BitTorrent) are orthogonal to the information object itself.

Using the notion of Information Objects, there is a method that can accommodate different kinds of distribution schemes and network scenarios. This means that all that is needed is the identifier of the information object to be retrieved. Given this identifier, the NetInf infrastructure will then decide what the optimal source or sources are, and deliver the content to the user. The source in this context is not defined until the retrieval begins. Possible distribution schemes include simple point-to-point transfers and live streaming as well as more sophisticated P2P methods. The choice may well be determined on whether the object is to be shared, and by how many potential recipients. The key differentiator to todays solutions is that that the whole range of possible distribution mechanisms can be exploited for a given transfer. Especially in the light of the trend towards high definition video that leads to constantly in- creasing data volumes, this enables the networks to choose source and distribution mode in an optimized manner.

But also the classical web browsing can benefit from NetInf technology. Today, many of the distributed data storage technologies (particularly P2P systems) are not applicable to web browsing. The setup times associated to a file transfer (e.g. finding and connecting to the peers) can account for a substantial fraction of the download time, often tens of seconds. For large downloads, this set-up delay is acceptable, as it only represents a small fraction of the overall time spent performing the download, however web usage typically requires many smaller transmissions. Such set-up latencies would mean that the average web page would take several minutes to load, clearly out of most users expectations. As a result of these latencies, web-browsing maintains a strictly client-server model, with the consequences that resources are frequently concentrated in one topological region, meaning that availability of these resources are threatened by any agent capable of disabling or destroying a single server (peer-to-peer systems are typically able to offer some degree of resilience in this regard). This envision a Serverless Web which attempts to create a peer-to-peer style system with latency characteristics which would be considered acceptable for web browsing. The goal, therefore, is to attempt to develop in NetInf a decentralised method of distributing data, while minimising the associated coordination latencies.

As further aspects of the work in NetInf, planning is there to explore the benefits of including network services in the distribution chain, such as different types of content adaptation. Transcoding, for example, has already been used in the existing Internet where properties such as the resolution or the bit rate of an information object are changed in real time. Content personalization, and the required content adaptation, is also increasing in importance. The actual media processing may also occur external to NetInf, with NetInf selecting the appropri- ate transport mechanism when available. The cryptographic properties of information objects can be exploited in order to control access to the information. This embraces the well-known Digital Rights Management (DRM) scheme already known today to e.g. only grant access to a media file when a fee has been paid. In addition, the protection of user-generated content is an important issue to be solved.

3.2 Augmented internet

As mobile Internet-enabled devices become more ubiquitous, users will want to use them to access greater amounts of information and services in the real world. This includes objects close to the users such as everyday objects, people they meet, or places they visit. When accessing information on the move, it is essential that the accessing information does not detract the user's attention from the real-world activities. Unfortunately, mobile Internet access as experiencing it today requires a lot of attention by the user and is therefore not suitable for many scenarios. To support such scenarios, a smooth integration with the real world that enables service access without interrupting the user's real-world work flow is needed by Internet applications. However, such applications are difficult to build on a large scale due to the current Internet architecture. Basically it does not provide a notion for real-world integra- tion.

The concept of network support for real-world-integrated applications the Augmented Internet paradigm, for example, a tourist near the Eiffel Tower cares about opening hours, ticket cost, the history of the monument, and so on. Whether this information is located on a server in Paris or elsewhere is irrelevant to the user. URIs such as www.tour-eiffel.fr provide an abstraction layer to the location of information, but nevertheless tie it to specific network nodes. Real-world integrated applications are inherently information-centric and that NetInf is well suited to provide an architecturally sound infrastructure to enable and support real-world-integrated applications. Such applications pose two main requirements on an underlying infrastructure. First, the Augmented Internet needs a notion of virtual represen- tations for physical entities that can cumulate and provide access to physical-entity-related services. Second, the Augmented Internet has to build and maintain a binding between the physical entity and its virtual representation on the future Internet. The NetInf architecture addresses these requirements by providing an API that is common for objects that represent real-world entities as well as objects that represent services, content, and other digital entities. Based on the common API and the information model, bindings and interactions can be defined between the objects representing real-world and digital entities to enable Augmented Internet applications. The Augmented Internet paradigm combines multiple families of use cases that are described next. One family combines use cases that enable the interaction with physical entities via entity-related Internet services directly from the real world. This facilitates a completely new real-world interface to access physical-entity-related services without the need for a conventional Internet search. A tourist could, for example, obtain information about buildings and places by simply pointing with his cellular phone at a building instead of performing a cumber- some full-text search. Any output interface like an audio headset or the cell phone display could be used to unobtrusively present the resulting information as illustrated in Figure 2. In this use case, identifying the monument can technically be realized by using physical attrib- utes of the building such as its estimated GPS position, via the users position and an elec- tronic compass. These attributes will be used by the Augmented Internet infrastructure to build and maintain a connection between entity-related information and the physical entity itself.

Similarly an Augmented Internet service could identify objects based on, for example, Radio Frequency Identification (RFID) tags and could execute Internet services related with the referenced object. For example, a user could "click on" a library book, thereby executing the online service to renew the book. Furthermore, a selection menu could be displayed to the user, providing additional options like accessing the bibliographic information of the book. All book-related information are cumulated in the book's virtual representation, provided and maintained by the information-centric network infrastructure. A simple Augmented Internet service

Figure 3.1: Tourist application: Accessing information about buildings

could also enable a user to store personal notes in this virtual representation, hence directly binding them to the book instead of storing the notes in some separated text document that can be difficult to retrieve.

Furthermore, an Augmented Internet could enable people to use web methods such as bookmarking in the real world: a user could bookmark interesting products by clicking on the physical object with a cell phone. Likewise, a user could add a virtual post-it to a specific object by the same method.

3.3 Personal Mobile Scenario

A person on the move is a common scenario for many projects investigating new ideas for improving communication in a mobile and wireless environment. Past projects have looked at composing networks, seamless integration of multiple radio technologies and various other multi-access mechanisms. Most solutions however cannot handle frequent disconnec- tions because they rely on that there always is enough connectivity to keep an end-to-end TCP connection alive.

In the NetInf personal mobile scenario, disruptions and disconnections for shorter or longer times are important characteristics making it different from previous projects. The argument is that it is impossible to provide good wireless service everywhere and therefore also in the future often will experience disruptions when communicating on the move. The purpose with the scenario is to show that the NetInf approach can provide better service under these conditions compared to current technology by circumventing the need for end-to-end connec- tions and making use of intermediate storage, much in the same way as Delay Tolerant Networking (DTN) does. The extreme case is when there never is connectivity simultaneously along the end-to-end path.

In the personal mobile scenario, a person is using a laptop to read and send email while travelling on a train as illustrated in Figure 3.2. The train passes stations now and then where local hot-spots with high-speed wireless communication are available. The hot-spots also have storage that the NetInf functionality can use temporarily. In addition to the hot-spots, cellular service (3G) is available with varying quality during the journey. The varying quality can have a severe impact on the application performance experienced by the user.

On the down-link to the train NetInf can make full use of the hot-spot capacity by prefetching desired information before the train arrives, and on the up-link by quickly offloading informa- tion to the storage and later relay to the final destination. The goal for NetInf in this scenario is to make the best use of both the cellular and the hot-spot capacity.

The scenario proceeds as follows:

- The user initiates email retrieval from the mailbox at A over the cellular radio link.
- The system concludes that retrieval will take a long time due to insufficient capacity, possibly caused by disruptions.
- The system determines that the next available hot-spot is at station B.

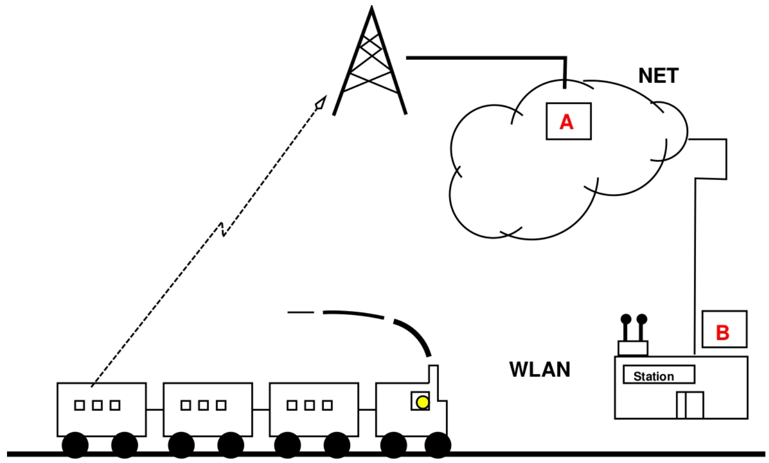

Figure 3.2: Personal Mobile Scenario

- A is instructed to send the mailbox to the storage at B.
- The train arrives at B and the user can quickly receive the remaining part of the mailbox.

An alternative is that the user communicates directly with B through the cellular link and sends a pre-fetch message to B. B would then pull the data from A on the users behalf. To achieve this, the network must have the ability to locate the closest copy of an object and be able to move the data closer to where the user will be soon. This replication may either be triggered by some (low-bandwidth) control message or automatically by the network. Also, either the network or the client needs a way to predict which stations would be suitable to replicate data to. Such network primitives would provide benefits to the current synchronous Internet model, by providing an asynchronous model of communication to mobility aware applications. This would also allow decoupling of the wanted information from the originating nodes. It would finally provide bandwidth adaptation and improve usage enabling the use of transient access.

Chapter 4

Architecture Overview For Cloud Computing With NETINF

In Figure 1.1 it is shown how the basic cloud computing service IaaS is composed of a set of virtualized resources, namely CPU, memory, storage and network transport. In the previous section, it is already explained how NetInf can offer an enhanced solution for network transport and storage. But lets take one step back and look at the two main reasons to virtualize resources: one is security - to restrict access to resources in order to offer a virtual private environment, the other is resource separation in order to avoid resource conflicts.

NetInf can offer an alternative to the security aspect of virtualization for two of the cloud computing resources, storage and network transport. The NetInf networking architecture inherently secures the access to information objects and network resources by cryptographic means. There is therefore no need to virtualize storage and network transport resources to ensure the access rights. This stems from the fact that NetInf abstracts away the boxes and links that are implementing these resources and secures the resources themselves. Thus, there is no need to virtualize these boxes and links from a security perspective.

The need to virtualize for resource separation may or may not be a problem depending on the network configuration. With the current trend of reduced prices, especially on storage but also on transmission links, this might be dealt with by traditional overprovisioning, which can be combined with active network management in order to ensure that new (real) resources are added before resource conflicts appear. NetInf in combination with advanced virtualization techniques like Vnet proposed in 4WARD can provide a simpler solution to this problem by virtualizing both storage and network transport resources. One key advantage with Vnet compared to other virtualization techniques is that it also addresses the problem of virtualizing wireless network resources.

The NetInf architecture provides an API for communication between arbitrary types of information objects, independently of which hosts they are attached to, and of how they move between hosts. Arbitrary information object means any information object which adheres to the NetInf naming scheme, and which is registered with the NetInf name resolution system. Examples of such objects are data files, service objects, or digital representations of physical objects, such as RFID tags.

The NetInf API supports a mode of communication which is object-centric in the sense that only the object identity or a set of attributes are needed to access an object. The object naming scheme allows users and applications to create object names based on cryptographic hashes of the owners public key. This avoids the need for introducing a new naming authority. Such object names are entirely independent of the location of the object. Using this naming scheme, the API hides the location of an object, as well as the dynamics of the underlying transport network. In addition, if several identical copies of an object exist in the network, the NetInf name resolution and routing system finds the "best" copy in an anycast fashion.

The API includes methods such as publish(objectName), resolve(objectName), and get(ObjectName). These methods allow for registrations of objects in the name resolutions system, resolving the location of an object, and establishing connectivity with the object. The API also includes methods for object storage and retrieval.

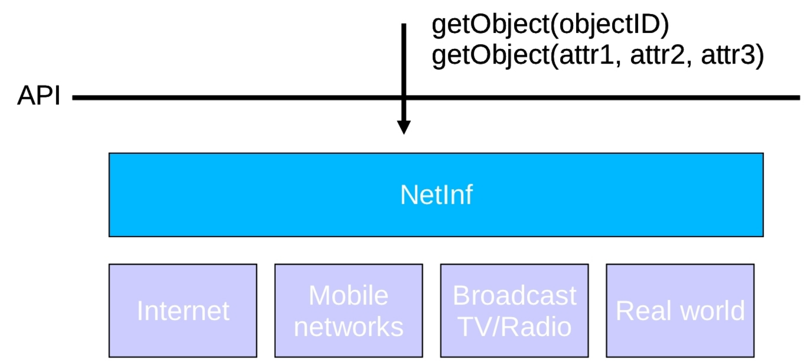

Figure 4.1: NETINF API

Figure 4.2 shows how NetInf can support a set of different cloud computing services. The cloud computing service uses NetInf objects, which communicate over a dynamic infrastructure network, such as a global network, using the NetInf API. The service uses object names to retrieve and interact with other objects over the API, and has no notion of object locators. The dynamic infrastructure network on the other hand uses traditional addresses (locators), and has no notion of service layer objects.

A cornerstone of the NetInf architecture is the name resolution and routing system, which resolves the name of an object to a set of current network locators. The resolution mechanism is designed to handle highly dynamic network topologies in a scalable fashion, and provides an updated ocator for a digital object that is moved between hosts, or for digital objects that are stored on mobile hosts, which in turn may be attached to moving networks. Also, the resolution mechanism is capable of handling multihoming of objects, hosts, and networks. Note that the capability of handling these rather general mobility and multihoming scenarios also provides a basis for the handling of dynamic events in fixed networks as described earlier.

The name resolution and routing system is designed to scale to large networks and to a large number of objects. Likewise, the routing system must allow for short convergence times also in a dynamic network topology. The focus of the next section is on the name resolution and routing system and its interoperation with the dynamic network infrastructure. A novel mechanism is described that allows for a strict separation between the object-centric view of the API on the one hand, and a highly dynamic network topology on the other hand.

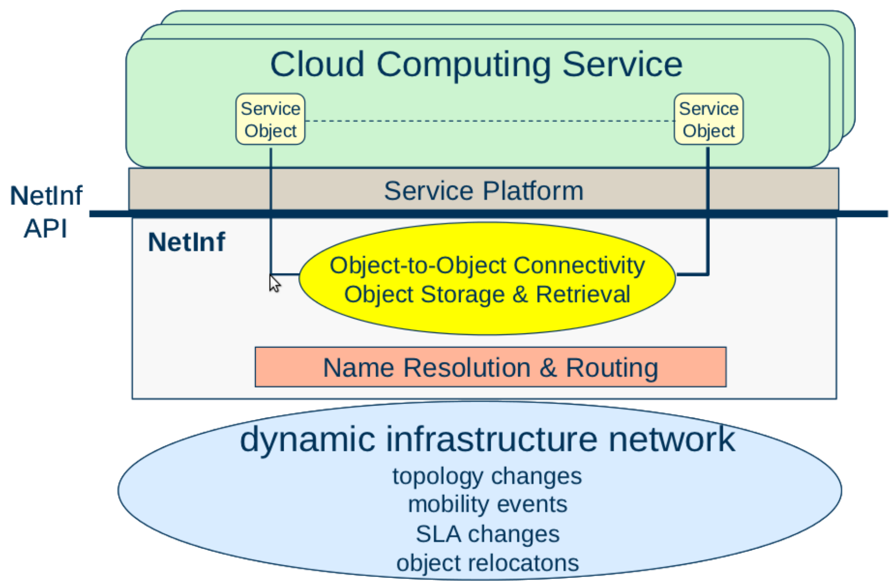

Figure 4.2: Architecture Overview For The Support Of Cloud Computing

Chapter 5

Conclusion and Future Work

The main purpose of this paper is to investigate what benefits the Networks of Information (NetInf) technology can bring to the infrastructure that is the foundation for cloud computing. In particular, the management of a cloud computing infrastructure can be simplified, as it does not have to deal with the details of storing and transporting information objects.

This paper presents the NetInf architecture, and described how it can support cloud computing services by offering an API that hides the dynamics of object locations and network topologies. One single name resolution and routing mechanism is used, regardless of whether the dynamics depend on network reconfigurations, change of service level agreements, mobility events, re- homing events, or any other type of network event. The task of designing cloud computing services that are robust against object re-locations or changes in the topology of the underlying infrastructure network can thereby be significantly simplified.

To illustrate the NetInf approach, a novel routing mechanism based on late locator construction has been described that performs object-to-object routing rather than traditional host-to-host routing. This mechanism can operate over a highly dynamic network topology and allows for scalable handling of a very large number of objects.

Future work includes more detailed investigations on how NetInf can handle services, including the use of NetInf as a service directory for Web Services. Apart from the features described in this paper, also automated and distributed processing of information objects are being investigated, e.g. to offer a delay-sensitive service as close as possible to the end-user. As both NetInf and virtualization (Vnet) of network resources are part of a common architecture being developed in the 4WARD project, studies are being conducted on which additional benefits their combination can bring to cloud computing.

Bibliography

[1] Bengt Ahlgren, Matteo D'Ambrosio, Christian Dannewitz, Marco Marchisio, Ian Marsh, Brje Ohlman, Kostas Pentikousis, ReneRembarz, Ove Strandberg, and Vinicio Vercellone. Design Considerations for a Network of Information (position paper). In ReArch'08 - Re-Architecting the Internet, Madrid, Spain, December 9, In conjunction with ACM CoNEXT 2008.

[2] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. IETF RFC 3963, Network Mobility (nemo) Basic Support Protocol, January 2005.

[3] Rao Mikkilineni. Cloud Computing and the lLessons from the Past. 2009.

[4] Borje Ohlman, et al. First Netinf Architecture Description, fp7-ict- 2007-1-216041-4ward / D6.1. Technical report, January 2009.

[5] Kostas Pentikousis. Distributed Information Object Resolution. In Proc. Eighth International Conference on Networks (ICN), Gosier, Guadeloupe/France, March 2009. IEEE Computer Society Press.

[6] EU FP7 4WARD project, http://www.4ward-project.eu/