Content-Centric Routing for the Autonomic Networks

Yosra Barouni Promethee Spathis

UPMC, Paris Universitas, University of Paris 6
104 Avenue du prsident kennedy
Paris, 75016, France
Emails : {name.lastname}@lip6.fr

ABSTRACT

In this paper, we address the problem of content routing in the context of the Autonomic Network Architecture (ANA) project [1]. The general aim of the ANA project is to provide an innovative communication framework that helps to surpass the current IP limitations in an heterogeneous networking environment and provide autonomicity to the management of these environments. A flexible and dynamic framework that allows the integration of newly designed and adapted communication protocols is already proposed and its prototyping phase is ongoing. Many efforts are deployed in order to demonstrate its feasibility and its benefits [2]. As the content distribution has been shown as one of the most popular services of the current networks, we chosed to study the feasibility of content routing using the ANA framework.

Specifically, we use a filtering mechanism to forward client requests for a content towards the most relevant content registered in the network. We use also an optimisation process in order to determinate a limited set of neighbours to which the received requests has to be forwarded in order to find the matching content. Our protocol is designed in a modular way and its functionalities are decomposed in small blocks that can be configured and tested separately.

I. INTRODUCTION

Routing is the basic service of a network and any other service resides on this fundamental functionality. Hence, routing needs to be adaptable to diverse environments and usages required by users and applications [9]. First developments in routing were the wired protocols (e.g., RIP, OSPF). As soon as the wireless and mobile networks appeared, these developments were adapted to the new environment giving birth to protocols such as OLSR, DRS, AODV, ZRP and many others. All these protocols were centered on network addresses. As many statistics showed that the main usage of these networks is the content access and distribution, new approaches based on content search and location were proposed to resolve this issue such as the peer to peer overlay networks or the Content Distribution Networks (CDN).

As the networks are evolving to a self-managed and autonomic infrastructures, the majority of the existing routing and content distribution protocols showed their limitations when they came to be applied in these new generation networks. That's why the European project ANA was created, in order to focus on the design of new architectures and protocols to provide a coherent and self-managed functioning of the new generation networks and to adapt the existing protocols to the autonomic context. As we already said, because of the importance of the content access and distribution, we will focus on this issue in our paper.

The goal of the ANA project [3] is to explore novel ways of organising and using networks beyond legacy Internet technology and to demonstrate the feasibility of autonomic networking. For this aim, a new communication framework was proposed inside the project to allow the design of new communications protocols that are adapted to the evolutions and needs of the autonomic networks.

The proposed ANA framework has many interesting properties for us. First, it provides a flexibility we'll

---

[1] Autonomic Network Architecture (ANA), European project number : FP6-IST-27489

[2] see papers [7] and [8]

[3] http://www.ana-project.org/

AUTONOMICS 2008, September 23-25, Turin, Italy

DOI 10.4108/ICST.AUTONOMICS2008.4711

detail later and which allows the adaptation of the communication protocols to the dynamicity and the needs of the autonomic networks. Second, as we will show in the next section, the ANA framework was designed to be IP independent so that is becomes possible to integrate new protocols and paradigms for which the current Internet architecture is not well suited.

In this paper, we propose to exploit the ANA framework in order to design a content routing protocol that demonstrates the feasibility of such protocols in the autonomic networks and that confirms the benefits of such a flexible framework. Our protocol, designed in a modular way, is based on a filtering process that allows to the ANA nodes to forward the content requests received from other neighbours or clients towards the most adequate destinations independently from their addresses or physical locations.

The independence of the networking addresses is provided through the use of the only descriptions of content requests and published contents in the network in order to route the requests to an optimised destination matching them. Our content routing protocol is still dependant from the content naming scheme but at least, we provide and IP independence that allows its deployment in heterogeneous networks.

Finally, we describe the ongoing implementation of our content routing protocol using the ANA framework already developed in the project.

II. IMPLEMENTATION OF A CONTENT ROUTING PROTOCOL INSIDE ANA

In the first part of this implementation section, we describe the architecture proposed by the ANA project in order to redesign the autonomic networks, whereas in the second part, we present our content routing protocol using the basic ANA paradigms and abstractions and discuss the interfaces between its main components. Finally, we shortly discuss a validation scenario that we are developing currently.

A. Presentation of the ANA Framework

This section, we introduce the network framework proposed by the ANA project [5].

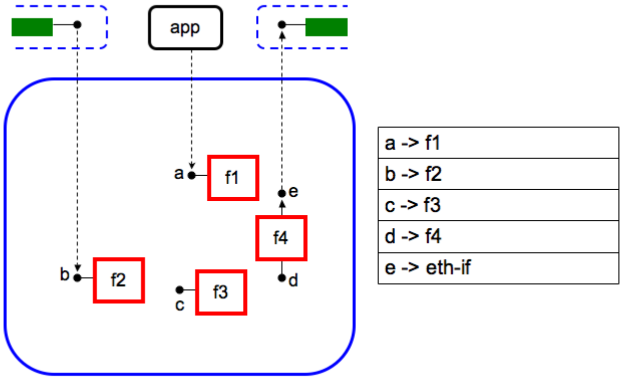
Actually, the ANA framework separates the network architecture into two main components which are the MINMEX and the ANA playground. The minmex (Minimal INfrastructure for Maximal EXtensibility) provides the basic low level functionality which is required to bootstrap and run ANA on nodes. The ANA playground is the execution environment that hosts more complex networking functionality. Our content routing protocol will be then placed inside the playground element.

Inside this playground, three essentials elements are defined : the Functional Blocks (FB), the Information Channel (IC) and the Information dispatch points (IDP). The FBs are the information processing units in ANA. They are the elements that represent the protocols of any other functionality implemented in the network. FBs can be composed of many 'bricks' which are the smallest possible elements providing some functionality in the ANA playground.

The ICs represent an abstraction for communication channels that allow nodes and FBs to communicate between each others.

Finally, the IDPs are the access points that address the previously defined elements in the playground like FB, bricks or IC. Each elements is bound to an IDP and can see its binding changed dynamically if needed. The concept of binding between the IDPs and the other playground components is very important to the ANA framework. In fact, it was designed to allow to dynamically compose and re-compose the bindings to protocols FB for example. As data is not sent directly between FBs using their own addresses as done in the classical Internet architecture but to the IDP associated with it, what is behind the IDPs is totally transparent and can be dynamically changed by the ANA framework.

The organisation of an ANA node is illustrated in the figure 1 which belongs originally to the ANA deliverable [5].



a -> f1
b -> f2
c -> f3
d -> f4
e -> eth-if

Fig. 1. Organisation of an ANA node

FBs are represented by red boxes and labelled f1, f2, f3 and f4. Each FB can be bound to one or more IDP labelled a, e, b, c, d and e. When a client application 'app' wants to access a functionality implemented in the FB 'f1' inside an ANA node or send it a data, it do not address it explicitly to the destination FB but has to search for its corresponding IDP 'a' instead. In the other sense, if a functional block f4 want to communicate with another FB or want to send data to the network interface. It has to resolve the binding between the IDP corresponding to the Ethernet interface and its registered IDP.

These identifiers IDPs are maintained inside an Information Dispatching Table (IDT) in which each ANA node keeps track of the bindings between IDPs and the other playground elements (FBs, ICs or bricks). A simple changing in the entries of this IDT allows a flexibility that makes easier the autonomic recall of the required FB. This dynamic recomposition will thus allow to the AN A framework to integrate more easily new communication paradigms and protocols that were not possible to integrate in the current Internet architecture.

In our task, we envision to exploit the benefits of the ANA framework to design a totally content-centric system for routing. We aim to demonstrate the feasibility of our system using the ANA abstractions and paradigms as well as confirm the benefits of such an architecture as it was already demonstrated by [8]. In our system, contents are registered by their providers using publications sent to the other ANA nodes. Clients interested in some content express their interest through keywords which they encapsulate inside content requests that are disseminated to the neighbours.

In order to avoid a blind flooding to all the nodes of the network, we use a cooperative and hop-by-hop filtering approach based on a filtering vector that allows to the system to select the optimised neighbours that can guide the request to the content locations without the constraint of their specific addresses. To reach this optimised selection of neighbours, we use an optimisation process and a filtering vector which we will detail in the following sections. Contents results are sent back and merged using the reverse path created by the disseminated content request.

B. A General View of the Content Routing

Before giving the details of our system design, it is necessary to explain our vision of content routing.

Content routing is a paradigm that appeared with the need to reduce the time of content access by providers. To accomplish this task, a set of content servers is proposed with a redirection mechanism that directs the clients to one or many possible content servers that provides the required content. Thanks to this set of servers, round-trip latency and overload on original content provider is prevented [10].

Currently, some varieties of content routing protocols are proposed in the literature. A first kind is based on ad hoc mechanisms and simply tries to resolve the domain name of the desired web site or content provider by a specialised name server. When a content request is received by the name server, it is resolved into the address of a server near the client, based on special routing and mapping mechanisms [3]. This kind of conventional content routing protocols suffers from a major problem which is the non-scalability because of the centralised names servers on which the resolve process is based.

Another kind of content routing protocol is based on the flooding of content requests to a set of content servers or brokers like the pub/syb systems[2]. The content servers and brokers match the received requests to a set of publications they store in their local tables, if a publication is matching the received subscription then the result is sent back to the initial sender with a pointer to the location of the corresponding content, and the requests is forwarded to the other neighbours who will execute the same procedure until the network is flooded entirely. Some optimisations of this kind of approach were proposed but there still a lack for a convenient approach that avoids flooding especially in large scale and heterogeneous networks while providing a good pertinence of the search results.

In our mind, we believe the content routing problem generally and specially in autonomic networks can be defined not as a simple qos and host location problem but as a filtering problem based on both the interest of clients and the descriptions of contents. Clients do not desire an access to a particular content servers but only to some piece of content, which can be described using friendly-like terms and structure. The whole infrastructure classically based on the naming and content servers will be assimilated to only one in which the content routing becomes a service totally transparent from the point of view of the clients and why not of the content providers. They have just to push their content registrations and requests to the infrastructure and the content should be sent back in response to this request. No more addresses or locations should be returned to clients.

We see also the management of the routing information as a problem of information storage and merging while the routing itself is assimilated to a filtering process which is executed on the basis of some filtering criteria. By this design, one can expect to prevent the translation of content host addresses from a network to another and to avoid the flooding of requests in large scale networks. The content informations can be easily managed in a self-organised manner and the unavailability of a content or a host do not mean necessarily the re-sending of another content request. Such a self-organisation and resilience is a very important one inside the autonomic networks as already analyzed in the ANA deliverable [6].

C. Content-centric routing for ANA : Basic Design

In this section, we propose a protocol that takes profit from the properties of the ANA framework to process content requests between cooperative nodes in a flexible and IP-independent way. This independence is important as it allows to route contents between nodes using the clients interests expressed inside their requests. We propose also to

Our content centric routing (CCR) protocol is represented by a functional block that is aimed to find out which IDP is associated to a specific content requested by clients.

The CCR FB is based on a hybrid and a hop-by-hop based routing protocol that is centered on the meta-data describing content. It implements a content centred routing protocol that filters and distributes content according to the interest of users. Users need just to express their interest by declaring some expressive keywords in their request. After that, content management and routing infrastructures, composed of specific ANA nodes that offers the CCR service, have the responsibility to match the client requests to specific filtering tables stored in each of these nodes. The matching process leads to the selection of the next neighbour(s) that provides the optimal content corresponding to the user request according to some metrics we'll present later. Then, requests are forwarded to the returned neighbours.

Each ANA node offering this service has to bind an IDP to its CCR FB. FBs use a publish primitive already defined in the ANA API in order to publish about their availability to offer the associate content to their peers and register the corresponding IDP inside the IDT.

Basically, each node willing to use the CCR has to disseminate a request containing a set of keywords describing the interest of a client in registered contents. ANA nodes that offers the CCR service maintain a content dispatching table (CDT) that allows to reach the node that hosts the optimal content matching the client request without having to search for it using addresses. As we said before, the filtering process is done "hop by hop" and no final destination address is returned by the system. This is a feature that helps to achieve the autonomicity of our system; if a request has to go through different networking infrastructures to reach a matching content to the client request, nodes do not have to translate a final destination address that can belong to an unknown name-space. They have just to specify the next IDP that is supposed to provide the best content for the request and forward it.

After that, an optimisation process is executed in order to select a limited set of the neighbours CCR IDPs to which the client request is forwarded instead of flooding the received request to all the known CCR IDPs. In order to filter these neighbours, ANA nodes have to maintain in a vector the value of three filtering metrics for each IDP : D the distance expressed in number of jumps which separates the processing FB of the potential contents matching the request, P which measure the popularity of the contents associated by and IDP and expressed as the number of access to the same IDP for similar requests, and finally M which represent the rate of satisfaction for the similar request by the associated IDP. Determining the most relevant IDPs for a content requests consists in selecting the top-k IDPs that maximize the P and M metric while minimizing the D metric.

Our CCR protocol is supposed to work independently from the networking infrastructure in which it is deployed for two reasons. The first one is the address-independence of our filtering process. The second reason is the design of the ANA framework so that the contents that we will reach are matched to some IDPs identifiers not to classical node addresses.

D. Decomposition

The CCR system is decomposed in a request parser, a request defragmentor, an optimal vector solver and a request processor. The figure 2 illustrates a generic decomposition of the CCR system. We will detail the organisation of its different FBs in the figure 3.

The request parser FB is responsible for parsing a client request which is received from a the client application and extracting the keywords contained in the request. Then it determinate the corresponding naming structure. Once the content request is received by the defragmentor FB, it process it in order to determinate whether it is necessary to decompose the request into many sub-requests. The generated sub-requests are treated separately in each mediation router and thus can be faster and simpler to process.The request processor is the module that will match the corresponding output

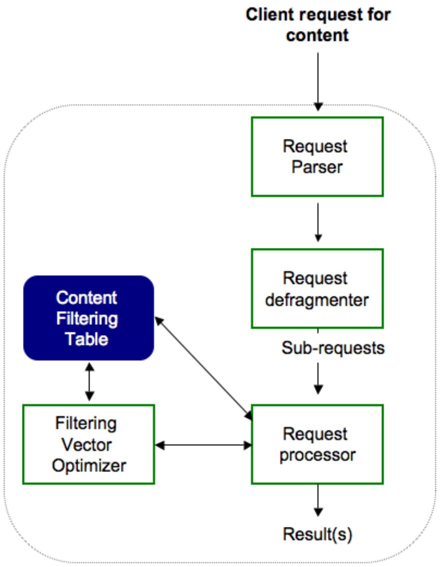


Fig. 2. Organisation of the CCR protocol

interface to the optimal filtering vector matching for each atomic sub-request. Then, the entries of the CDT containing the matching IDPs with their filtering vectors are transmitted to the filtering vector optimize to help the ANA node to compute the optimal neighbour that possesses a matching content to the client request. As The resulting IDPs will be returned again to the processor FB which has the responsibility of aggregating the different results of each sub-request and eliminate doubles if exiting. Once merged results are computed, the request is redirected to the corresponding IDPs as shown in the figure 3.

As we already said, the decomposition of the CCR system into individual functional blocks provides a flexibility of the system and allows to develop each functionality independently from the others. It allows also to test its influence on the functioning of the other blocks.

The figure 2 shows simply the interaction between the different FBs composing our content routing protocol when deployed on an IP network.

Each FB is registered with an IDP in the IDT table. An FB that want to communicate with an other one has to search for the associated IDP in the IDT.

When a client request is received on the IDP of the initial FB, is it processed as explained previously and the corresponding neighbour IDPs are returned by the processor FB. Then, an ETHERNET FB which is already implemented inside the ANA framework resolves the returned IDPs to ETHERNET addresses and forward the

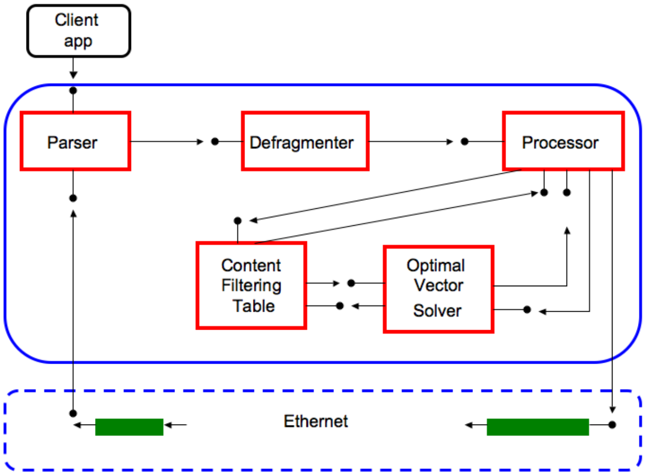


Fig. 3. Decomposition of the CCR FBs

messages to the corresponding neighbour. This example shows clearly that the IP-independence of the ANA framework do not mean that it cannot be used inside the IP-based subnetworks of the ANA network. In the contrary, it allows to make the communication between these kind of subnetworks and the others which do not use IP addressing more transparent and feasible.

E. Validation Scenario

An important step after the explanation of our content centric routing protocol is the validation of this system by an execution scenario.

A client application app_1 publishes a set of keywords Kj,i = k1, k2, k3, k4 that describes a content c_1 he stores to a specific CCR FB. The CCR stores the received keywords along with the IDP_1 associated to the publisher app_1.

If another app client app_2 want to search content, he have to send a request to its CCR FB. Similarly to the publication, the request contains a set of keywords $Ki = k1, k4$. The CCR FB first stores the received keywords along with the IDP_2 of the client app_2. Then, the CCR checks the CDT (content dispatching table) of previously published content descriptions and execute the filtering to match the required keywords with all the entries in the content dispatching table.

The CDT entries are maintained periodicly through a proactive dissemination process that disseminates the published contents to the network. The values of the filtering vectors are also recalculated all along the maintenance of the CDT table and thus provide dynamic and optimised values for each client request.

After a set of CDT entries is selected, the optimisation FB is called to select the most optimised entries for the requested keywords and forward the request to the resulting associated IDPs. The translation of the resulting IDPs to a physical ETHERNET address is managed by another ETHERNET Brick which is implemented in the current version of the ANA prototype.

We plan to validate this scenario in the next weeks. The aim of this validation phase is not to show the efficiency of the content-centric routing compared to the other protocols but only the proof of feasibility of this kind of routing using the ANA Framework described above.

III. CONCLUSION

In this paper, we have addressed one of the challenging communication protocols that have to be supported by the ANA framework which is the content-centric routing. Because of the heterogeneity that characterises the infrastructure of the ANA network, host-centric routing becomes a major problem that has to be resolved through an address-independent approach. We proposed to solve this problem through a content-centric protocol that achieves routing based on the clients interest in registered contents and a filtering process based on an optimised FB and a filtering vector. Thanks to the optimisation FB that we presented above, our protocol allows to forward the content requests to the neighbours that possess an optimised content for the same request.

Beyond the properties of the CCR protocol, the ANA framework facilitates the integration of such a protocol in its architecture thanks to its flexibility and the dynamic composition of its protocols stack when needed.

The decomposition of our CCR protocol into different FBs shows clearly the flexibility of the ANA, we tried to demonstrate that it is feasible using the ANA API and framework to provide content routing functionalities.

As already noticed in [8], whereas the use of IDPs to exchange data and messages between nodes offers an interesting flexibility, there still the overhead problem to be resolved. That is a part of the ongoing work in the ANA project. Another ongoing work is the implementation of the validation scenario using the ANA API already implemented.

REFERENCES

[1] T. Plagemann and V. Goebel and A. Mauthe and L. Mathy and T. Turletti and G. Urvoy-Keller, *From content distribution networks to content networks - issues and challenges*, Computer Communications, 551-562. 2006

[2] Y. Liu and B. Plale, *Survey of publish subscribe event systems*, Technical Report TR574, Indiana University. 2003.

[3] M. Pathan and R. Buyya, *A Taxonomy and Survey of Content Delivery Networks*, Technical Report GRIDS-TR-2007-19, The University of Melbourne, Australia. February 2007.

[4] C. Jelger and C. Tschudin and S. Schmid and G. Leduc, *Basic Abstractions for an Autonomic Network Architecture*, World of Wireless, Mobile and Multimedia Network, WoWMoM. 2007

[5] *ANA Blueprint - First Version Updated*, Workpackage 1 Deliverable 1.4/5/6, ANA Project FP6-IST-27489, December 2007

[6] *D2.1 : ANA Requirements*, Workpackage 1 Deliverable 1.2, ANA Project FP6-IST-27489, December 2007

[7] A. Keller and T. Hossmann and M. May and G. Bouabene and C. Jelger and C. Tschudin *A System Architecture for Evolving Protocol Stacks*, Proceedings of ICCCN 2008, IEEE. 2008.

[8] T. Heussman and A. Keller and S. Dudler and M. May, *Implementing the Future Internet : A Case Study of Service Discovery using Pub/Sub in the ANA Framework*, International Conference on Future Internet Technologies (CFI08). 2008

[9] F. Legendre and M Dias de Amorim and S. Fdida, *Some Requirements for Autonomic Routing in Self-Organizing Networks*, Proc. WAC2004, Germany. October 2004

[10] M. A. Sheldon and A. Duda and R. Weiss and J. W. O'Toole Jr. and D. K. Gifford, *Content routing for distributed information servers* Proc. Fourth International Conference on Extending Database Technology, Cambridge, England. March 1994