From active networks to cognitive networks

Manolis Sifalakis and David Hutchison

Lancaster University, Computing Dept.
LA1 3BX, Lancaster, Bailrigg, U.K.
m.sifalakis,d.hutchison@lancaster.ac.uk

Abstract. Future networks need to be autonomic self-managed and provide resilient servicing, even when the hardware fails. To achieve this goal, two fundamental requirements need to be satisfied: (i) the service management and provisioning must be independent and decoupled of the infrastructure management, and (ii) a certain degree of cognitive behaviour needs to be achieved at the service management level. In achieving the first goal, which in turn will enable the pursuing of the second goal, active and programmable networks will play an important role. A problem though arises when we try to build and use actual active networks, as most research so far has focused at the node level and has left us with a unbridged diversity of platforms and execution environments, which are largely uninteroperable with each other. We introduce a toolkit that provides a set of mechanisms aiming to bridge this diversity and provide a set of functionalities and abstractions for uniform installation and deployment of services over active and programmable networks.

Keywords: Active networks, self organisation, service mobility, dynamic service deployment

1 Introduction

In the future we would want (or even expect), that networks are able to sense our needs and react to our demands in a much more dynamic way than today's Internet. Furthermore, network providers would prefer that networks be reactive to high-level expression of policies and goals by auto-configuring and auto-organising themselves.

In order to develop network fabrics that will exhibit the aforementioned functionality we identify two key requirements.

1. The need to decouple the service management from the infrastructure management (this will enable resiliency in case or infrastructure failures).
2. The incorporation of cognitive functionalities in the service provisioning and management.

The latter is a longer term objective which we don't consider in depth here [1].

Dagstuhl Seminar Proceedings 04411
Service Management and Self-Organization in IP-based Networks
http://drops.dagstuhl.de/opus/volltexte/2005/96

2 Problem Statement

Active and programmable networks will play a key role as the enabling technology for the decoupling of the service management from the infrastructure by allowing the dynamic instantiation or migration of services wherever and whenever required.

The main problem that arises in this model is the diversity and multiplicity of paradigms, platforms, frameworks and deployment facilities in the active networking world (eg. [2], [3], [4], [5], etc). This diversity although useful (and possibly necessary), however it imposes a fundamental *interoperability* problem for the large scale deployment and use of active networks. The big challenge here is to find a way of abstracting all this diversity under a unified framework/API that will allow uniform and generic use of a programmable fabric by the overlying service solutions.

3 Proposed Solution

We propose the developement and deployment of a toolkit providing low level service deployment facilities at the execution enviroment level as a means to address the interoperability problem in the active networks world.

The toolkit provides a set of primitives and mechanisms that integrate and abstract various aspects and functionalities of active and programmable node platforms and helps organise them to form service specific network topologies.

Higher level implementations and services can use of the toolkit in order to uniformly address and access active and programmable resources without the need to know about the details of the underlying programmable infrastructure.

The functionalities provided by the proposed toolkit aim to:

- Determine the interfacing between active and programmable resources
- Discover and recruit (allocate) active and programmable resources
- Deploy service elements over the allocated active and programmable resources
- Assist the organisation and management of composite service provisioning

4 Service Deployment Toolkit Building Blocks

4.1 Network Pools of Active Resources

A two level overlay topology (intra-domain versus inter-domain) and an enabling mechanism is proposed for the discrete and fine grained organisation of the active resources in the Internet. Over this infrastructure, services for listing, browsing and allocating active resources can be developed. This mechanism aims to organise active resources in the internet as a *distributed pool of active network resources.*

4.2 Active Proxies: Service Deployment Aggregation Points

Once active resources have been discovered in the network, along the service path, the toolkit provides an *active proxy* service that aims to offload the end system from the need to instrument the service deployment process, by mitigating this responsibility to the active network itself. This enables a more secure model were all service requests are issued to the active proxy service point and all active resources within a domain need to trust only this specific service component for the installation of services on them. This leads to a single point of trust and control.

On the other hand the existence of many instanses of this service component in an active network reduces the potential of becoming a single point of failure or congestion.

The flexible design of this service allows the integration/update, in form of plug-ins, of elements for service specification processing, policy enforcement, cognitive decision making, etc.

4.3 A Common Service Loading and Deployment Interface

When all the conflicts between administrative policies, service dependencies, and user needs have been resolved and a plan has been reached of how a composite service will be laid over the active network infrastructure, it is time to install the service components on the selected active nodes in the network. The toolkit provides a *common service deployment interface* [6] that enables the installation, configuration and activation of the service components (that comprise the composite service) on active nodes, using only a set of basic primitives that abstract any platform specific service deployment mechanism.

4.4 Dynamic Kernel Level Overlays

After a service has been deployed over the allocated active resources there is usually need for coordination and synchronisation of the composing elements, at least on the control path. This, inherently, is an application suitable for overlay and possibly p2p networks. However, since an active node may be facilitating multiple (tens or hundreds) service components, using application level solutions, as most today's p2p systems do, can be simply prohibitive in terms of performance.

To overcome this limitation and provided that the service specification describes sufficiently how the service components can be coupled and managed, the toolkit considers a *kernel level dynamic tunneling* facility as a means of relaxing the cost of building overlays at the application level. This mechanism aims to enable the structuring of overlay service specific topologies at the kernel level whilst preserving the benefits of application specific routing.

4.5 Migrating Active Service Components

Finally, in order to leverage the independence of service provisioning from the underlying infrastructure we often need to resort in service mobility so as to easily and generically move runtime service code and/or state, between active and programmable node platforms.

The toolkit provides a framework against which a *mobile active service* can be programmed [7]. The aim is to mitigate the functionality related to service mobility from the application design to the hosting execution enviroment, thus making it generic and reusable and allow an application to either migrate itself, or be migrated by the managing execution environment.

5 Conclusions

Future networks need to be autonomic self-managed and provide resilient servicing, even when the hardware fails. Active and programmable networks are expected to play a key role in achieving this goal by enabling the dynamic deployment and service elements and structuring of service topologies. However the increased diversity of active network solutions and the lack of interoperability is a limiting factor. To address this problem we have proposed the developement and adoption of a toolkit that provides a set of low-level active service deployment and organisation facilities at the execution enviroment level.

References

1. Sifalakis, M., Mavrikis, M., Maistros, G.: Adding reasoning and cognition to the internet. In: Proceedings (companion volume) of 3rd Hellenic Conference on Artificial Intelligence (EETN). (2004) 365–375
2. Schmid, S., Joe Finney, Amdrew C. Scott, D.W.S.: Component-based active network architecture. In: Proceedings of IEEE Symposium on Computers and Communications. (2001)
3. Wetherall, D., Guttag, J., Tennenhouse, D.: Ants: A toolkit for building and dynamically deploying network protocols. In: In Proceedings of OPENARCH. (1998)
4. Hicks, M., Moore, J., Alexander, D., Gunter, C., Nettles, S.: Planet: An active internetwork. In: Proceedings of IEEE INFOCOM (3), 1124-1133. (1999)
5. Keller, R., Ruf, L., Guindehi, A., Plattner, B.: Promethos: A dynamically extensible router architecture supporting explicit routing. In: Proceedings of 4th Annual Int. Working Conf. on Active Networking (IWAN), Zürich, Switzerland. Number 2546 in Lecture Notes in Computer Science, Springer Verlag, Heidelberg (2002)
6. Sifalakis, M., Schmid, S., Chart, T., Hutchison, D.: A generic active service deployment protocol. In: Proceedings of 2nd International Workshop on Active Network Technologies and Applications (IECE). (2003) 100–111
7. Sifalakis, M., Schmid, S., Chart, T., Scott, A.: A framework for developing mobile network services. In: To appear in Proceedings of 6th International Working Conference on Active Networks (IFIP). (2004)