

Xweb: A Framework for Application Network Deployment in a Programmable Internet Service Infrastructure

O. Ardaiz, F. Freitag, L. Navarro

Computer Architecture Department, Polytechnic University of Catalonia, Spain
{oardaiz, felix, leandro}@ac.upc.es

Abstract

Current programmable Internet service infrastructures permit to activate Internet services on single nodes. Future Internet services, however, will be provided by a set of coordinated servers forming an application network.

This work presents a framework for deploying service as application network in a programmable Internet service infrastructure. Our framework consists of four building blocks: resource discovery, service specification, resource allocation, and service composition.

We have implemented a prototype of this framework called the Xweb. This prototype allows deploying web caching hierarchies and application networks of replicated servlet services.

1. Introduction

1.1. Programmable Internet Service Infrastructure

Society will require next generation Internet to increase the availability of services. A number of commercial and academic approaches are pursued to introduce new services in Internet.

In [11], an Internet-wide extension of distributed operating system, WebOS, is proposed, which takes into account the heterogeneity, scale and security challenges of Internet. Another approach is ActiveServices [2], which develops active networking technology, and aims at providing on demand service activation. Grids [7] provide service functionalities for computationally intensive applications. They gather Internet-wide resources for the provision of services, permit remote third party service activation, and offer mechanisms for local resource binding. Distributed Storage Infrastructures such as proposed in [4] provide service functionalities for non-executable services.

1.2 Motivation for Application Network Deployment

Current programmable infrastructures allow the activation of a service instance on a single node and, in an uncoordinated way, on multiple nodes. Future Internet services, however, will be provided by a set

of coordinated servers forming an application network. Sophisticated Internet services can be abstracted as application networks considering that servers will interact with each other on the application layer for replica consistency, request forwarding, and so on, creating an application layer network.

Examples of current application networks are content distribution networks such as Akamai [1], and peer-to-peer networks such as Freenet [5].

The existing programmable infrastructures allow activating a service on several nodes in an uncoordinated manner, either on the provider's behalf or on the client's request. Such lack of coordination control results in an inefficient resource utilization and service provision, which are provided on a best effort basis [9].

We claim that Internet services have to be deployed in a coordinated way forming an application network to achieve efficient resource utilization and a guaranteed service level.

In this work we present a framework, which allows deploying services as an application network on a set of coordinated servers.

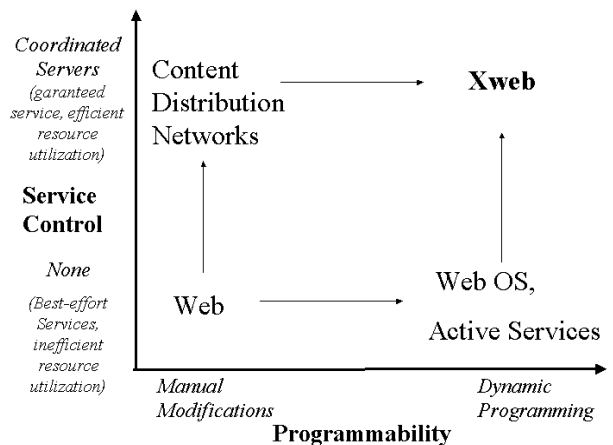


Fig. 1. Xweb vs. other Internet service frameworks.

As it can be seen in figure 1, our approach evolves from the programmable service infrastructure by adding a service control into coordinating application network servers, and it evolves from the current static configurable application networks by facilitating dynamic programmability.

1.3. Related Work

A number of technologies need to provide functionality to activate a service in several nodes. Grids deploy computationally intensive applications by co-allocation of resources in several processing nodes, for instance Globus GRAM [8]. Xbone [10] deploys virtual private networks, setting up coordinated virtual links and routing tables on several nodes. Darwin [6] is a generic framework for coordinated allocation of generic resources for services in several nodes.

2. Deployment Model

Deployment can be considered as "to spread out or arrange for effective action". Deployment exist in several engineering and scientific fields: sales forces deployment, army deployment, game teams deployment, etc.

Our deployment model for application networks deployment has the following components: 1) application users (App Users); 2) resources; 3) a service plan (see left side of figure 2).

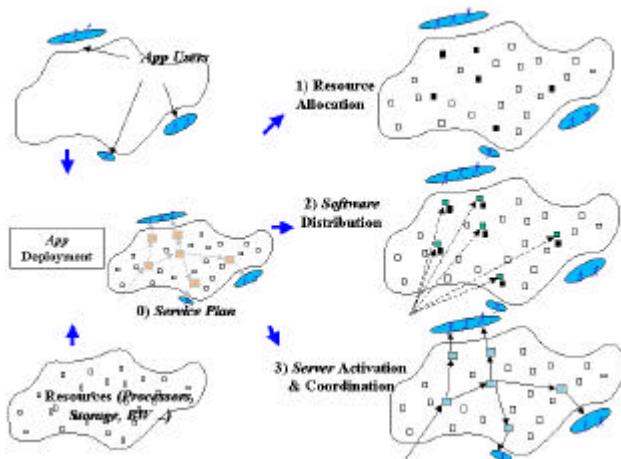


Fig. 2. Deployment model.

The application user requires a service. There are a limited number of resources on which to construct that service. The service plan determines which resources have to be allocated (Resource Allocation), how to coordinate the service instances, and on which servers to build the application (Software Distribution, Server Activation). The service plan matches the service demand with the resource offer and accomplishes two objectives: 1) provide the requested service level, and 2) minimise the resource utilization or resource cost.

3. Xweb Framework and Building Blocks

We propose the Xweb framework as a programmable service infrastructure for service deployment. It has several building blocks with

which to compose and deploy application networks, see figure 3.

The Xweb programmable infrastructure is made up of hosts with a programming environment that allows activating services. Every node implements a deployment mechanism, code distribution mechanism, local resource management, resource binding, and application network coordination mechanisms.

Deployment managers, which accept deployment request, receive application network specifications from service providers. Deployment managers interact with resource agents to deploy application networks.

Resource agents provide local resource management, allocating service traffic capacity and storage on the deployment managers' command. Resource agents download service software from a location provided by the deployment managers. To compose the application networks, the resource agents bind service code to local resources and to other service instances.

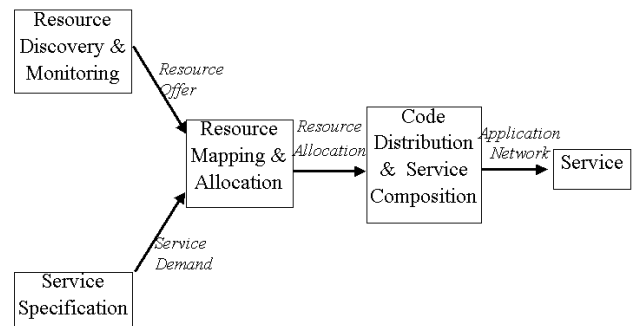


Fig. 3. Xweb building blocks.

3.1. Xweb Programmable Infrastructure and Resources

Currently in the Xweb prototype implementation, nodes provide Linux and Java-Tomcat programming environments. The resource agents are configured with data about the available local resources such as service traffic capacity, service storage capacity, nearby regions where service can be provided, maximum number of services accepted and code distribution mechanisms.

In our prototype the local resources such as network bandwidth, storage, and computing resources are configured manually. In an automatic way, bandwidth probes, load and storage monitors could be used to discover the capacity and availability of resources. The network topology adjacent to the resource could be obtained with network topology discovering tools.

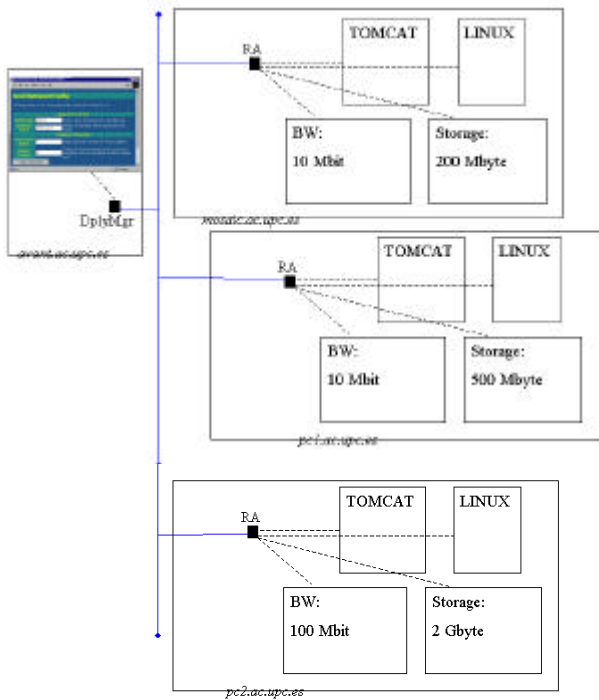


Fig.4. Xweb programmable infrastructure.

3.2. Xweb Application Network Specification

The deployment managers located on Xweb nodes have the web interface shown in figure 5. Service providers from remote location can access them and specify the required service.

In our Xweb prototype, the service is specified as follows:

- “name” of service, as DNS domain and server name.
- “code location” as the server address, which has the service code.
- “type” is the type of service, which has to be deployed. Currently the possible services are squid proxy caches and servlets.
- “minimum storage” is the capacity required for the provision of the service (in Mbytes) concerning the application code and the data.
- Internet “regions” indicate where the service must be provided, with the quality determined by the next input parameter “Max Distance”. Internet regions are expressed as a list of Autonomous Systems numbers ASnum.
- “Traffic” represents the total network capacity required to provide the service in Mbits.
- “Number of surrogates” represents the minimum number of nodes that must provide the service.
- “Max distance” is the maximum number of Inet hops between client and servers.

In the Xweb interface, the service provider also indicates when and for which duration a service

should be deployed (“start time”, “service duration” fields), how deployment has to take place - or fetched by the agent or pushed by deployment managers -, and how often deployment has to be optimised (“code distribution”, “evaluation interval” fields).

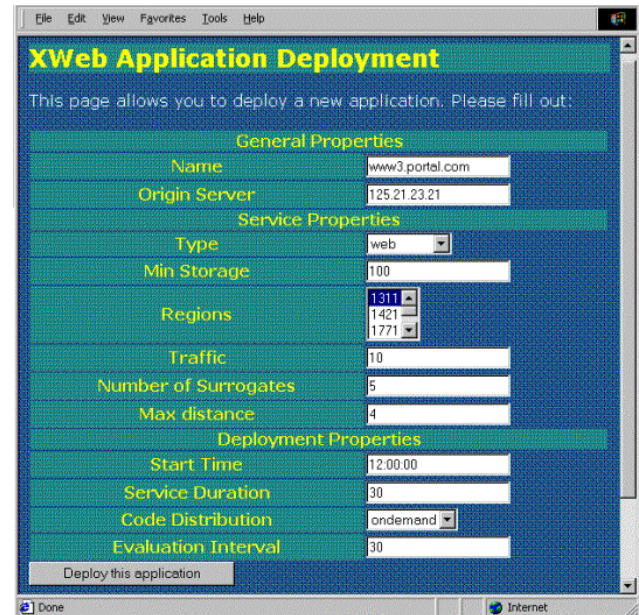


Fig. 5. Xweb interface for service providers to fill in application network specifications.

3.3. Xweb Resource Allocation and Service Composition Mechanisms: Multicast Injection

Resource allocation and service provision is achieved by the interaction between deployment managers and resource agents. Different models exist for this interaction: The centralized and the decentralized approach.

In the centralized model, the centralized manager allocates the resources, and the central manager instructs the resource agents. One of the known problems of the centralized approach is its scalability.

In addition to the centralized approach, we have implemented in the Xweb a decentralized approach, which we call multicast injection [3].

Multicast injection deploys an application network in three steps:

- First, a deployment manager, which has received a deployment request, creates a message containing an application network specification and multicast it to a global multicast channel. The nodes, which decide that they can provide an acceptable service according to the specification, allocate resources autonomously.

- Second, the nodes, which have decided to accept providing a service, join an application specific multicast channel and inform other nodes on their availabilities. Listening to these messages some nodes will proceed to join or leave in providing the service.
- Finally, the nodes select the nearest node on the opposite direction to clients as parent and configure an application network.

3.4. Xweb Application Network Composition

Xweb resource agents bind service code to local resources and to other service instances to form an application network. Resource agents keep a list of local resources, which are labelled as allocated or free. Resource agents allocate free resources to application networks using the centralized or decentralized model.

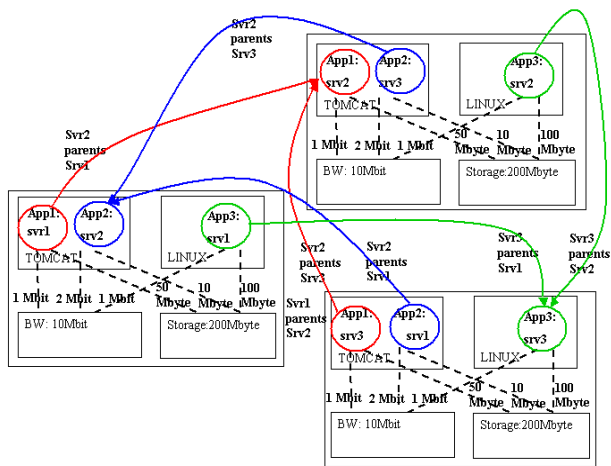


Fig. 6. Xweb application network provision: Resource agents bind application to local resources and to other service instances.

Figure 6 illustrates how resource agents bind application servers to local resources. In this example, we consider three applications (App1, App2, and App3). App1 and App2 executes in a Tomcat execution environment, whereas App3 is a Linux application. App1 at node 1 executes as server srv1; it is bound to 1Mbit of network capacity and 50 Mbytes of storage space, and it is configured so that server 2 at node 2 is its parent (service requests, which it cannot resolve, will be forwarded to the parent node), and so on.

At the Xweb prototype local resource binding is done using primitives provide by the Linux and Java-Tomcat programming environment.

3.5. Distributed Web Servlets Service Deployment in the Xweb

Figure 7 illustrates how Xweb deploys a distributed Servlets application forming an application network. Service providers request deployment managers to

deploy a servlet based service according to an application network specification. Deploying servlets involves providing servlet code to resource agents, which is bound to local resources through the Tomcat environment. Rules governing the application network configuration are shipped to a DNS server, which acts as a request redirector service. For instance, requests coming from clients in a certain region could be redirected to be served from servletA at node mosaic.ac.upc.es.

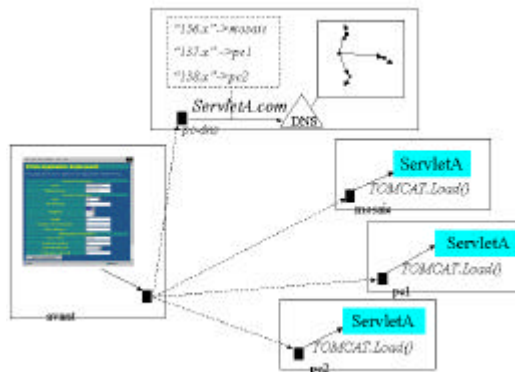


Fig. 7. Xweb prototype deploying a web servlet service.

4. Conclusions and Future Work

This work presents a framework for application network deployment in a programmable Internet service infrastructure. Our framework consists of the following building blocks: resource discovery, service specification, resource allocation, and service composition. In our implementation the deployment can be carried out based on a centralized or a decentralized model. We have implemented a prototype of the framework called the Xweb. Currently, the prototype allows experimenting with the deployment of web caching hierarchies and web servlet services using the centralized and decentralized approach in a programmable Internet service infrastructure.

References

- [1] Akamai Inc. "FreeFlow", Dic 1999, <http://www.akamai.com>
- [2] Amir E., McCanne S., Katz R. "An Active Service Framework and its Application to Real-time Multimedia Transcoding". Proc. SIGCOMM'98 Sept. 1998.
- [3] Ardaiz O., Freitag F., Navarro L., "Multicast Injection for Application Network Deployment", 26nd IEEE Conference on Local Computer Networks. Tampa, Florida, Nov 14-16, 2001.
- [4] M. Beck and T. Moore "The Internet2 Distributed Storage Infrastructure Project: An Architecture for Internet Content Channels". Computer Networking and ISDN Systems, 1998, 30(22-23): pp. 2141-2148.

- [5] Clark, I.: A distributed decentralised information storage and retrieval system. Url: <http://freenet.sourceforge.net/Freenet.ps>, 1999.
- [6] Chandra P. et. Al., "Darwin: Customizable Resource Management for Value-Added Network Services", 6th IEEE International Conference on Network Protocols (ICNP'98), 1998.
- [7] Foster, I.; Kesselman, C.: Globus: A Metacomputing Infrastructure Toolkit, in International Journal of Supercomputing Applications ,11/2, p. 115-129, 1997.
- [8] Foster I., Kesselman C., Lee C., Lindell B., Nahrstedt K., Roy A., "A Distributed Resource Management Architecture that Supports Advance Reservation and Co-Allocations", In International Workshop on Quality of Service, 1999.
- [9] Matei Ripeanu "Peer-to-Peer Architecture Case Study: Gnutella Network Analysis" 1st International Conference in Peer-to-Peer Networks Aug. 2001, Linköpings Universitet, Sweden.
- [10] Touch J., Hotz S., "X-bone: a System for Automatic Network Overlay Deployment", Third Global Internet Mini Conference in conjunction with Globecom'98, Nov. 1998. <http://www.isi.edu/xbone>
- [11] Vahdat A.; Anderson T.; Dahlin M.; Culler D.; Belani E.: `WebOS: Operating System Services For Wide Area Applications, in The Seventh IEEE Symposium on High Performance Distributed Computing. July 1998.