Toward Internet Distributed Computing

By Milan Milenkovic, Scott H. Robinson, rob C. Knauerhase, David Barkai, Sharad Garg, Vijay Tewari, Todd A. Anderson, Mic Bowman
Intel
Computer (Published by the IEEE Computer Society) May 2003 Pp. 38-45

Summary of the paper

The Internet of today is oriented towards information retrieval and its presentation in a format suitable for human reading. It is observed that the current data formats and protocols are not intended for machine processing and automation; they need to be modified. In addition to this, the evolving new design will use P2P and grid computing to turn the Internet into a distributed computing platform on a global scale. It would require adding a functional layer to it that can handle the resource allocations for application execution. This transparency in the Internet hosting environment would allow developers to use late binding to write applications without having to know the size and complexity of the underlying framework on which they would execute. This is a significant advantage over early binding that needs static allocation of resources and is less efficient – cannot adapt well to resource, load and availability variations. Requirements for distributed computing mainly include allowing heterogeneity in the network and scalability. Developments in wireless technology also mandate that the network provide complete support to mobile computing too.

The two key IDC design principles are - inherent intelligence in the network and being able to create self-configuring and self-organizing network structures. Having these features contributes to the improvement in scalability as the network becomes much more flexible. The IDC must support a flexible way of service advertisement and discovery mechanism as opposed to the static discovery mechanisms of the current networks. Dynamic configuration of the network structure and late binding enhances portability in addition to providing load balancing and increased reliability. Resource pooling is another key aspect of the IDC framework. It relies upon web services, p2p and grid computing to provide the desired functionality. Web services contribute the service description, discovery and platform-independent invocation features for the framework. P2P techniques allow the IDC network to do independent routing of messages. Grid computing implements the resource pool. It facilitates large-scale sharing of computational and storage resources and provides architectural solution and middleware for resource virtualization.

The discovery mechanism should work equally well in ad-hoc as well as traditional work environments, should be scalable, platform-independent and should provide support to use unreliable networks and transport protocols by taking care of the reliability issues.

Strength

The authors list multiple benefits of distributed computing like resource and information sharing, load balancing, scalability and reliability, fault tolerance and how could they be achieved in real life. The ubiquitous connectivity would enable Internet to be enhanced into an Internet distributed computing (IDC) environment. Sharing computing resources over an IDC environment reduces total cost of ownership and makes an efficient use of resources. Such evolution of an IDC framework build upon the pervasive computing and proactive computing approaches. Pervasive computing
idea includes many other devices into the network apart from just the computers (like sensors, home appliances and automobiles). Broadening this base warrants finding intelligent ways and techniques to configure the network.

The IDC network offers many excellent services, one of them being network virtualization, which essentially is having the nodes expose a set of resources and their functionality to the open web. To materialize this virtualization, the web services approach can be used.

IDC can help increase the use of distributed computing by taking it over the Internet. It will also take to the next step the idea of pervasive computing.

Weakness

1. Web services, grid computing, and P2P computing can work well only when there is a fast and reliable connection to the Internet.

2. There is no separate scheme for allowing mobile devices to connect to this distributed computing framework. They can connect only by mimicking a traditional network entity for the duration of the communication session.

3. Since the network can be dynamic, nodes getting added and leaving the network in real-time, we need some reliable and efficient mechanism to do resource discovery. Also it should not cause too much strain on the computing resources just to perform the resource discovery functions. Ensuring scalability of the network is also an issue.

Interesting Points

The authors’ research is centered around splitting the current Internet framework into individual services (which would most likely be web services) that can be described, discovered and dynamically configured at runtime. They contend that the system can be built as a combination of web services, P2P and grid computing technologies. However, this Internet Distributing Computing (IDC) approach is entirely different from the idea of having an “Internet Operating System” that provides some form of uniform or centralized global resource management. It is based upon certain key design principles and transparency in the network structure. The idea is to have the nodes (devices, computers) in the network share their resources and have them managed by the network. The IDC system adds these resources to a pool and provides them on-demand. After their use is over, the resources are returned back to the network pool. There are no stipulations on the network; it can be a public network or a private one as long as it uses standard Internet communication mechanisms.

Critical Questions

1. Before one can use the resources over the Internet, we need mechanisms to announce their availability to the devices on the network.

2. The resource discovery mechanism should work equally well in ad-hoc as well as traditional work environments, should be scalable, platform-independent and should provide support to use unreliable networks and transport protocols by taking care of the reliability issues.

3. Security and authentication issues exist with such a network