SELFMAN

Scope

Many new applications are appearing that take advantage of the scale of the Internet: file-sharing, collaboration tools, social networks, role-playing games, vendors, and so forth. These applications are like services: they should keep running reliably for long periods (even during maintenance) and they should be scalable. This is difficult to achieve with current technology. Who has not seen an Internet application crash because it is overloaded? Who has not seen applications break because of a missing plug-in or a network problem? What's more, it takes a team of specialists to keep the application running.

Selfman proposes a solution to this problem: we have developed a robust, decentralized transaction service. Any number of computing nodes make up the service, and nodes can come and go freely during execution. Nodes or networks can crash, and the transaction service keeps running. We plan three application scenarios for this service: a distributed Wiki (from the Zuse Institute Berlin), a machine-to-machine messenging application (from France Telecom), and a video streaming application (from Peerialism AB). At this point, we have a prototype of the transaction service that is running a distributed Wiki, and the video streaming application will soon become a product.

Advances

Selfman will provide a robust, decentralized transaction service implemented using standard Java tools. The transaction service will run on any number of computing nodes in a scalable way, maintaining its performance even up to measured systems of thousands of nodes. The transaction service will survive node and network crashes even in extremely hostile environments. There will be a basic security layer to prevent the most common external attacks. There will be extra services to enable three industrial application scenarios that are suggested by our industrial partners.

Such a transaction service does not currently exist. We are building the service by using advanced research work in structured overlay networks, extended with transactions implemented through symmetric replication and an atomic commit using a uniform consensus based on the Paxos protocol. Structured overlay networks have grown out of peer-to-peer technology and provide guaranteed routing efficiency and guaranteed search, when compared with peer-to-peer services. We have advanced the state of structured overlay networks by extending them to survive network partitions and extreme Internet-style failures.

Positioning in global context

No existing product exists to our knowledge that provides a robust, scalable, decentralized transaction service. Existing transaction services are "tightly-coupled": based on client/server architectures, where the transactions are implemented on a cluster machine. Our service is "loosely-coupled": it is a true decentralized Internet service.

Contribution to standardization and interoperability issues

This work builds on standard technologies but does not itself extend standards.

Target users / sectors in business and society

The users of this technology are application developers for Internet and technology providers for Internet.

Overall benefits for business and society

- The transaction service will greatly ease the development of scalable, robust Internet applications. Productivity of developers will be increased since they can focus on application functionality instead of Internet problems.
- The transaction service will greatly decrease the material investment needed for an Internet application. Current applications depend on a massive investment in cluster machines, expert technicians, and high-bandwidth Internet connectivity, because they are essentially using a client/server architecture. With Selfman's decentralized transaction service, this investment will be much decreased (no need for cluster machines) and it will be made incremental (it is possible to increase performance gradually by adding machines).
- The transaction service will be more resistant to denial-of-service attacks and sudden increases in load. There are two main reasons: first, the load is distributed across all computing nodes, and second, powerful computing nodes can be temporarily added if the load shows a sudden increase.

Examples of use

Selfman is building three scenarios to show the usefulness of our transaction service and related services:

- A distributed Wiki collaboration tool (defined by the Zuse Institute Berlin). Wikis are one of the most popular collaborative tools. They let a group of people create and organize large documents. When Wikis get large, performance goes down. We have built a Wiki over a structured overlay network using our transaction system for updates. This currently exists as a prototype application using a prototype implementation of our transaction service.
- A video streaming application (defined by Peerialism AB). We want to distribute video on demand to large numbers of customers and guarantee quality of service on the Internet. Customers come and go often, sometimes they look at the same movies and sometimes they don't. To manage all these video streams, we need to do dynamic reconfiguration. The video streaming application will soon be available as a product by start-up company Peerialism AB.
- A machine-to-machine messaging application (defined by France Telecom). This application creates a large ad-hoc network to reliably transmit messages across the world. If nodes go down or new nodes appear, the application has to keep working reliably and transparently.

Achievements

Selfman has the following results after 24 months:

- Decentralized transaction algorithm for structured overlay networks, publications and implementation (prototype)
- Distributed Wiki implementation (prototype)
- Network partition detection and merge algorithm, publications and implementation (prototype)
- Improved failure handling in structured overlay networks, publications and implementation (prototype)
- KOMPICS component model and implementation (beta release)
- Mozart distributed programming platform, publications and software (beta release)
- User requirements for application scenarios
- First guidelines for building self-managing applications

Contact the Selfman coordinator (see www.ist-selfman.org) for information on how to obtain software and other results.

THE ADVENTURES OF SELFMAN

title
Self management for large-scale distributed systems based on structured overlay networks and components

contract number
034084

type of project
Specific Targeted Research Project

contact point
Peter Van Roy
UNIVERSITE CATHOLIQUE DE LOUVAIN, BE
e-mail: pvr@info.ucl.ac.be

project website and partner list
http://www.ist-selfman.org

EC contribution
1 960 000 €

start date
01/06/2006

duration
36