

Realizing a Vision

Interesting Student Projects

Do you want to be part of a revolution? We are looking for exceptional students who can help us realize a big vision: a global, distributed storage system for the Internet. We have researched and developed Kangoo, which provides its users with a free, unlimited, and secure personal online data storage. They can use it to store their private files and access them from anywhere, share files with friends, or even publish their files to a global audience. It is our vision that in 3 years, more than 50 million people are using Kangoo. But to make this happen, a lot of interesting aspects still need to be researched and developed, for which we need your help.

Kangoo is a global, distributed storage system for the Internet. It harnesses idle disk space and bandwidth of participating computers to provide its users with a free, unlimited, personal online storage. A user can securely store his private files online, so that he can conveniently access them from anywhere at any time. More importantly, he can also use it to share his files such as photos, digital videos, music, or documents with his friends. Furthermore, he can even publish them to a global audience. He can search and browse this public area for whatever others have published, be that images of sunsets, funny video clips, or music of an independent artist. The openness of the system allows for a multitude of possibilities.

Services such as Yahoo Photo, Flickr, YouTube, and YouSendIt impressively demonstrate the need for free online storage. Thanks to the massive sales of digital photo and video cameras and the continuing spread of software production tools for images, videos and music, this need will constantly grow. More than 100 million users are already sharing their digital photos online today, and a rapidly growing number of people are sharing their video clips online.

Kangoo is based on leading-edge technology that we have researched and developed at ETH Zurich. It leverages the power of grid storage, resulting in very low operating costs, so that we can provide virtually unlimited free storage for everyone. We are currently finishing the development of Kangoo. The plan is to release the first private beta by the end of this year and the public beta next year. It is our vision that in 3 years more than 50 million people are using Kangoo every day.

But to make this happen, a lot of interesting aspects still need to be researched and developed, for which we need your help. Have a look at the open student projects on the following pages, or come up with your own idea that explores other issues of this big vision.

Do not hesitate to contact us. We are looking forward to hearing from you.

Dominik Grolimund
grolimund@inf.ethz.ch

PS: Don't forget the demo presentation on **August 21, 17:15 in IFW A34!**



Demo Presentation on August 21

In order to give you a chance to find out more about Kangoo, we will give a demo presentation on **August 21, 17:15 at ETH Zurich, IFW A34**. This will be the first demo ever! Just show up to have a glance at the alpha version, get more information, or ask questions about the student projects.

Student Projects at ETH Zurich

Be part of the revolution by tackling an interesting aspect of Kangoo in a **semester project**, a **Bachelor thesis** or a **Master thesis**. All projects can be adapted to your special interests and they can be narrowed down or extended so that they fit the criteria and workload of your desired project type (semester, Bachelor, Master). In case you are doing your Master in distributed systems, you can also do a **lab project** on Kangoo together with your friends. All projects are challenging and require a great deal of self-motivation. If implementation is necessary, very good programming skills in Java are required.

Distributed Systems: Prof. R. Wattenhofer

Attack!

Peer-to-peer systems are very open and exposed, and therefore susceptible to a number of attacks. We have tried hard to make Kangoo secure and robust. Challenge us! Try to attack Kangoo, any way you want. Try to exploit the weak points on different levels of abstractions, such as protocol, routing, resources, incentives, users, fragments, files. Interesting attacks could be to try partitioning the underlying overlay network, changing fragments without being caught, deleting other's fragments, etc. It will be interesting to see how much harm you can bring to the system.

Free-Ride First, Be Fair After

Kangoo is based on the resource contribution of the participating peers. Therefore, its success in practice crucially depends on its ability to cope with selfish peers which aim at consuming more than they contribute. We have built a number of mechanisms which try to prevent free-riding and encourage peers to provide their resources. In this project, you should take on the hat of a free-rider and try to exploit other peers as much as possible, for example by storing fragments without providing own storage, or by consuming lots of bandwidth without contributing upload bandwidth in return. You will also try to exploit an extension to the original protocol, a load distribution protocol similar to BitTorrent for popular fragments, which employs a tit-for-tat fairness protocol. If you find a potential point for exploitation, it will be your task to extend the protocol and fix it, ultimately preventing any potential to free-ride.

The World Turns Around

A distributed storage system needs to cope with the inherent dynamics of its constituent parts, the peers, which can join, leave, and even depart from the system at any time. In Kangoo, files are divided into blocks, which are encrypted and then encoded into a number of redundant fragments, which are stored on unreliable peers. When peers depart forever, new fragments need to be reconstructed in order to maintain a high-level of availability. Although Kangoo is built to cope with these dynamics, it would be especially interesting to investigate "extreme conditions", such as an exponential growth in the beginning of the system, or a sudden drop



in the user base. Another very important aspect is the geographical distribution of the peers on the world map. An equal distribution in different time zones would be ideal to keep the availability constant over time. But what happens if the distribution is skewed (e.g., only spread in Europe), and the access and online behaviour of the users is varying? In this rather theoretical work, you will investigate a number of important factors of Kangoo, apply models and growth scenarios, and find out whether the growth rate needs to be controlled and how much “buffers” will be necessary.

Analyze the Routing Protocol

Kangoo uses its own routing protocol to send messages to nodes in the overlay network. Thereby, no explicit structure among nodes is maintained, and still, routing is very fast. This is because the routing protocol exploits the properties of a small world graph. Simulations and experiments have shown that it scales very well and that it has a very good routing efficiency with a very small number of hops, given that there is enough traffic in the network. Many interesting theoretical questions, however, remain open and have not been tackled yet. In this project, you will shed light on the routing protocol and analyze its mathematical properties.

Access through the Web

Kangoo allows you to access your data from anywhere at any time...given that Kangoo is installed on that specific computer. But what if Kangoo is not installed in an Internet cafe where you want to access your files? In this project, you will solve exactly that problem! You will explore and evaluate the different solutions (e.g., Java applets, Java web start, or an AJAX web interface) and you will develop and implement either a stripped down version of Kangoo that can be run anywhere, or a web interface to access your data by relaying the data if necessary.

By-Pass Firewalls


Kangoo’s protocol is based on UDP, and hole punching techniques are used to by-pass routers and NATs. However, corporate environments are often very restrictive, sometimes even filtering out all UDP packets. In this project, you will try to make Kangoo work even in the most restrictive environment, by extending the basic protocol of Kangoo and relaying traffic through special nodes over TCP, similar to what Skype does to overcome firewalls.

Integrate Kangoo into the Ecosystem

The openness of Kangoo allows for a multitude of usage scenarios. However, it is not the goal of Kangoo to reinvent the wheel for applications that already exist. Kangoo must be integrated smoothly into the existing ecosystem, such as e-mail, chat and VOIP (Skype, etc.), blogs and social networks (MySpace, etc.), photo browsers (Picasa, iPhoto, etc.), web search (Google, etc.), buddy lists and profiles, etc. In this project, you will integrate Kangoo into the existing ecosystem by opening up Kangoo on different fronts, making files universally accessible through URLs, connect systems using RSS, jump from application to website, expose a developer API to allow for mashups and other applications to be built on top, etc.

P2P-Backend?

P2P mostly refers to pure client applications that implement all the functionality. However, in order to build a huge global online storage, mechanisms are needed to monitor, control and intervene into the system



if necessary to keep it stable. In this project, you will get an in-depth look on what is going on behind the scenes and you will develop interesting backend tools and a control center where the connectivity and the growth can be monitored, and parameters can be tuned. Furthermore, you will develop statistical tools to evaluate parameters and measurements.

Information Systems: Prof. D. Kossmann

As in Amazon: If You Like This, You Will Also Like...

One of the most interesting features of Amazon is its recommender system: If you look at a book, it recommends you other books that you might like based on what other users have liked. Recommender systems mostly use a collaborative filtering approach and have successfully been applied to a number of domains, such as books, movies, and music. In Kangoo, the recommender system will be a very important building block in the public area, where it should recommend files related to the current file, for instance suggesting a speech by Bill Gates when you watch a keynote of Steve Jobs. In this project, you will implement the recommender system into Kangoo.

Search and Rank

Kangoo does not only allow users to share files among friends, but also to publish them to the world. In order to make them findable, the owner can add a title, descriptions and tags, similar to Flickr or YouTube. Good search and ranking algorithms are necessary to find the relevant images or videos among the millions of files in Kangoo. We have developed a rudimentary search and ranking algorithm – however, glancing at Google, you will know that this is a very interesting but rather endless research topic. In this project, you will improve, enhance and extend the existing search and ranking algorithms find the most relevant among millions of files.

Large-Scale Search Engine


While the previous topic focuses on the algorithms, this project tackles the problem of scalability of the search engine. When millions of users want to search in millions of files, the search engine needs to be built very scalable. In this project, you will design and implement a clustered search engine architecture that distributes the load among the servers so that it can cope with terabytes of data and massive parallel access.

Search Locally

Kangoo respects the privacy of the users and encrypts all data before it leaves your computer. Therefore, only published files can be searched over the search server – no information about private files and files shared with friends are known to the search server. Therefore, in addition to the public search engine, a local search engine is needed to search in your own private files and those shared by your friends. In this project, you will implement a local search engine à la Spotlight or Google Desktop, which lists matching files as you type.

Going Live and Growing Large

So far, Kangoo only runs as an alpha version and we can freely and easily change implementation and the information architecture. Kangoo will launch with a small user base, but hopefully grow very fast. This changes the requirements of the information system very much within a short period of time, and every change to the information architecture must be made in a running system which can be very dangerous. In this project, you will investigate possible problems, look ahead, design a flexible



information architecture, which can easily migrate to new servers, introduce clusters and balance the load, so that it can cope with the ever changing requirements.

Browse the Folksonomy

Websites such as Delicious, Flickr, or YouTube have made tagging very popular: Users can tag any item by adding a label to it. This has brought up new and interesting ways to browse huge information repositories. In this project, you will adopt the best of these ideas (e.g., tag clouds) to Kangoo, but also be creative and find new ways to browse the millions of files in Kangoo in a fun way, for example by connecting related files or put files with similar tags into virtual folders.

More than Bits and Bytes

A file in Kangoo is more than its bitstring. It is usually enhanced with a file name, its size, tags, and comments. But there is even more: Photos know when they have been taken, and songs have an artist, a title, and an album. In this project, you will enrich the files as much as possible by extracting all contained information. You will need to display these information, make it searchable and find new ways to browse, sort and group, for instance by putting photos into virtual albums based on their date or group songs by artists. This is a very creative project; you will look at existing projects such as Flickr, read the literature, adopt new ideas from operating systems such as Mac OS X or Windows Vista, and most importantly, invent and implement your own ideas.

Notify Me

When you publish a video, interested people should be notified. Similarly, if you share new photos with your friends, they should know about it without constantly polling. The notification system in Kangoo will be a very convenient feature. There might be an internal notification mechanisms which highlights users, folders and files where changes have occurred, but there might also be an external notification over RSS. This would allow you to update your website or blog automatically whenever you publish some new photos. In this project, you will develop and implement such an event and notification system for Kangoo.

Software Engineering: Prof. P. Müller

Different projects in software engineering are available under the supervision of Prof. P. Müller. Please contact him to discuss your personal preferences for a project based on Kangoo in software engineering. Sample topics include exposing a developer API layer so that other applications can be built on top of Kangoo, the obfuscation of the Java byte code, localization issues of an international software, design patterns and architecture of a large software system, testing and verification on different levels. Kangoo provides a large and real code-base in Java and can serve as an interesting case study for various software engineering topics.