Cross-Desktop

A Web-based Desktop-System

Dirk Fokken

Table of Contents

Dirk Fokken

Motivation

It was only two years ago, myself being in a project as the client started to constrain the access of mail accounts of popular service providers. They may have done it for serious reasons, but I found myself being cut of my business. Therefore, I started to look out for web-based mail software in order to setup the required service on our own dedicated web-server we already run at that time.

As I found some software and installed it, it took quite some months to feel comfortable with the new environment. Not because of a worse user interface, but more because of knowing to have personal data on our own remote server. Also, I was worrying about the situation having a private machine, but doing much synchronisation work with my local pc at the home office.

Since the situation was not satisfying I decided to forget about the hassling synchronisation and started to manage content on the remote server. The mail system only had a linear web-drive, mainly to manage mail attachments. Soon the necessity arose to have folders and copy and paste functions available.

After starting our own development of a web-drive extension, called cross-drive today, for the mail system, soon more functionality was desirable as like search, zip, downloads, bookmarks, and a simple text editor. After a while of using this system by me, staff and many friends it became imminent that the integration of mail and file-system could be enhanced. We also started using cross-drive as the standard web-site management tool to replace simple FTP.

The solution of our thoughts was a complete web-based desktop being a productive, content creation solution, to possess extensibility and connectivity with a mobile working habit without any asp-contract with a third-party business. The Cross-Desktop.

The Approach

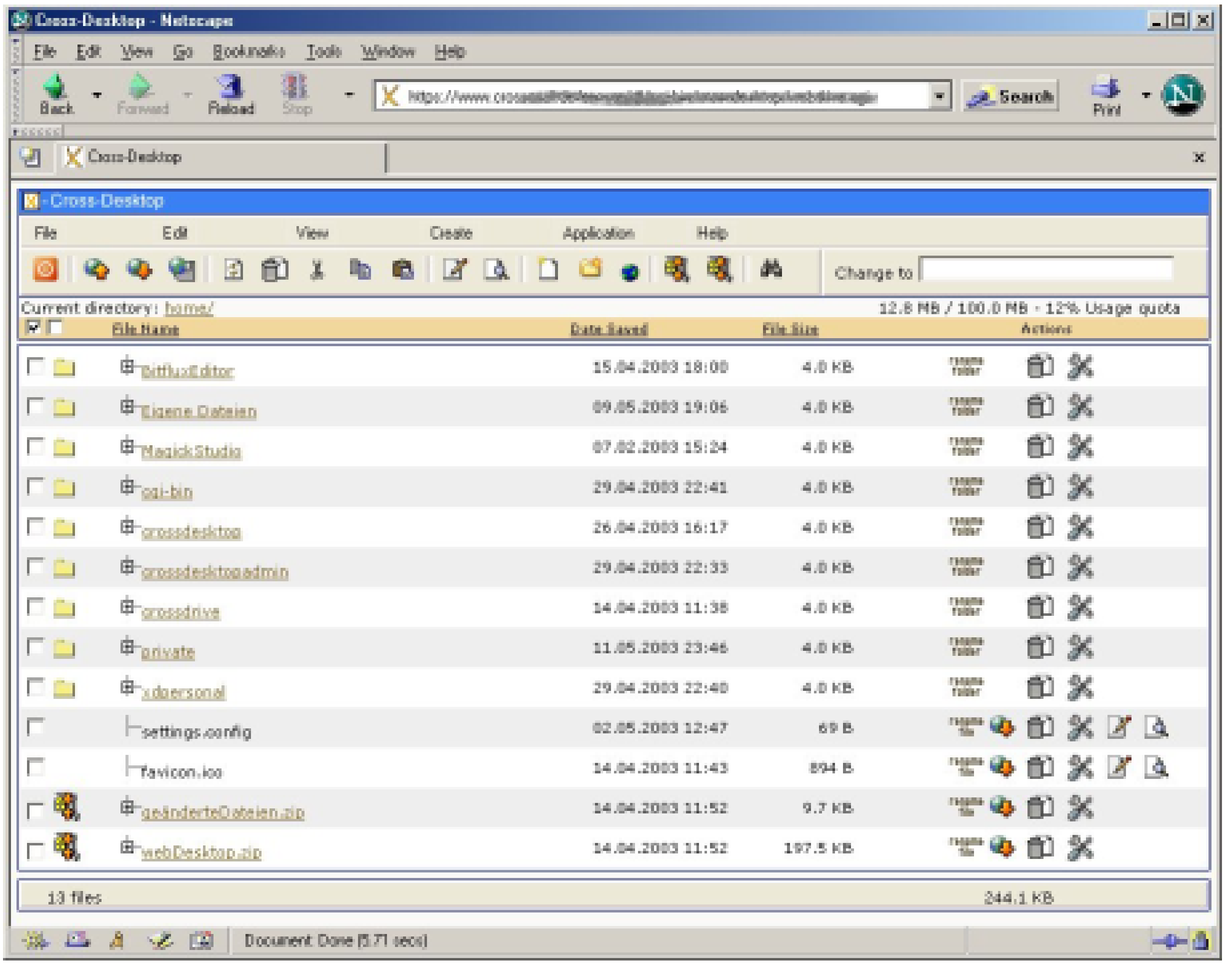
In order not to have a complex window in a window situation being implemented inside the browser as the desktop is supplied by the device which we already use, the graphical desktop on the pc or other devices, we decided to start at the explorer level. We think this decision is more targeting the usability. But we wanted to have a classical user interface style containing menus, icon-trays and still windows. We also wanted to have a clipboard. The technology should be very simple and common. So we decided to use HTML and JavaScript on the client side as most browsers support these features reasonably today. For the server side we decided to use Perl as it is very rich in features and offers comfortable application integration. Perl is also a reliable platform independent language.

With these base techniques secure communication can easily be established by the use of HTTPs. Additionally, we would be able to manage our content, work, and enjoy 24 hours a day, 7 days a week at near any place around the world from nearly any internet connected computer having a common browser installed. It is not evaluated, but this should be about a 95% availability approach.

I.e. in Germany you can rent a dedicated web-server including 2.0 GHz Intel Celeron, 256 MB RAM, 40 GB disk space, and 75 GB traffic with direct backbone connection for about EUR 49,00 a month or less. Calculating a two year subscription period computes to EUR 1.176,00. This a reasonable offer for a mobile working environment that you can even share with your staff or friends.

The Solution

Being implemented as an explorer like interface, the Cross-Desktop embeds directly into the desktop of the operating system being used. The system provides its own menu and icon bar. Even a title bar is integrated. Therefore, the desktop offers a familiar impression to the users. Although, the Cross-Desktop having an explorer like entry point there is still a desktop experience by the offering of its own title bar, menu, and icon bar.


Cross-Desktop - Netscape
File Edit View Go Bookmarks Tools Window Help
Back
Forward
Reload
Stop
https://www.cros[redacted]
Search
Print
Cross-Desktop
Cross-Desktop
File Edit View Create Application Help
Change to

Current directory: home/ 12.8 MB / 100.0 MB - 12% Usage quota

File Name	Date Saved	File Size
BitfluxEditor	15.04.2003 18:00	4.0 KB
Eigene Dateien	09.05.2003 19:06	4.0 KB
MagickStudio	07.02.2003 15:24	4.0 KB
cgi-bin	29.04.2003 22:41	4.0 KB
crossdesktop	26.04.2003 16:17	4.0 KB
crossdesktopadmin	29.04.2003 22:33	4.0 KB
crossdrive	14.04.2003 11:38	4.0 KB
private	11.05.2003 23:46	4.0 KB
xdpersonal	29.04.2003 22:40	4.0 KB
settings.config	02.05.2003 12:47	69 B
favicon.ico	14.04.2003 11:43	894 B
geänderteDateien.zip	14.04.2003 11:52	9.7 KB
webDesktop.zip	14.04.2003 11:52	197.5 KB

13 files 244.1 KB

Document: Done (5.71 secs)

Image 1 shows the Cross-Desktop initial interface.

The handling of the Cross-Desktop is very easy. The use of familiar desktop-application construction elements diminishes the need of learning application logic. Standard menus like File, Edit, View, Create, Application, and Help comply with the application menus of the embedding browser. With DSL or T1 based internet connections the handling experience is no different to standard explorer applications of many desktops of common operating systems. The Cross-Desktop even compares to the Microsoft Windows Explorer which embed the Internet Explorer as the driving application.

A productive desktop system claims for a full set of file management functions. Cross-Desktop already offers the round up of these functions, but also extend to comfort functions for finding, compressing, and downloading files.

All of these functions are not only integrated, but interact with each other, i.e. you can cut, copy, and paste one or more files into zip files. Another example is that you can navigate in zip files like you can do in directories. This operating comfort presents a maximum of flexibility to the user.

A very powerful behind-the-scene feature of a common desktop is the clipboard. Cross-Desktop also has a clipboard integrated that can be addressed by the cut, copy, and paste functions available in the icon-bar and edit menu. The clipboard shows its power when you run more than one Cross-Desktop window at a time. A new Cross-Desktop window can be created by clicking File/New Window (Image 2) in the Cross-Desktop integrated menu. Now you can navigate through different directory trees at a time starting at the root directory and cut, copy, paste documents across all browser windows running the same Cross-Desktop login session.

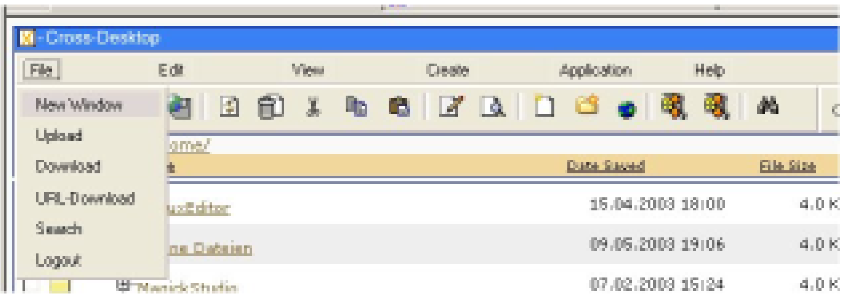


Image 2: Create another Cross-Desktop window and use the clipboard across windows.

Cross-Desktop would remain to be a simple file store if it would not integrate connectivity, extensibility, and further applications. Connectivity means that you can directly download via server-to-server connection. With this feature you can retrieve small files, but also large quantities. Files consisting of 700 MB and more are proven. Currently, HTTP and FTP requests are supported. By the integration of certificates HTTPS requests are possible. Other protocols can be integrated as well.

Extensibility is the ability of creating different behaviour in Cross-Desktop. The documented API allows you to address nearly every peace of the system. Additionally, the open source of the Perl programming language Cross-Desktop being implemented in leave no desire unsatisfied.

After all, application integration allow to integrate things like mailing system, calendar, office-components, graphic application, multimedia, application development, financial applications and many others, but all of these sharing a simple web-based desktop interface. We plan to create xml based office components with WYSIWYG experience. An image manipulation application is under development that will integrate the web-based ImageMagick user interface called MagickStudio. A new mail system integrating into the file system of Cross-Desktop should be available by the Linux Tag.

It is also planned to use the always-on approach of Cross-Desktop when installed on a server having access to crontab. We think of agents that provide information selection from the internet while the user being logged of the system, so that user's information requests are always up to date when the user logs in again. Agents can be article matches from common news sites as well as business specific selections based on keywords. You can decide whether to download (copy) the article or create a bookmark.

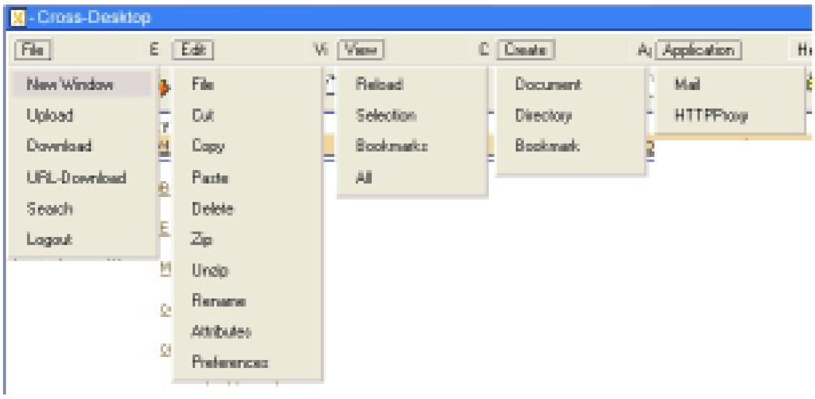


Image 3: The current menus are showing the main functionality.

The Technique

The structure of the Cross-Desktop is fairly simple. The desktop is split into 4 visible areas plus one invisible area (image 4). The menu bar contains the overall functionality of the Cross-Desktop. Its representation on the server side is controlled by the file menubar.cgi and some configuration files containing JavaScript or menu data. The menu can be fully customised in design and behaviour.

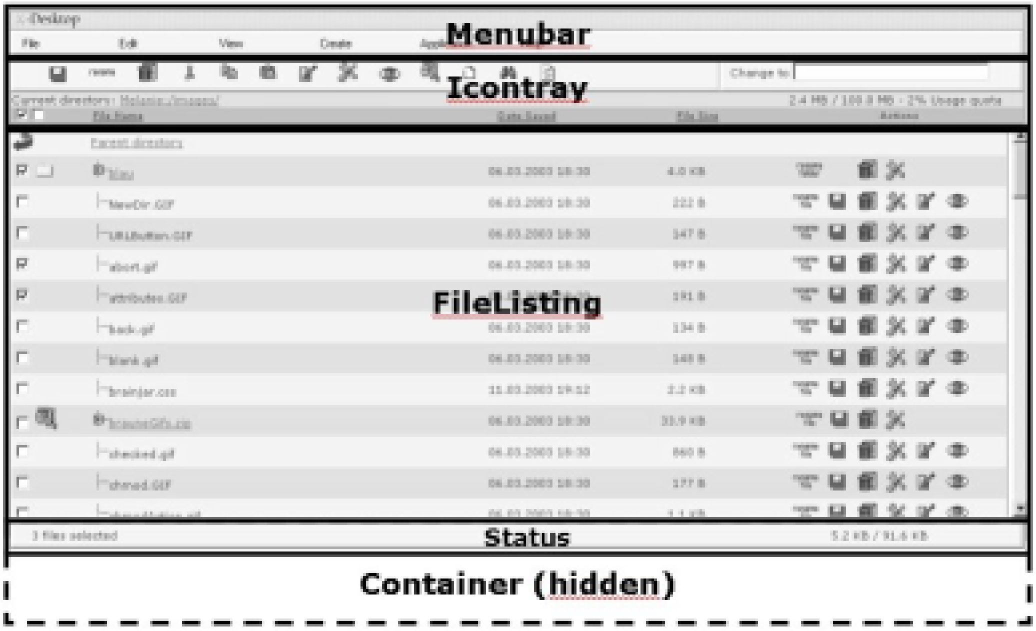


Image 4: Main graphical interface structure of the Cross-Desktop

The icon bar offers direct access to common functions. On the server side the control is handed over to the file icontray.cgi. The icon bar is fully customisable and configurable in design and behaviour. Communication on the browser side works with JavaScript, calling either menu functions or individual functions.

The most expanding area of the graphical user interface of the Cross-Desktop is occupied by the FileListing. The FileListing consists of the navigation functionality of files and directories. Whereas directories and files can be logical representations of server side structure that is not a single file or directory but can be function. On the server side communication is handed over to the kernel.cgi. Performing security checks and functional checks kernel.cgi will call a target function via functions.cgi. This model allows a standardised and simplified client/server communication with ease of configuration on the issue of extending desktop functionality (image 5).

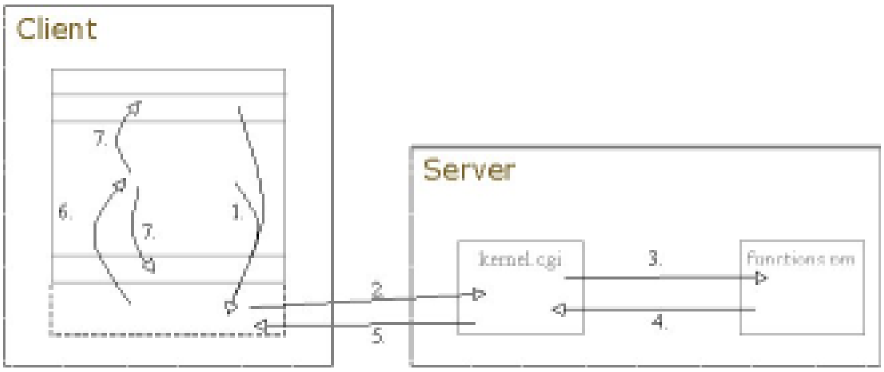


Image 5: Browser communication with the server

All communication is being based on HTML, JavaScript, and Perl. Communication will need no other than standard port 80 for HTTP protocol and port 443 for HTTPS communication. The Cross-Desktop can be centrally configured to use any other port if requested. It is important to know that the philosophy of the Cross-Desktop follows the port 80 approach in order to result in high availability and reliability. There is no restriction to extend the Cross-Desktop with features like Java (TM), Flash, or any other plug-in technology.

Case Scenarios

Cross-Desktop is a flexible desktop environment that can be employed for diverse tasks. Our personal use is the mobile approach being consultants ourselves working at different places every day, but always having an internet connection available.

Another task for the Cross-Desktop can be using the desktop interface as a front for document or content-management-systems. In this approach Cross-Desktop can be employed as a standard application interface for many applications that aim at content creation and content management.

Enterprises may be interested in a thin-client approach that may become achievable by Cross-Desktop being configured in a way that every employee will find their individual or personal desktop configuration and applications on any workstation running any operating system around the world. Documents are managed in large storage environments keeping track of backup and security. Configuration can distinguish availability of information upon internet, extranet or intranet level login. Letters to customers or business partners being written in an integrated office component based on xml may be electronically sent off and will be distributed to the nearest location to the addressee and then printed out automatically and being delivered. This saves time and costs.

With a future mobile internet connection via WLAN or UMTS sales representatives can work with a Cross-Desktop providing file-management and application integration without the hassle of synchronisation of notebook content to a desktop base with the risk of data corruption and missing backup. Direct connection and integration with large web-based server applications would be common.

Whatever application you think of, the Cross-Desktop can be attached to document-management and backup, consist of flexible configuration and availability, and secure your data.

All product names mentioned in this article maybe trademarks or registered trademarks of their respective owners.