

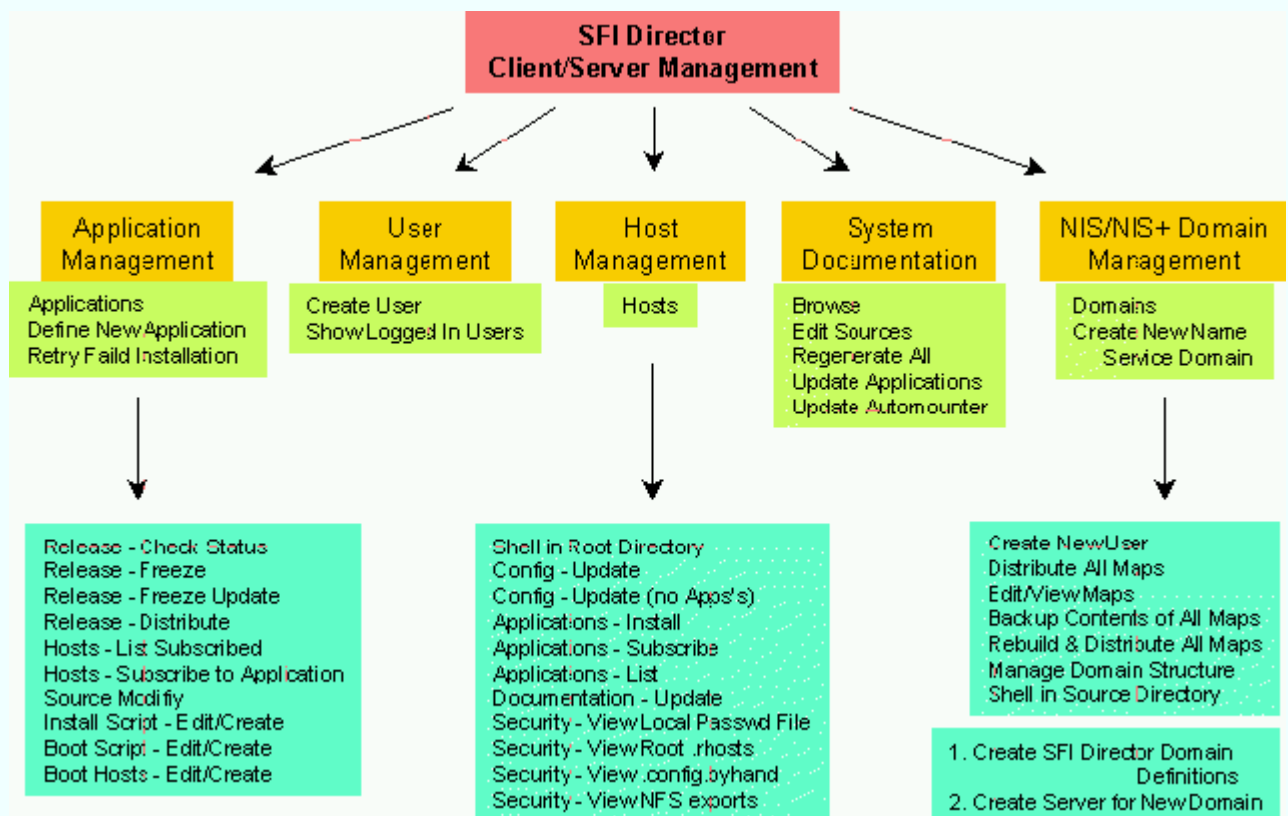
SFI Director Submission to the Swiss Open Source Software Competition

This web page is my entry for the [Swiss Open Source Software Competition](#) sponsored by the ETH, /ch/open and Sun Microsystems.

Overview of the SFI Director

The SFI Director is a tool for managing distributed, heterogeneous UNIX Systems. Its functionality includes System Configuration, Application Distribution, NIS and NIS+ Management, User Creation, and Dynamic System Documentation. The Director is a mature product with several reference installations in Switzerland, managing networks between 10 and 250 UNIX nodes, under Linux, Solaris, Digital UNIX, and HP-UX. It currently consists of 35,753 lines of code and 36,964 lines of comments and documentation.

SFI Director Menu Structure



The name is an allusion to the German word for Conductor, [*Orchester-Dirigent*], whose job is to ensure that all members of the Symphony are playing, not identically, but together and in harmony.

Introduction to the Author

My name is Peter Stevens. I am the designer and principle author of the SFI Director. I started my career in 1982 working for a "little" company called Microsoft - at the time there were ca. 150 employees. After coming to Switzerland in 1985, I worked more and more with UNIX and eventually became an independent consultant. I started working on what became known as the "Client/Server Concept" in 1992 and acquired the rights to the software in 1996. In 1997, I founded SFI and sought to build the company around the program I renamed the SFI Director.

In 1998 I became aware of the Open Source Movement and quickly became convinced of its importance, not just in University circles, but to commerce, government and industry. Released in January 1999, Version 3.0 of the SFI Director was the first version to be released under the GPL. I believe it is the first commercial

program in Switzerland to have been converted to open source (no one has challenged me on this assertion!).

I have since become a lobbyist for Linux and Open Source Software in Switzerland. In March 1999, SFI co-sponsored Switzerland's first "Open Source Strategy Seminar," with the CEO of Red Hat, Bob Young, as Keynote Speaker. To further Open Source Software in Switzerland, I have participated in press events, written articles (most recently for the NZZ), given presentations (/ch/open, Internet Expo, Common - the IBM User Group) and become active in professional circles (member of [/ch/open](#) Executive Committee, instigator of their [Open Business Lunch](#) - to encourage "networking" among professionals - and their *Jugend Flügel*, together with the ZLI, which should help show to students and apprentices that there is an alternative to a Microsoft-only education.).

Submission

The rest of this document is structured according to the Call for Participation. Each section documents that the corresponding criteria have been satisfied.

- [Conditions for Participation](#)
- [Selection Criteria](#)
- [Other Requested Information](#)

Conditions for Participation

The following conditions for participation have been specified by the organizers and are fully satisfied by the SFI Director.

1. **Swiss Residence or Nationality** - I moved to Switzerland in 1985 and became a Swiss National in 1995. Work on the SFI Director began at the SIG in Neuhausen in 1992.
2. **Genuine Open Source License** - The SFI Director is licensed according to the [GPL](#) - GNU General Public License and is distributed entirely in source form.
3. **Full Documentation** - This SFI Director is extensively documented. Each of its 197 modules and files is documented in HTML format in the structure of a classical [man\(1\)](#) page. See for example [rmtconfig](#), the system configuration tool, or [app_dist](#), the applications distribution utility. [Function](#), [usage](#), [arguments](#), [related commands](#) and [bugs and limitations](#) are all extensively documented. Graphics are used extensively to facilitate comprehension. Furthermore, an additional 57 files provide:
 - [Table of Contents](#) and [Index](#)
 - [Menu Structure](#)
 - [File Formats](#)
 - Answers to [FAQ's](#) (needs work),
 - [HOWTO's](#) for common tasks,
 - Documentation of the [Release History](#) and [Change Log](#)
 - [Credits and Contributors](#)
 - Descriptions of the [conceptual framework](#) of the Director

Documentation is for the most part stored in the source code, then extracted and converted into HTML by [Quadif](#), the QUick and Dirty Internet Formatter.

The SFI Director also generates documentation about the [system environment](#) which it is managing. This includes:

- The status of all known [hosts](#)
- Directory and distribution information for all centrally managed [applications](#)
- [Automounter configuration](#)
- [IP Address Usage](#)
- NIS and NIS+ [Name Servers](#)
- [NFS](#) Client Server Relationships.

■ [Overview of all Start Scripts](#)

The information is usually updated daily and can be accessed through files or through an active web server.

4. **Detailed Change Log** - Change logs have been maintained since the beginning. Originally just comments code and in the headers of each file, this information was integrated into the CVS history when the entire SFI Director was put under source code control in the Fall of 1999. The following is an extract from the CVS change log:

```
2000/09/04 07:04:06 (stefank) <many>
    updated documentation
```

```
2000/08/24 13:04:57 (taeby)      etc/csdocs/host_getinfo:
NISchk(): Suche nach /var/yp/*/{"ypservers.pag,passwd.byname} um die
bedienten NIS-Domaenen zu finden. Einige Systeme (Linux) verwenden keine
.pag-Endung auf NIS-Tabellen.
```

```
2000/08/23 07:46:23 (taeby)      etc/pkg_install:
pkg_add auf Systemen mit rpm: Falls /home Automounter-verwaltet ist, dann
verschiebe Dateien fuer /home nach /export/home
```

```
2000/08/23 07:32:09 (taeby)      etc/app_mkrel, etc/app_freeze:
mkmetafiles(): Fuege "" in Pruefsummenskript ein - so funktioniert auch
das Freezen von Dateien mit Sonderzeichen (Space, '$', etc.) richtig.
```

```
2000/08/21 08:38:24 (taeby)      etc/sfi_progress:
set pipe to non-blocking via fconfigure in order to avoid "blocking
channel driver did not block" errors.
```

...<big snip>...

```
1992/12/17 (peters)      etc/rmtconfig.classes, etc/rmtconfig:
Initial Version (actually a rewrite of the version I
previously lost.
```

```
1992/12/17 (peters)      etc/app_mkrel, etc/app_freeze:
Initial Version
```

```
1992/12/03 (peters)      etc/buildpasswd:
Initial Version
```

```
1992/10/01 (peters)      bin/sav, bin/rest, bin/check:
1. Version
```

The full change log can be found under

<http://www.sfi-director.org/doc/man/doc/Revisions/changes.html>

5. **Co-Authors must be listed in the documentation** - A list of Co-authors and other contributors is part of the standard documentation delivered with the software, and can be found under

<http://www.sfi-director.org/doc/man/doc/contrib.html>.

Selection Criteria

1. Originality and Uniqueness of the Software

The Director builds on a concept pioneered by the "yellow pages" (aka NIS) from Sun: All machines in a single domain receive the same administrative data (passwd, hosts, etc). The Director extends this concept in four ways:

1. All machines in a single domain should have the same basic *behavior*.

2. Each machine in the network should have a defined *desired state*, which is the basic behavior plus individual customization.
3. There should be a defined relationship *between* domains. For instance, all administrators can log into all servers and all clients, but normal users can only log onto clients.
4. That the actual state conforms to the desired state should be *guaranteed* through automation and documentation.

These concepts are implemented through the tools to automate system configuration ([rmtconfig](#)), Application Distribution ([app_dist](#)), Name Service Management ([ypvi](#)) and System Documentation ([docs_build](#)).

The Director goes beyond basic reactive system management by providing the tools to automate and guarantee the system configuration. This makes it possible to manage and incrementally improve the system environment. It also differentiates the Director from tools like rpm or pkg_add - which automate the installation of individual packages, but do not attempt to define the overall configuration of the system - or tools like linuxconf, webadmin - which provide remote and/or GUI access to System Management tasks, but do not define an overall state for each system.

Any resemblance to the objectives of ISO-9000 quality certification is purely co-incidental!

2. Quality of the Code

The oldest installation is now seven years old. No one who has deployed the Director has been able to replace it with anything better (and several people have tried!). These satisfied users are the proof of the Director's inherent quality.

The SFI Director is written using `/bin/sh` and `wish`. Therefore it cannot be compiled with `gcc`.

3. Portability of the Program and Usage of Tools

The SFI Director is known to be managing approximately 670 Nodes for 6 companies at 17 sites running at least 8 Versions of the leading Unixes, including Solaris, HP-UX, and Linux. [I believe all the SunOS 4.1.x Systems were decommissioned prior to January 2000.]

As a tool built from scripting languages, rather than compiled languages, there is no need for `autoconf`, `automake` or similar tools. The Director is now managed using [SourceForge.net](#), which provides essential Project Management functionality, including CVS trees, majordomo lists, bug tracking and "help wanted" advertising, etc.

The original objective of the Director was that it should run on a basic SunOS system, literally "out of the box." Therefore it is built on the least common denominator of the classic UNIX utilities, especially `grep(1)`, `cut(1)`, `sed(1)`, and `awk(1)`. It makes extensive use of NIS (later NIS+) and the automounter. Perl was not used because, until Linux became widespread, perl was not usually included with the operating system. The Director's role has been to make such "exotic" tools as perl consistently available throughout a large network.

`Make(1)` is used to manage the distribution of NIS and NIS+ maps. A unique feature of the Director is its Name Service Make Depend ([ns_mkdepend](#)) utility, which allows automatic generation of the `Makefile` used to control Name Service (NIS/NIS+) data distribution.

The Director does not seek to reinvent the wheel. If a good utility is available from another source, we use it. For example, basic OS installation is handled by the native tools of the OS or Hardware supplier, e.g. `Jumpstart`, `Kickstart` or `ALICE`. These tools are close to the hardware, but far from operational needs of individual companies, departments or users. The Director builds on these utilities to do the basic installation, and provides layered tools to provide the necessary customization to the needs of the individual company, group or user.

4. Overall Impression of the Software

I guess this is something you'll have to decide for yourself! I believe in the SFI Director! Some of our customer projects have gotten international recognition and the Director was key to winning those projects. (Visit [success.redhat.com](#) and search by country for Switzerland. At least one and maybe two SFI Projects are cited). I hope you have gotten a feel for what the Director is and what it can do,

especially for the sites that use it!

Other Requested Information

1. Description of the Software and of the Problem Which it Solves

In the broadest sense, system management is a very complex task which requires establishing consistency between many different components - some at runtime, some at initialization time, some on the client and some on the server - in order for complex client/server networks to run properly. The larger the network, the more difficult the task. The poor reliability and high cost of maintenance associated with client/server networks is caused by the exponentially expanding complexity of this task.

The SFI Director is a collection of scripts, most of which run in batch mode, which guarantee proper configuration of UNIX and Linux systems. The Tcl based menu program and logging console provide the necessary user interface to guide the administrator through complex tasks. Driven by simple, file-based data structures, it installs applications, sets configuration files, creates users and manages the Name Services. The Director's configuration files are meta-data, which define the desired state of each system in the network. The Director can therefore automatically recreate the state of any system. Fully automatic (re-)installation and configuration is possible, even after a catastrophic system failure.

2. History of the Project

The development of the Director really corresponds to the classical open source case study. A customer's problem needed a solution - I solved the problem based on some simple principles - and the solution thrived (despite many obstacles).

Back in 1992, I was consulting to SIG in Neuhausen/SH. They had just decided to buy a Sun Server and 65 Sun Workstations to replace an obsolete Minicomputer system. They knew that managing their existing 30 Workstations had been difficult and that they needed a better solution if they were going to manage 100!

I wasn't the first person to be asked the question, but I was the first person with a good answer: the "Client/Server Concept." The original version was quite small: about 10 scripts to do basic system configuration, application distribution, and multi-domain yp management for a SunOS 4.1.3 environment. At the time, I thought we would be managing 5 to 10 application on 40 hosts; little did I know that the number of applications would grow to over 100, and the number of hosts to over 200!

In my department we set up a test lab and built a proof-of-concept. Had I asked upper management beforehand whether I could develop these tools, the answer would almost certainly have been "No". However, before talking to management, I was able to demonstrate a working version, and so the project was approved and we went forward with the deployment.

And it grew. The first additional requirement was to patch an application. The IT Department was later outsourced to EDS, who introduced HP Workstations. They wanted other things too, like the automatic system documentation. The Swiss Stock Exchange wanted NIS+ support and to run mixed DEC and Sun Environments. Later they wanted a very high standard of documentation. ASCOM wanted improvements to the application distribution, NIS support and our ability to handle geographically distributed sites.

In November 1998, several members of the SFI Staff attended SANE, the System and Network Administrators Conference/Europe in Maastricht. There we heard Bob Young for the first time and realized that Open Source was changing the rules of the software industry. It became clear that UNIX was rising from the ashes, and that especially in the area of system software, OSS was going to eat alive anyone who attempted to do commercial systems products.

At that point I released the SFI Director under the GNU General Public License and started lobbying for Open Source Software. I preferred to take the risk of adapting early to the OSS Model, rather than risk being destroyed by it later. To my [slight] surprise, all of my customers accepted this change quite positively, even the customer who had just bought the last commercial version. To date, no one has cancelled their support contracts.

We tried to launch the Director in Spring 1999 as an Open Source Project. We ran into problems because the Director is a mature program and deals with a very complex problem. It is difficult for a developer to grasp the entirety of the program itself and the problem it is trying to solve. The Director also had certain preconceptions about its system architecture which made it difficult to use in other environments. We also had some personnel issues and therefore did not have the resources to manage an OSS Project.

Despite these difficulties, we did win some reference projects with Linux and the SFI Director, most notably the IVI Project and the Elektro-Material project.

Today, the SFI Director is a solid, complete product which serves an important function for Swiss Companies and Institutions. namely Elektro-Material AG, SIG, ASCOM, the Swiss Government (IVI - Institute for Virology and Immunoprophylaxie), and the SWX Swiss Exchange. The IVI Project is probably the first commercial project in Switzerland to deploy Linux on the desktop.

Now that the staff is available to do it, we have relaunched the Director as an OSS Project. It we announced the availability of the latest update (v3.11, published on September 1) on Freshmeat and moved the project management to SourceForge. This competition spurred us to get the project on-line a bit quicker than we had originally planned.

SFI Director 4.0 will be a complete rewrite in Java. We intend to launch the project soon (as I write this in September, the plan is to announce the 4.0 Project in October or November) so that we can get requirements, participation and hopefully buy-in from potential users at an early phase.

3. My Role in the Project

I was the designer and principal author of the SFI Director through Version 3. Various other people have worked on it, most notably Stefan Keller, who wrote most of the documentation, and Tom Aeby, who has taken the lead on relaunching the Open Source Development Project and in designing Version 4. I feel the changing of the guard is important, so that fresh blood comes into the project with new ideas and fresh experience with new tools and technologies such as LDAP or Java.

Today I am primarily involved in the project as a consultant in the design process and as cheerleader, coach and marketer. This Open Source Project/Product is serving as my cornerstone for building SFI in to a profitable and (I hope) well-regarded company in the Swiss IT Marketplace.

Why I Believe That I/the SFI Director Merit the Prize

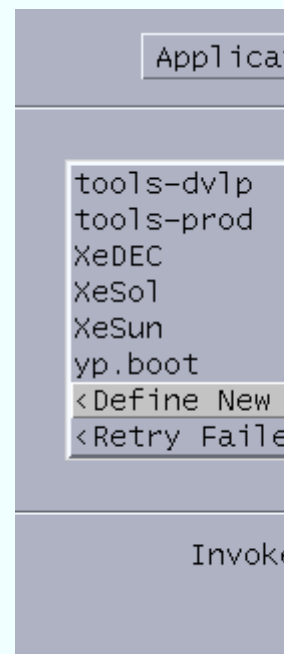
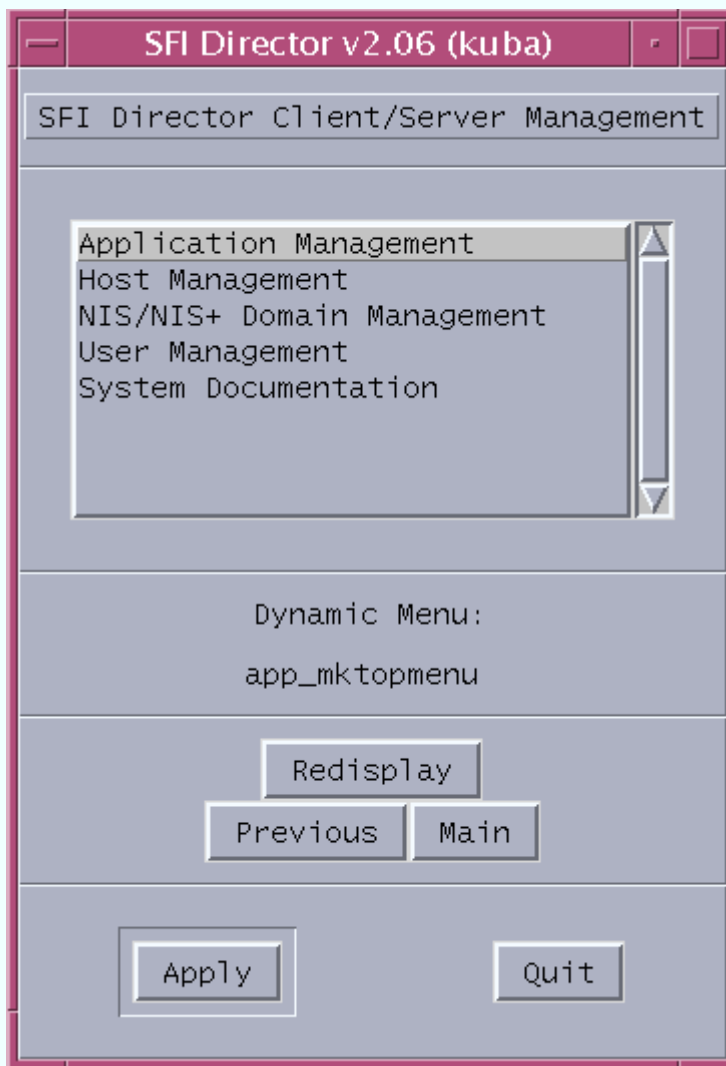
I believe there are a number of factors which speak for the SFI Director:

- It is a solid, complete product which serves an important function for Swiss Companies and Institutions.
- It is the first commercial product in Switzerland to be migrated to an Open Source License.
- It proves that Open Source Software from a Swiss company can be accepted by Swiss Institutions for mission critical tasks.
- It proves that a Swiss company can be give its software away "for free" and still be profitable and well regarded in the marketplace.

As for myself, I can only point to my role in promoting Open Source in Switzerland. If someone considers that noteworthy, well, um so besser!

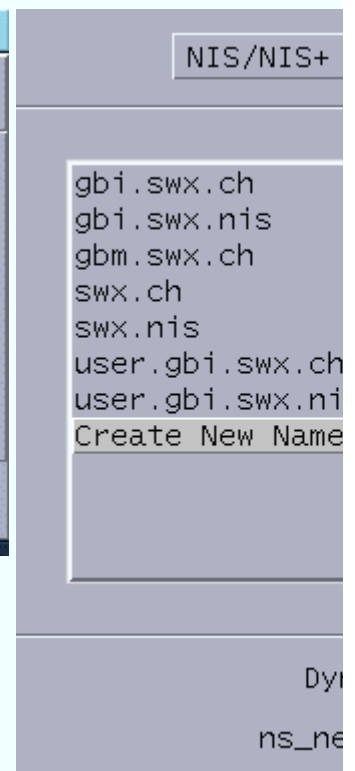
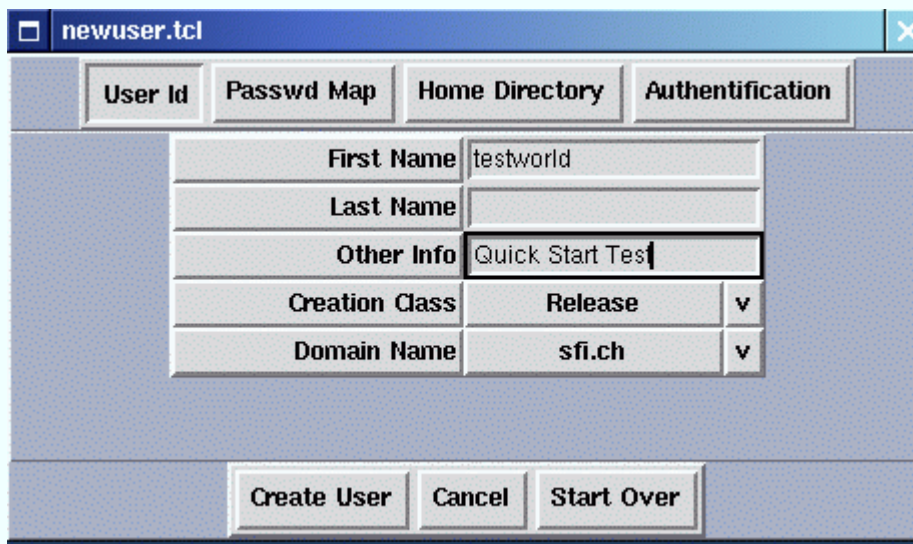
Last but not least, my submission to the contest was ready on October 1st, 2000, the original submission deadline.

4. Screen-Shots of the Program in Action

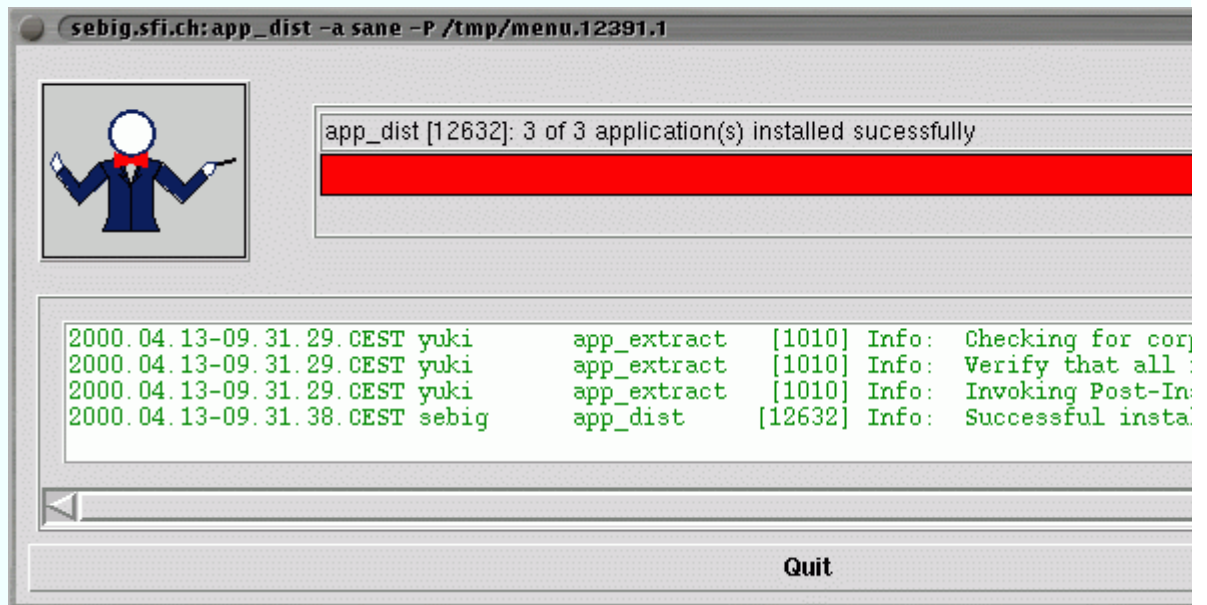


SFI Director Creating a User

SFI Director I

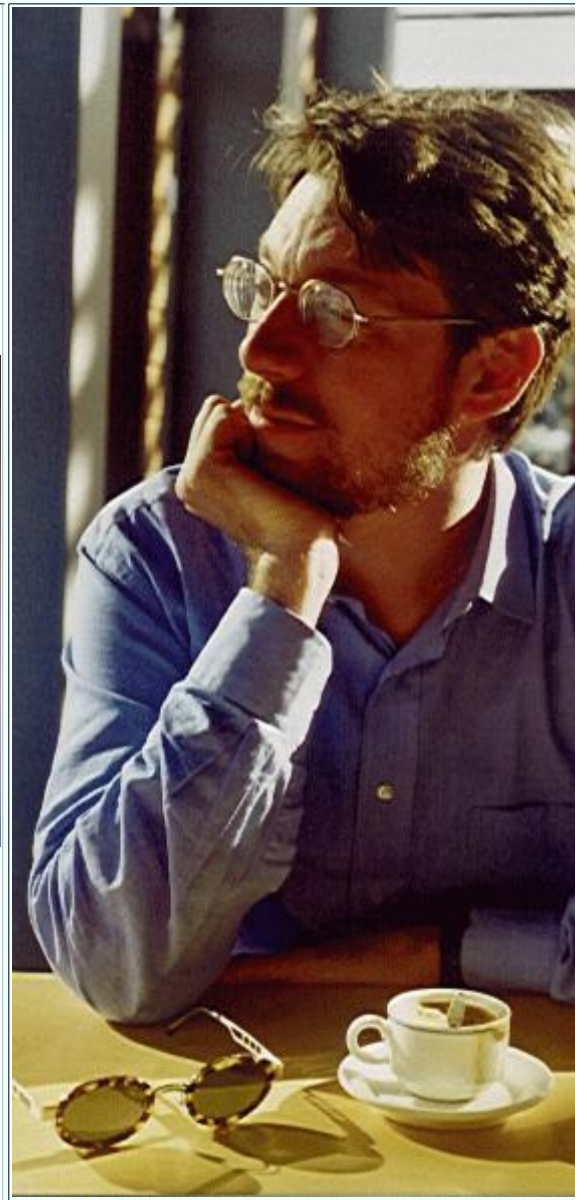


Running the Application Distribution



5. Information about myself (with pictures)

Peter B. Stevens, Jr. Email: peters@sfi.ch	
Work	Home
SFI Open Source Services Stettbachstr. 10 CH-8600 Dübendorf	Eichweg 9 CH-6300 Zug
Tel. +41/1/824-4900 Fax. +41/1/824-4901	+41/41/763-6720



6. Download Link

The SFI Director Download Page is http://sourceforge.net/project/showfiles.php?group_id=11622