

Disnix: A toolset for distributed deployment

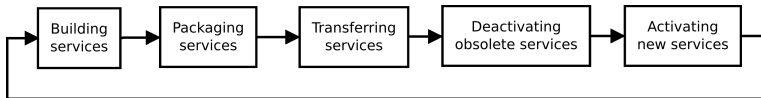
Sander van der Burg Eelco Dolstra

Delft University of Technology, EEMCS,
Department of Software Technology

September 20, 2010



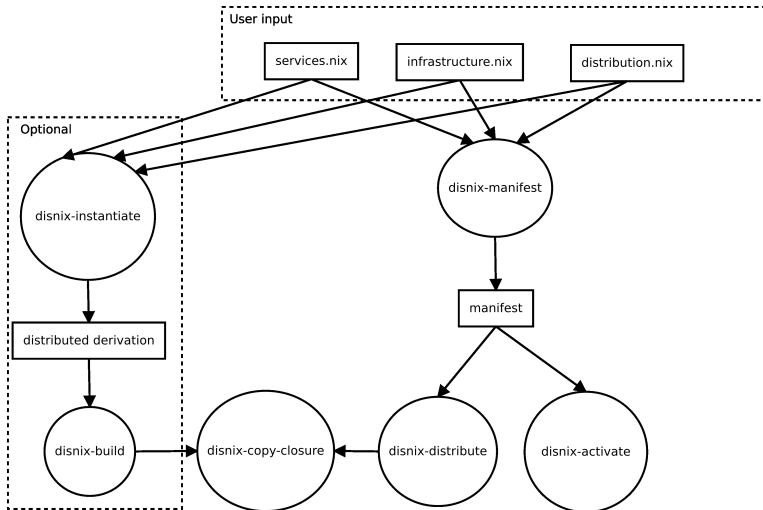
Disnix is a tool to automatically deploy a distributed system in a network of machines



- General issues:
 - Automatic deployment
 - Reliable deployment
 - Efficient deployment
 - Atomic upgrades/rollbacks
- Domain specific issues:
 - Security
 - Privacy
 - Performance

It is complex to implement and maintain a toolset dealing with these issues!

- How to develop this toolset in a convenient way?
- Some non-functional requirements of the domain cannot be solved generically
- Maintaining everything ourselves is a lot of effort:
 - Programming language compilers/interpreters: C, Java, Python, PHP, ...
 - Libraries: Apache Axis2, libxml2, ...
 - Infrastructure components: Apache Tomcat, MySQL, ...
- The tool is used in a distributed setting. How to test?



User input

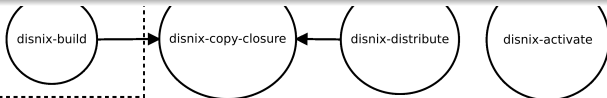
services.nix

infrastructure.nix

distribution.nix

Architecture

- Architecture is mainly composition of processes (command-line tools)
- Each tool performs a separate deployment task in the deployment process:
 - Building
 - Transferring
 - Activating
- Inspired by Art of UNIX programming book



User input

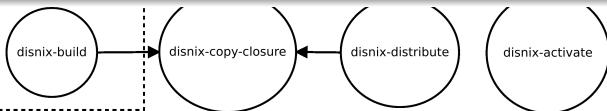
services.nix

infrastructure.nix

distribution.nix

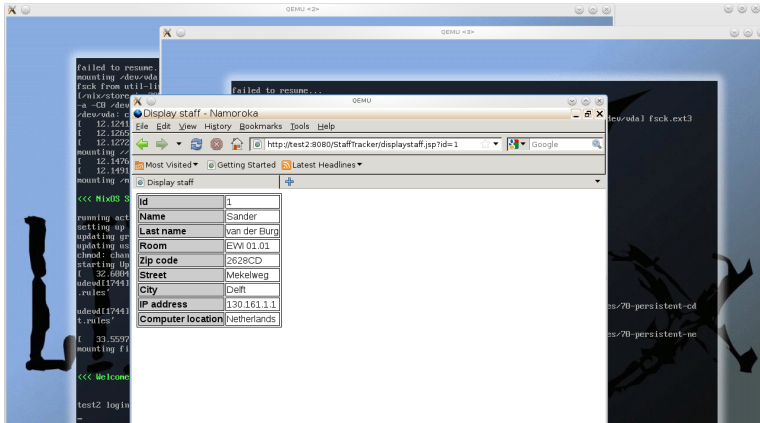
Architecture

- A user can easily perform steps separately
- Testing and debugging is relatively easy; things can be easily scripted
- Components can be implemented using various programming languages: Nix, C, shell scripts
- Prototyping is relatively easy. First use a high-level language, implement later in a low-level language

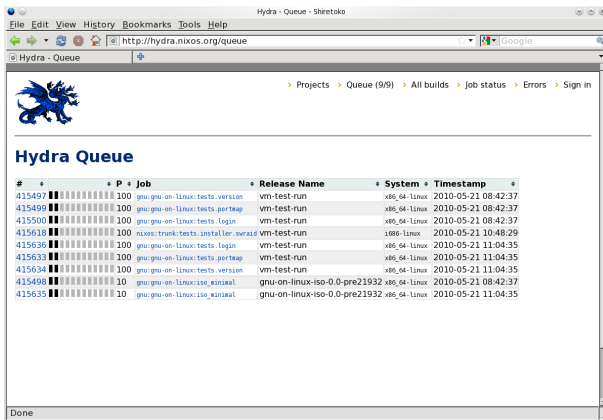


Extensions with custom modules are implemented by composing processes:

- Virtualization extension
- Dynamic deployment
- Service abstraction layer

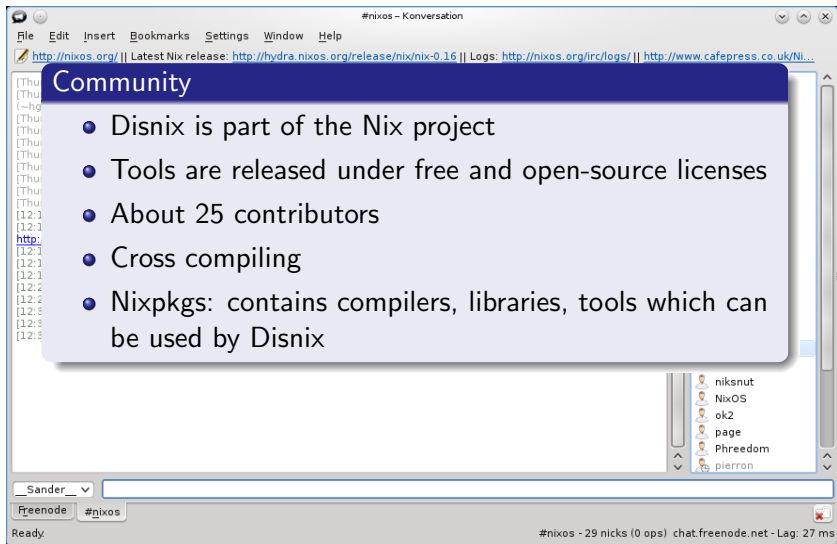


- Generate cheap virtual networks from declarative specifications
- Automatically performs testcases on them
- Completely scripted
- Disnix is continuously integrated and tested by *Hydra*



The screenshot shows a web browser window titled "Hydra - Queue - Shiretoko" with the URL "http://hydra.nixos.org/queue". The page features a blue dragon logo and navigation links: "Projects", "Queue (9/9)", "All builds", "Job status", "Errors", and "Sign in". The main content is titled "Hydra Queue" and displays a table of jobs with columns for ID, progress, priority, job name, release name, system, and timestamp.

#	Progress	P	Job	Release Name	System	Timestamp
415497	██████████	100	gnu:gnu-on-linux:tests.version	vm-test-run	x86_64-linux	2010-05-21 08:42:37
415499	██████████	100	gnu:gnu-on-linux:tests.portmap	vm-test-run	x86_64-linux	2010-05-21 08:42:37
415500	██████████	100	gnu:gnu-on-linux:tests.login	vm-test-run	x86_64-linux	2010-05-21 08:42:37
415618	██████████	100	nixos:trunk:tests.installer.swraid	vm-test-run	i686-linux	2010-05-21 10:48:29
415636	██████████	100	gnu:gnu-on-linux:tests.login	vm-test-run	x86_64-linux	2010-05-21 11:04:35
415633	██████████	100	gnu:gnu-on-linux:tests.portmap	vm-test-run	x86_64-linux	2010-05-21 11:04:35
415634	██████████	100	gnu:gnu-on-linux:tests.version	vm-test-run	x86_64-linux	2010-05-21 11:04:35
415498	██████████	10	gnu:gnu-on-linux:iso_minimal	gnu-on-linux-iso-0.0-pre21932	x86_64-linux	2010-05-21 08:42:37
415635	██████████	10	gnu:gnu-on-linux:iso_minimal	gnu-on-linux-iso-0.0-pre21932	x86_64-linux	2010-05-21 11:04:35



The screenshot shows an IRC chat window titled "#nixos - Konversation". The window has a menu bar with "File", "Edit", "Insert", "Bookmarks", "Settings", "Window", and "Help". Below the menu bar is a status bar with a URL: "http://nixos.org/ | Latest Nix release: http://hydra.nixos.org/release/nix/nix-0.16 | Logs: http://nixos.org/irc/logs/ | http://www.cafepress.co.uk/Ni...". The main chat area contains a list of users: niksnut, NixOS, ok2, page, Phreedom, and pierron. A message box is overlaid on the chat area with the title "Community" and a list of bullet points:

- Disnix is part of the Nix project
- Tools are released under free and open-source licenses
- About 25 contributors
- Cross compiling
- Nixpkgs: contains compilers, libraries, tools which can be used by Disnix

At the bottom of the window, there is a "Ready:" status and a channel name "#nixos". The bottom right corner shows "#nixos - 29 nicks (0 ops) chat.freenode.net - Lag: 27 ms".

- Architecture works well for supporting development
 - Easy prototyping, debugging, testing, extendable
- Our test approach works well; we can automatically perform distributed testcases
- Community is crucial; too much effort to maintain everything ourselves

Disnix, and other related Nix tooling: Hydra, NixOS can be freely downloaded from: `http://nixos.org`

Questions