opsi.org

opsi

client management for heterogenous environments

uib

Speaker

- Niko Wenselowski
  - http://nerdno.de/
  - Passionate Pythonista
  - Works for uib GmbH, Germany
    - http://www.uib.de/
    - Responsible for developing opsi
    - Focus on Linux-side
    - Customising

opsi – the beginning

- Tool to deploy Windows for Workgroups 3.11
  - WfW 3.11 was released in August 1993
  - opsi for unattended deployment
    - Via BOOTP
    - Used *opsi-winst*
  - Installation: copy files to harddisk & unpack
- Server: Solaris
  - Installation media on Samba share

opsi – the past

- Gained more software deployment features
- Centralised management of deployments
- Management Interface
  - Edited config-files on server
- Server moved from Solaris to Debian
- Public release at SourceForge 2004
  - Open source from the beginning

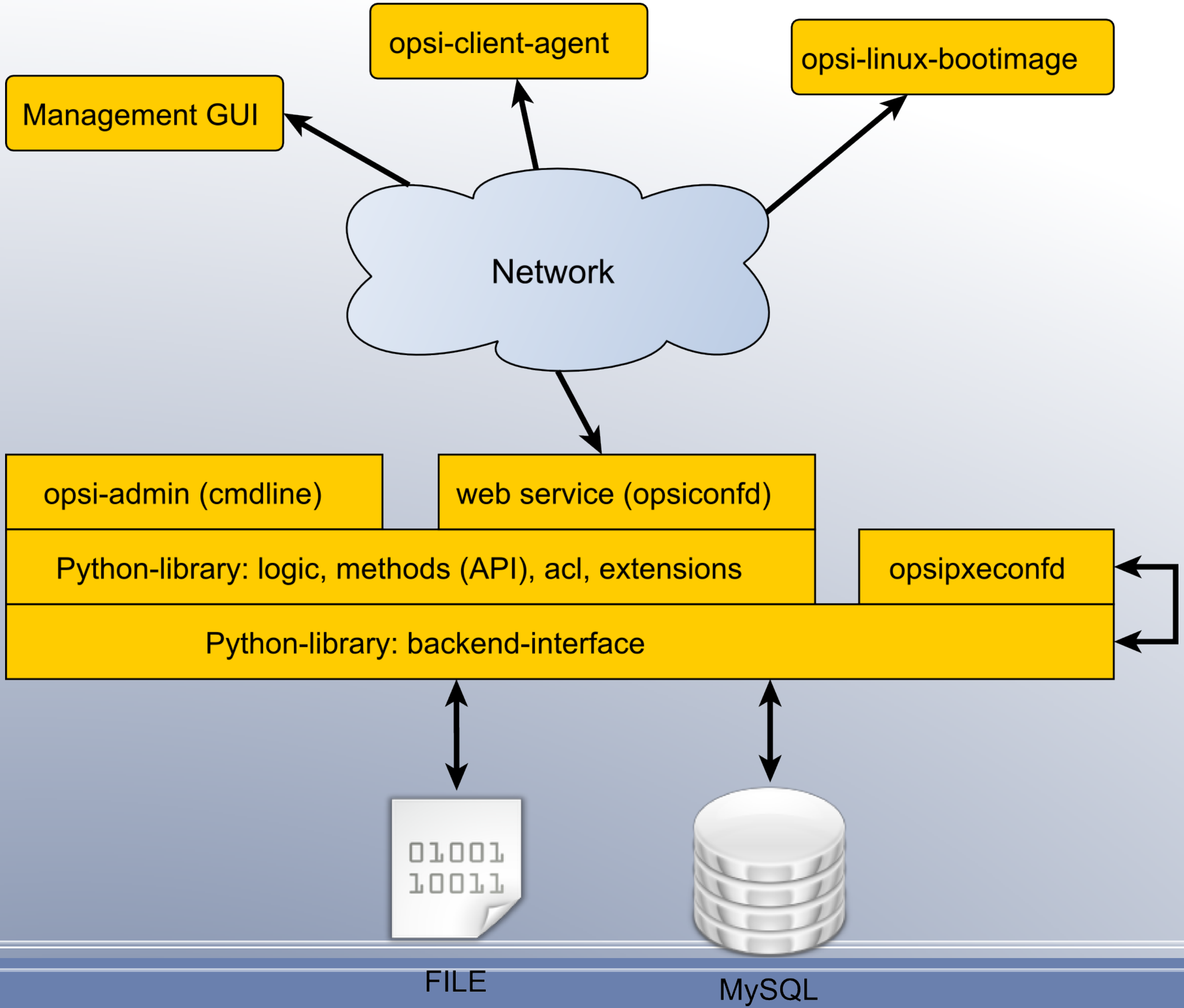
opsi now

- Server: Linux
  - Webservice
    - Communication: JSON-RPC
  - Samba share with install media
- Java-based management interface
- Clients
  - Various Linux distributions (Debian, CentOS, openSUSE, RHEL, SLES, Ubuntu)
  - Windows 7 to 10

opsi now – buzzword edition

- OS deployment
  - Linux & Windows
  - Unattended installation
  - Deploying of images
- Software deployment
  - Linux & Windows
  - Usable for patch management
- Configuration management
- Inventory
  - Hard- & Software

Architecture overview



Overview: Server

- Webservice *opsiconfd*
  - Accessible through JSON-RPC
- Samba
  - Provides files for installation to clients
- dhcpd & tftpd & *opsipxeconfd*
  - Used for PXE boot of clients
  - *opsipxeconfd* writes named pipes
    - Readable once

Overview: Clients

- Graphical management interface *configed*
- Linux bootimage
  - Booted over PXE to prepare OS installation
- Agent *opsi-client-agent*
  - Runs as a service on client
  - Checks if work needs to be done
    - Can be triggered from opsi Server
  - Starts *opsi-script* to handle installation scripts

Software deployment with opsi

- opsi package
  - Contains files to deploy
    - no files → just configuration
  - Script(s) for (de)installation
  - Single archive
- Packages extracted on server
  - Clients access files over SMB

opsi-script: What is it?

- Scripting language for deploying software
- Specific syntax
  - Tailored to various tasks
- Integrate existing scripts / tools
  - No limitation on programming language
- *opsi-script* also name of interpreter
  - Formerly known as *opsi-winst*

One script to handle Win & Linux

- Good idea?
- Example: Thunderbird
  - Win: .exe installer
  - Linux
    - In distro repos?
    - Wanted version in repos?
    - What about plugins?
  - Did I mention configuration?
    - Registry vs. Files

opsi-script: Architecture detection

```
Set $SystemType$ = GetSystemType
if $SystemType$ = "x86 System"
    Comment "on 32 bit"
else
    ; $SystemType$ is "64 Bit System"
    Comment "on 64 bit"
endif
```

opsi-script: OS detection

```
set $OS$ = GetOS

if not ($OS$ = "Linux")
; or: if $OS$ = "Windows_NT"
    isFatalError "wrong OS"
endif
```

opsi-script: Windows release detection

- Win 10 reports API version as *10*
  - Was *6.4* in early versions

```
set $INST_NTVersion$ = GetMsVersionInfo
if
CompareDotSeparatedNumbers($INST_NTVersion$
, "10.0") >= "0"
        ; We are running Win 10
endif
```

- Different Win 10 versions: Check for *ReleaseID* with **getMSVersionMap**

opsi-script: Linux distribution family detection

```
set $distrotype$ = getLinuxDistroType
Switch $distrotype$
        Case "debian"
                ; handle Debian / Ubuntu / UCS
        EndCase
        Case "redhat"
                ; handle CentOS / RHEL
        EndCase
        Case "suse"
                ; handle openSUSE / SLES
        EndCase
EndSwitch
```

- Check for *codename*, *distributor* or *release* with **`getLinuxVersionMap`**

Linux: mind the package lock

- Package managers will lock their resources
  - We usually want to wait for the lock

```
; 5 minutes timeout to get package log
; Do not kill package manager if we don‘t
if waitForPackageLock("300", "false")
    comment "we got the package lock."
else
    LogError "could not get Package Lock"
endif
```

Best practices

- Use opsi-script constants to address locations
  - ie. *%ScriptPath%* or *%ScriptDir%*
  - Avoid hard-coding paths!
  - Auto-conversion for slashes in paths
- Use opsi-script functions
  - Most work on Linux and Windows
- Extend and share your scripting library

Why opsi?

- Works in different environments
  - Can be run without DNS
  - External dhcpd possible
  - Support for multiple locations
- Ready-to-use solution
- Versatile
- Open API
  - Easy to extend (through Python)

What now?

- New to opsi? Try it!
  - http://www.opsi.org/en/download
  - Share your experiences
    - https://forum.opsi.org/
    - https://forum.opsi.org/wiki/
- Already using opsi?
  - Automate!
  - Integrate!

Roadmap - Development

- Improve Linux support
  - Better integration
  - Support new OS
- Improving administrative tools
- Backend cleanup and refactorings

- Dev blog: https://blog.opsi.org/

Roadmap

- Move remaining repos to git
  - Old repos are at https://svn.opsi.org/
  - New repos on Github: https://github.com/opsi-org
- Improve work with community
  - Provide a Contributors License Agreement
    - Looking forward to *Legal and Policy Issues* devroom!
- New opsi.org

Thanks for your time!