HYPERCONVERGENCE MEETS BIG DATA
Hyperconvergence Meets Big Data

Rafael Monnerat
rafael (at) nexedi (dot) com
@ramonnerat

https://lab.nexedi.com/u/rafael
Agenda

- Hyperconvergence with SlapOS
- Big Data with Wendelin
- How to deploy?
- Upload Data
- Jupyter Quick Demos
Nexedi: Largest OSS Publisher In Europe
Stack 100% Open Source

<table>
<thead>
<tr>
<th>Stack Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wendelin Core</td>
</tr>
<tr>
<td>NEO</td>
</tr>
<tr>
<td>ERP5</td>
</tr>
<tr>
<td>SlapOS</td>
</tr>
<tr>
<td>re6st</td>
</tr>
<tr>
<td>fluentd</td>
</tr>
<tr>
<td>Scikit.Learn et al</td>
</tr>
</tbody>
</table>
SlapOS - HyperconvergenceOS

Distributed Cloud on top of Any Provider.

http://community.slapos.org/
SlapOS - HyperconvergenceOS

Cloud Mobile  
Tv Boxes  
Sensors  
IoT Routers

Beyond the data centers....
SlapOS Node

Software Instances
/srv/slapgrid

Software Releases
/opt/slapgrid

SlapOS Kernel
/opt/slapos

SLAPOS NODE

ERP5 Dev
Wendelin
Wordpress
PowerDNS
Apache Frontend
KVM

Supervisord
Buildout
SlapOS Core
Gnu Linux
"One System To Rule Them All"

- CDN/Mesh Networking (Grandenet)
- KVM Clusters for Big Data (Teralab)
- Wendelin Environments for Big Data (Wendelin)
- Development PaaS for Developers (Nexedi)
- Distributed Test Nodes to run Unit Test (Nexedi)
- Automated Ready to Use VMs (VIFIB)
- ChromiumOS images Builder (NayuOS)
Wendelin - Out-Of-Core Pydata

Wendelin Exanalytics Core 100% open source

- Scikit Learn
- NEO
- ERP5
- SlapOS

100% Python

Data Analytics
Distributed Storage
Elastic PaaS
Multicloud Deployment
Multi Data Center

http://www.wendelin.io/architecture
Data Ingestion

Diagram showing the process of data ingestion with components COMP-0, COMP-1, COMP-2, and COMP-N, each connected to a web crawler and fluentd. The diagram also includes elements related to GrandeNet and Wendelin, indicating the integration of out-of-core pydata and Machine Learning.

Europython 2016
SlapOS Deployment (With Token)

```
wget https://deploy.erp5.cn/slapos && bash slapos
 [...] Install Ansible [...]
Starting Ansible playbook:
What is this computer name? (...) [noname]: COMPUTER-NAME
If you have slapos token if you have (...) [notoken]: 20010101-ABDC
```

Keep it simple with single command to type...
# Leave the computer name and token empty
wget https://deploy.erp5.cn/slapos && bash slapos

[... Ansible is installed...]
Starting Ansible playbook:
What is this computer name? (...): [noname]:
If you have slapos token if you have (...): [notoken]:

# Configura Local Master
slapos configure local

# Prepare the computer to run services.
slapos node format --now
Easy Deployment (Client-Only)

```bash
# You can use easy_install or pip
easy_install slapos.core

pip install slapos.core

slapos configure client
```
Supplying And Requesting Monitor (Fluentd)

```
# Supply will provide make the computer deploy the
# "product.monitor" software on COMPUTER with reference COMP-1239
slapos supply https://lab.nexedi.com/nexedi/slapos/raw/1.0.33/software/monitor/software.cfg COMP-1239

# The Request will ask to the COMP-1239 instantiate one instance
# of the Software Release "product.monitor"

  slapos request my_first_instance
  https://lab.nexedi.com/nexedi/slapos/raw/1.0.33/software/monitor/software.cfg
  --parameters item=True --node computer_guid=COMP-1239

# You can also use alias for give me the latest monitor release
slapos supply product.monitor COMP-1239

# By not passing --node, your instance will be allocated on any computer
# has the wanted software release (respecting security roles of your user)
slapos request my_first_instance product.monitor --parameters item=True
```

Monitor contains fluentd
Supplying And Requesting Wendelin Stack

# Supply will provide make the computer deploy the
# "product.monitor" software on COMPUTER with reference COMP-1239

   slapos supply https://lab.nexedi.com/nexedi/slapos/raw/1.0.33/software/wendelin/software.cfg COMP-1239

# The Request will ask to the COMP-1239 instantiate one instance
# of the Software Release "product.monitor"

   slapos request my_first_instance
   https://lab.nexedi.com/nexedi/slapos/raw/1.0.33/software/wendelin/software.cfg
   --parameters item=True --node computer_guid=COMP-1239

Monitor contains fluentd
Deploying Wendelin (Standalone)

`wget https://deploy.erp5.cn/wendelin-standalone && bash wendelin-standalone`
Ready To Use VMs (Soon)

Not ready yet but soon images will be released for qemu, ec2, digital ocean, VMware...
Uploading Your Wavs

Create configuration file

```yaml
@type bin
format none
path
/srv/slapgrid/slappart9/srv/runner/PUT_YOUR_WAV_HERE/*.wav
pos_file
/srv/slapgrid/slappart9/srv/runner/Demo.pos
enable_watch_timer false
read_from_head true
tag wavdemo

@type wendelin
@id wendelin_out

  streamtool_uri https://softinst11111.host.vifib.net/erp5/portal_ingestion_policies/wavdemo
  user      zzz
  password  yyy

  buffer_type  memory
  flush_interval  1s
  disable_retry_limit  true
```

and them run `fluentd -c configuration.cfg`
Files Uploaded!

The files are uploaded...
Wendelin Modules Overview

- Ingestion Policies
- Data Stream
- Data Arrays
Jupyter Wav Demos

Wav Demo Jupyter Notebook
Jupyter Wendelin Async

CMFActivities Jupyter Notebook
Javascript-Based Gadgets

Pip Install Wendelin.Core

$ ipython
Python 2.7.11+ (default, Jun 2 2016, 19:34:15)
...
# imports
In [1]:
from wendelin.bigarray.array_zodb import ZBigArray
In [2]:
from wendelin.lib.zodb import dbopen, dbclose
In [3]:
import transaction
In [4]:
import numpy as np
# open/create database for tests (on local disk for now)
In [5]:
root = dbopen('test.fs')
# create 10 items 1d array object
In [6]:
root['A'] = A = ZBigArray((10,), np.int)
In [7]:
transaction.commit()
# see what it is
In [8]:
A
Out[8]:

In [9]:
a = A[:]
In [10]:
a
Out[10]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
In [11]:
type(a)
Out[11]: numpy.ndarray
Wendelin Core Quick Tutorial

Thank You

Rafael Monnerat
rafael (at) nexedi (dot) com
@ramonnerat

We are hiring! https://www.nexedi.com/jobs